Cygna Forest Recovery for Active Directory

User Guide



For the latest information, visit online documentation at docs.cygnalabs.com

Published 11/2/2026

Version 26.1

Copyright © 2026

Copyright

©2026 Cygna Labs Corp. ALL RIGHTS RESERVED.

Trademarks

Cygna Labs and the Cygna Labs logo are trademarks and registered trademarks of Cygna Labs Corp. in the United States of America and other countries. All other trademarks are property of their respective owners.

Disclaimers

The product documentation is subject to change without notice. For the latest and more detailed documentation, please refer to online documentation at https://docs.cygnalabs.com.

The product functionality described in this document shall not be treated as a public offer or commitment.

The information regarding the use and installation of third-party software is provided to assist you but Cygna Labs Corp. shall not accept any responsibility or liability for any claims or damages caused by incorrect or incomplete information provided about third-party software. For detailed instructions on configuring third-party software components, refer to their respective owners.

Contents

Introduction 10
Why Cygna Forest Recovery 10
What Cygna Forest Recovery does 10
In this chapter 11
Components and Architecture 11
Supported Directories 18
Using This Guide 19
Related Cygna Products 21
Planning Your Deployment 23
Machines you need 23
Keep the Manager and storage machine out of the domain 23
Separate or shared Manager and storage machine 24
Where to put the Isolated Recovery Host 24
Moving a machine later 25
Give both machines fixed IP addresses 25
Network placement and isolation 25
Scale 26
Deployment checklist 26
In this chapter 27
System Requirements 27
Ports and Protocols 33
Accounts and Permissions 36
Certificates and Trust 41
Choosing Which Domain Controllers to Back Up 44
Sizing Backup Storage 46
Installing and Setting Up 52
Order of operations 52
What you need on hand 53

In this chapter 54
Quick Install Guide 54
Installing the Manager 62
Signing In for the First Time 66
Trusting the Manager Certificate 70
Setting Up the Storage Machine 73
Deploying DC Agents 79
Setting Up OS Media and Recovery Images 89
Creating Your First Backup Policies 93
Setting Up an Isolated Recovery Host 96
Upgrading and Uninstalling 102
How Backup Works 108
The path of a backup 108
Why this design is safe 109
Who does what 110
In this chapter 110
Backup Types 110
Restore Points 112
Vault Protection and Integrity 116
Retention 121
Recovery Images 125
Test Restores 129
Coverage and Readiness 135
How Recovery Works 141
Plan first 141
Separation of roles 141
Scopes 142
Methods 142
It works when the directory does not 143
In this chapter 143

Recovery Scopes 144
Alignment with Microsoft Guidance 149
Restore Methods 153
Choosing a Recovery Approach 158
Recovery Lifecycle 161
Recovery Requirements 167
Isolating Recovery Targets 173
What Recovery Does Not Cover 178
Using the Console 181
Opening the console 181
Signing in and out 181
Console layout 186
Roles and what you can see 190
Notifications 191
The recovery banner 191
The license banner 191
Working with screens and tables 192
In this chapter 194
Dashboard 195
Backup Targets and Rebuilt Domain Controllers 201
Forests 205
Domains 213
Sites 221
Domain Controllers 227
Backup Policies 239
OS Media 246
Backup Jobs 255
Restore Points Screen 262
Repository 267
Recovery Plans 273

Alerts ... 299
Protection Summary ... 303
Audit Trail ... 307
Agents ... 312
Operators and Roles Screen ... 327
Configuration ... 335
Backup Tasks ... 341
Who can do what ... 341
Before you start ... 341
In this chapter ... 341
Managing Backup Policies ... 342
Running a Backup Now ... 349
Monitoring and Cancelling Backup Jobs ... 352
Running a Test Restore ... 355
Excluding and Retiring Objects ... 359
Recovery Procedures ... 364
Pick the procedure ... 364
How every recovery runs ... 365
Before any recovery ... 365
In this chapter ... 365
Restoring a DC with DC State Recovery ... 366
Full System Recovery in Place ... 376
Full System Recovery onto a Candidate ... 384
Full System Recovery onto a New VM ... 389
Recovering a Domain ... 393
Recovering a Forest ... 404
After a Recovery ... 421
Administration ... 426
Who administers what ... 426
Where settings live ... 426

Watching the installation ... 426
In this chapter ... 427
Managing Operators and Roles ... 427
Multi-Factor Authentication ... 436
Settings Reference ... 444
Licensing ... 482
Maintaining Agents ... 492
Moving the Storage Machine or an Isolated Recovery Host ... 504
Alerts Reference ... 509
Auditing Operator Activity ... 528
Logs and Diagnostics ... 533
Protecting the Manager ... 540
What makes a Manager ... 540
Ways back ... 541
Watching the Manager's readiness ... 542
In this chapter ... 543
Manager Day-One Checklist ... 543
Manager Configuration Backups ... 545
Recovering the Manager ... 550
Re-Keying the Manager CA ... 556
Troubleshooting ... 561
General approach ... 561
Where to look ... 561
Using the command line ... 562
Collecting a support package ... 562
In this chapter ... 563
Preflight Check Reference ... 563
Backup Job Errors ... 595
Agent Enrollment and Connectivity ... 600
Installer Messages ... 607

Command-Line Reference 613
Where cdr is installed 613
Running cdr 613
Global options 614
Connecting to the Manager 614
Commands that act outside the Manager 615
Output 615
Confirmations and secrets 616
Exit codes 616
Environment variables 617
Permissions 618
Manager maintenance commands 619
Command groups 619
In this chapter 620
Connecting and Signing In 621
Inventory Commands 627
Agent and Trust Commands 634
Backup Commands 647
Storage and Media Commands 655
Restore Commands 664
Recovery Plan Commands 677
Alert, Audit, and Report Commands 690
Configuration and Exclusion Commands 696
Directory Cleanup Commands 706
Support Commands 714
Glossary 718
A 718
B 718
C 719
D 720

E 720
F 721
G 721
H 721
I 722
L 722
M 722
N 723
O 723
P 723
R 723
S 725
T 726
U 726
V 726
Index 727

Introduction

Cygna Forest Recovery for Active Directory backs up Active Directory and guides you through recovering it, from a single domain controller up to an entire forest. It is built for the worst case: a ransomware attack or other compromise that leaves Active Directory unusable, along with the credentials an attacker would normally use to reach your backups.

This guide describes Cygna Forest Recovery 26.1. It is part of the Cygna Disaster Recovery Suite. Its components, services, and messages carry the suite's short name, Cygna DR: for example, the Cygna DR Manager and the `cdr` command-line tool. Installer packages and email subjects spell it out as Cygna Disaster Recovery, for example Cygna Disaster Recovery DC Agent in **Programs and Features**.

Why Cygna Forest Recovery

When an attacker takes control of a directory, the backups are often the next target. They are usually reachable with the same domain credentials the attacker now holds, and restoring them depends on infrastructure that may itself be down. Cygna Forest Recovery is designed around three principles that address this:

- **Backups are out of the directory's reach**—Backups are kept in a vault on a machine that is not joined to the domain. No domain account has rights to the vault, and between a backup and a restore no network credential exists that can read or delete a backup.
- **It works when the directory does not**—The Manager, its console, and its operator accounts do not depend on Active Directory or DNS. You can sign in, review restore points, and run a recovery while every domain controller is offline.
- **Recovery is a checked, audited plan**—Each recovery is planned before anything changes. Preflight checks confirm that the chosen restore point and target machine can succeed, approval and execution are separate permissions, and steps that cannot be undone require typed confirmation. Every action is recorded in the audit trail.

What Cygna Forest Recovery does

- **Discovers your environment**—Agents on your domain controllers report the forests, domains, sites, and domain controllers they find, and the console shows which of them are protected.
- **Schedules and stores backups**—The Manager runs full system or DC state backups on the schedule you set for each domain controller, and the storage agent verifies each backup and moves it into the vault.

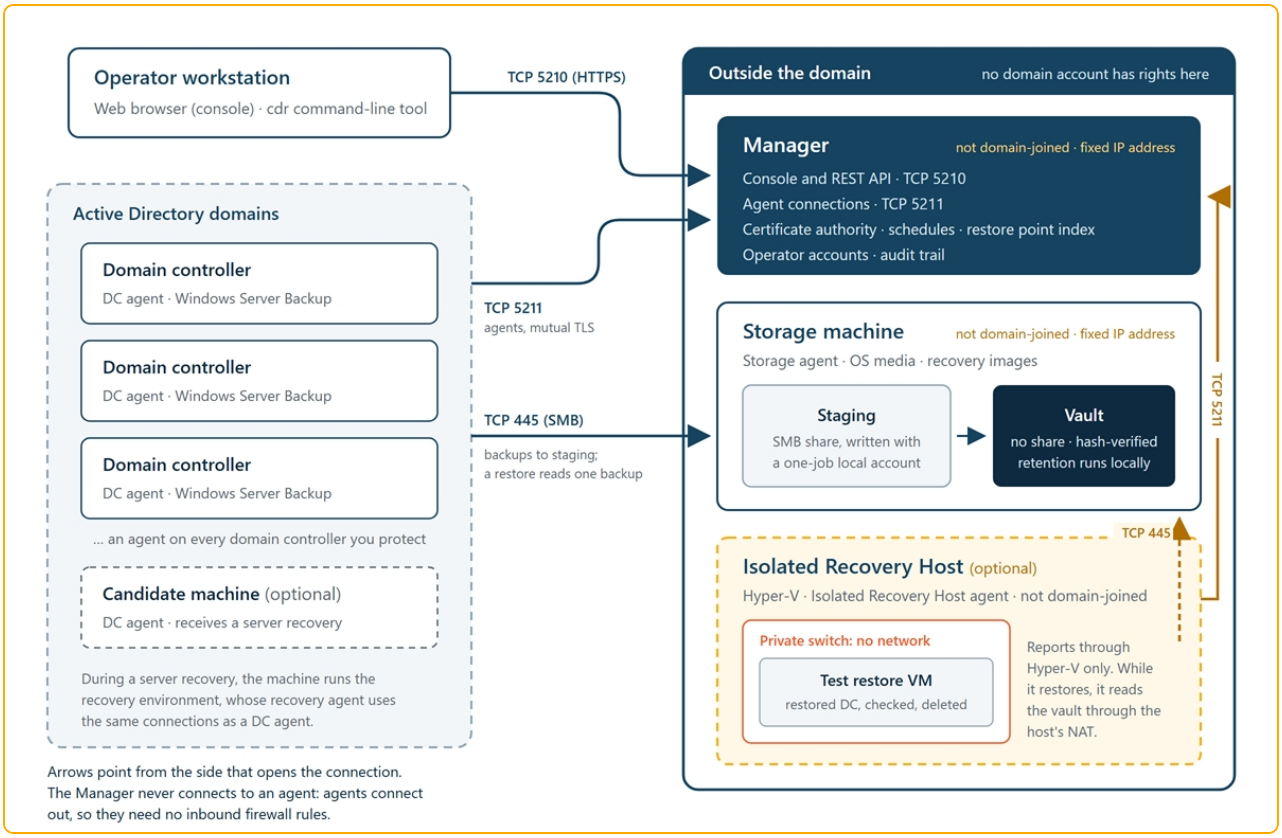
- **Keeps backups usable**—Stored backups are re-verified on a schedule, and retention rules decide how much history is kept for each domain controller.
- **Prepares for full system recovery**—From Windows Server installation media you supply, the storage agent builds the recovery images needed to restore a whole server onto the same or a replacement machine.
- **Orchestrates recovery**—Restore one domain controller, recover one domain in a running forest, or recover the whole forest following the Microsoft forest recovery procedure, with each step tracked in the console.
- **Reports on readiness**—The dashboard, alerts, and reports show whether each domain and forest can be recovered, and what to fix if it cannot.

In this chapter

- Components and Architecture
- Supported Directories
- Using This Guide
- Related Cygna Products

Components and Architecture

Cygna Forest Recovery is made up of a central Manager, agents on the machines it protects and on the machine that stores the backups, a recovery environment used to restore whole servers, and an optional Isolated Recovery Host on which backups are test-restored. This topic describes each component, where it runs, and how the components communicate.



Components at a glance

Component	Where it runs	What it does
Manager	A Windows Server machine that is not joined to any domain	Schedules backups and holds the inventory, the restore point index, recovery plans, settings, operator accounts, and the audit trail. Serves the console and the REST API, and runs its own certificate authority.
Console	Served by the Manager and opened in a web browser	The web interface for monitoring protection, managing backups, and planning and running recoveries.
`cdr` command-line tool	The Manager machine	Performs the same operations as the console from a command prompt or a script, plus deployment, sizing, and directory cleanup tasks.
DC agent	Every domain	Reports the directory topology, runs backups, and carries out restores and recovery plan

Component	Where it runs	What it does
	controller, and any candidate machine prepared to receive a recovered server	steps on its machine.
Storage agent	The storage machine, which is not joined to any domain	Receives backups, verifies them and moves them into the vault, applies retention, and builds recovery images.
Recovery agent	Inside a recovery image, on a machine being recovered	Restores a whole server from a full system backup when the server's own Windows cannot be used.
Isolated Recovery Host agent (optional)	A Hyper-V host that is not joined to any domain	Builds isolated virtual machines in which backups are test-restored and checked, and reports whether the host is ready.

Manager

The Manager is the control point of the deployment. It runs as the Windows service `CygnaDRManager` under the LocalSystem account and hosts its own web server, so it needs no IIS. Its data is kept in an embedded database, `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\cygnadr.db`, so no SQL Server is needed either.

The Manager:

- Decides when each domain controller is backed up and sends the job to its agent. Agents never start backups on their own.
- Keeps the inventory of forests, domains, sites, and domain controllers that the agents report, and the index of restore points.
- Builds, checks, and coordinates restore runs and recovery plans.
- Stores the operator accounts, roles, settings, and audit trail.
- Runs the certificate authority that secures every connection to it.

The Manager never reads or writes the vault. Only the storage agent, running on the storage machine, has access to stored backups.

Console

The console is a web application that the Manager serves on HTTPS port 5210. You open it in a web browser at `https://<manager address>:5210/`. If the redirect port is enabled, `http://<manager address>/` redirects you to the HTTPS address.

You sign in with a Cygna Forest Recovery operator account. These accounts are kept by the Manager, not by Active Directory, so you can sign in while the directory is unavailable. For a tour of the console, see Using the Console.

`cdr` command-line tool

The `cdr` command-line tool (`cdr.exe`) is installed with the Manager, in its program folder `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager`. It is not added to the system path, so run it from that folder or from a folder you copy it to. On the Manager machine it connects to `https://localhost:5210` by default.

`cdr` uses the same operator accounts and permissions as the console. A few commands work without the Manager, such as estimating backup storage before you deploy. For details, see Command-Line Reference.

DC agent

The DC agent runs as the Windows service `CygnaDRDcAgent` under the LocalSystem account. You install it on every domain controller, including read-only domain controllers. The DC agent:

- Reports the topology it can see in the directory, so the console shows every forest, domain, site, and domain controller, including those with no agent.
- Runs each backup with Windows Server Backup and writes it to a staging folder on the storage machine.
- Records the hardware and network configuration of its machine, which becomes part of each restore point.
- Carries out restores and recovery plan steps on its machine, using standard Windows tools such as `ntdsutil`, `nltest`, `w32tm`, `dcdiag`, `repadmin`, and PowerShell.
- Prepares the recovery image on its machine when a server is recovered in place, and applies network settings and other follow-up instructions when the recovered server first starts.

You also install the DC agent on a *candidate* machine: a machine that is not a domain controller, prepared to receive a full system recovery when the original domain controller cannot start. For more information, see Restore Methods.

Storage agent

The storage agent runs as the Windows service `CygnaDRStorageAgent` under the LocalSystem account on the storage machine. It owns two folders under the storage root you choose at installation: `staging`, where domain controllers write new backups, and `vault`, where committed backups are kept. The storage agent:

- Creates a short-lived local account for each backup job and removes it when the job ends.
- Computes a hash of each backup and moves it from staging into the vault.
- Applies the retention rules that decide which backups are kept, and re-verifies stored backups against their hashes on a schedule.
- Reports the capacity of the vault to the Manager.
- Scans a folder of Windows Server installation media (ISO files) that you supply, and builds the recovery images used for full system recovery. It uses the DISM tool included in Windows, so you don't need the Windows Assessment and Deployment Kit.

An installation has one storage machine, the first storage agent to enroll. Another storage agent that enrolls is given no work until an operator makes it the storage machine in place of the first. The storage agent also creates local groups on the storage machine that control access to staging and the vault. For details, see Accounts and Permissions.

Recovery agent and recovery images

A recovery image is a Windows PE environment that the storage agent builds from the Windows Recovery Environment in your own Windows Server installation media, with the recovery agent added. It is not a copy of the domain controller's own operating system. One image is built for each operating system variant in use.

You don't install or enroll the recovery agent separately. It is included with the storage agent and copied into every recovery image. During a full system recovery, the machine being recovered starts the recovery image once from memory. The recovery agent then reads the selected backup from the storage machine, restores the whole server, and reports its progress to the Manager using a credential that is valid for that run only. For more information, see Recovery Images.

Isolated Recovery Host agent

An Isolated Recovery Host is a Hyper-V host that you dedicate to Cygna Forest Recovery, on the same network as the Manager and the storage machine. It is optional: backup and recovery never depend on it. It runs the Isolated Recovery Host agent as the Windows service `CygnaDRHostAgent` under the LocalSystem account. The agent:

- Checks the host and reports whether it can run test restores: Hyper-V, its virtual switches, disk space, memory, and domain membership.
- Installs Hyper-V when it is missing, and creates the recovery network that test restore virtual machines use.
- For each test restore, builds a virtual machine shaped like the backed-up domain controller, moves it to a private switch of its own before the restored system first starts, relays the results of the restored system's checks, and deletes the virtual machine afterward.

A host is given test restores only once an operator approves it, and you can approve more than one. Like the storage machine, the host is not joined to any domain, and no credential is stored for it. The Manager never connects to the host; it asks the agent, and the agent enforces the isolation itself. For more information, see Test Restores.

How the components communicate

- **Agents connect to the Manager**–DC agents, storage agents, Isolated Recovery Host agents, and recovery agents open connections to the Manager on TCP port 5211. The Manager never opens a connection to an agent; it sends work, such as a backup job or a recovery step, over the connection the agent keeps open. Because agents only connect outward, they need no inbound firewall rules.
- **Operators connect to the Manager**–The console and `cdr` connect to the Manager on TCP port 5210 over HTTPS.
- **Domain controllers connect to the storage machine**–Domain controllers write backups to the staging share over SMB, and a domain controller or recovery environment reads a backup over SMB during a restore. A test restore reads its backup through the Isolated Recovery Host.
- **The Manager connects to a domain controller only to install the DC agent**–When you deploy the DC agent from the console, the Manager connects to the target to copy and run the installer. Normal operation needs no connection from the Manager to a domain controller.

Every connection to the Manager is encrypted with TLS, and agents and the Manager authenticate each other with certificates issued by the Manager's certificate authority. For details, see Certificates and Trust. For the full list of ports, see Ports and Protocols.

Addressing by IP address

Agents reach the Manager by its IP address, and the storage agent gives domain controllers its own IP address as the staging location. There are three reasons:

- The Manager and storage machine are not domain members, so their names are usually not in any DNS zone that the domain controllers use.

- The directory, and with it DNS, may be down at exactly the moment you need to recover.
- A host name can be redirected. If DNS were poisoned during an attack, a name could send backup data to a machine the attacker controls.

This is why the Manager and storage machine need fixed IP addresses. For more information, see Planning Your Deployment.

Off-domain storage design

Backups are kept on a storage machine that is not joined to any domain, and no domain account has rights to it. Access to backups is granted only through local accounts that the storage agent creates for one job and removes afterward.

Each backup follows the same sequence:

1. The Manager starts a backup job. The storage agent creates a local account for that job, named `cdr-` followed by 12 hexadecimal characters, with write access to a staging folder for that job only.
2. The DC agent writes the backup to that folder over SMB, using the job's account.
3. The storage agent computes a hash of the backup content and records it.
4. The storage agent revokes the job's access: it closes the account's SMB sessions, removes its permissions, and deletes the account.
5. The storage agent moves the backup into the vault. The vault has no network share and grants no access to any account the storage agent creates.

As a result, between a backup and a restore, no credential exists that can read or delete the backup over the network.

When a restore needs a backup, the storage agent first confirms that the backup still matches its hash. It then shares that one backup read-only, to a separate account created for the restore (named `cdr-r-` followed by 12 hexadecimal characters), and removes the share and the account when the restore has read it.

Backups are deleted only by the retention rules that the storage agent applies locally. No network interface can delete or overwrite a stored backup. The most protective retention limits are set on the storage machine itself, and the Manager can tighten them but never weaken them. For more information, see Vault Protection and Integrity and Retention.

Note: You can run the Manager and the storage agent on the same machine, but separate machines are recommended. On a shared machine, anyone who gains control of the Manager is also on the machine that holds the backups. For guidance, see Planning Your Deployment.

Working when the directory is down

Cygna Forest Recovery is designed to keep working when Active Directory and DNS are not available:

- Operator accounts are kept by the Manager, so you can sign in without the directory.
- The Manager runs its own certificate authority, so secure connections don't depend on a domain certificate service.
- Components find each other by IP address, so they don't depend on the directory's DNS.

Supported Directories

Cygna Forest Recovery 26.1 protects Active Directory Domain Services.

Active Directory

- **Domain controller operating systems**—Windows Server 2016 through Windows Server 2025, in both the Server Core and Desktop Experience installation options.
- **Forests and domains**—One or more forests, each with one or more domains.
- **Recovery scopes**—A single domain controller, one domain in a forest that is otherwise running, or an entire forest. For details, see Recovery Scopes.

Read-only domain controllers

Install the DC agent on read-only domain controllers (RODCs) as well, so they are part of the inventory and can take part in recovery steps. An RODC is never restored from a backup. When a recovery includes an RODC, it is rebuilt instead: its old directory data is cleaned up and it is promoted again. For the same reason, a recovery cannot start from an RODC, so each domain needs at least one writable domain controller with a usable backup.

What is not recovered

Some services that often run on domain controllers, such as Active Directory Certificate Services and DHCP, are not recovered by Cygna Forest Recovery. For the full list, see What Recovery Does Not Cover.

How this guide handles directory-specific content

Most of Cygna Forest Recovery works the same way regardless of the directory it protects: the Manager, the storage machine, backup storage and retention, operator accounts,

alerts, and the audit trail. Topics about these subjects use general terms such as *protected directory*.

Content that applies only to Active Directory is in separate topics, placed in the chapters where you need them:

- Installing the DC agent on domain controllers, in Deploying DC Agents.
- The Active Directory screens of the console, which appear under **Active Directory** in the navigation: Forests, Domains, Sites, and Domain Controllers.
- What can be recovered and how, in Recovery Scopes, Recovery Procedures, and What Recovery Does Not Cover.

Using This Guide

This guide is for everyone who deploys, operates, or oversees Cygna Forest Recovery. You don't need to read it from start to finish. Find your role below and start with the chapters listed for it.

Where to start

If you	Start with
Plan and install the deployment	Planning Your Deployment, Installing and Setting Up, and Manager Day-One Checklist. Read How Backup Works to understand what you are setting up.
Manage backups (Backup Operator role)	How Backup Works, Backup Tasks, Using the Console, and Coverage and Readiness.
Plan and run recoveries (Restore Operator role)	How Recovery Works, Choosing a Recovery Approach, Recovery Requirements, and Recovery Procedures. Read these before you need them.
Administer the installation (System Admin role)	Administration, Protecting the Manager, and Troubleshooting.
Review activity and readiness (Viewer / Auditor role)	Auditing Operator Activity, Protection Summary, Dashboard, and Coverage and Readiness.

The roles named above are the operator roles in Cygna Forest Recovery. Each role sees only the console screens it can use. For details, see Managing Operators and Roles.

If you work from the command line, see Command-Line Reference. For definitions of the terms used in this guide, see the Glossary.

How the guide is organized

- **Introduction**–What Cygna Forest Recovery is, its components, and the directories it supports.
- **Planning and installation**–Requirements, topology decisions, and the steps to install and set up each component.
- **Concepts**–How backup and recovery work, and why. Read these to make decisions.
- **Console**–What each console screen shows and what you can do from it.
- **Tasks and procedures**–Numbered steps for backup tasks and for each kind of recovery.
- **Administration and Manager protection**–Operators and roles, settings, agents, logs, and how to protect and recover the Manager itself.
- **Troubleshooting and reference**–Error messages, preflight checks, alerts, and the `cdr` command-line tool.

Conventions

Convention	Meaning
Bold	A label in the console, such as a screen, button, field, or tab name. For example, select **Back up now**.
Active Directory > **Domain Controllers**	A path through the console navigation or a series of menus.
`Monospace`	Commands, file and folder paths, setting keys, installer properties, and values that you enter.
`<value>`	A placeholder in a command or path. Replace it, including the angle brackets, with your own value.
Select	Click or tap a control in the console.
Enter	Type a value into a field or at a command prompt.

Notes contain information that helps you avoid a problem or understand a consequence. A note that starts with **Important** describes something that can affect your ability to recover, and you should read it before you continue.

Command examples use sample values such as `corp.example.com` and `10.0.0.10`. Replace them with the values for your environment.

Related Cygna Products

Cygna Forest Recovery recovers domain controllers, domains, and forests when the directory itself is damaged or compromised. Other Cygna Labs products cover the everyday changes that lead up to that point: who changed what, whether the change was risky, and how to undo it. They are separate products, installed and licensed separately. Cygna Forest Recovery does not connect to them, and you don't need any of them to use Cygna Forest Recovery.

Cygna Auditor

Cygna Auditor records changes across Active Directory, Entra ID, Microsoft 365, file systems, and other sources, with search, reports, and alerts that can be sent to a SIEM or Syslog server.

- **Risk Radar**—Detects unusual activity in the audit data, such as sign-ins from an unusual location or a burst of failed sign-ins, and runs a daily assessment of Active Directory security posture: privileged groups, Kerberos settings, delegation, identity hygiene, and trusts. Risk Radar reports what it finds; it doesn't change anything.
- **Recovery for Active Directory module**—Rolls back individual objects and attributes to earlier values, and recovers deleted objects. It works on objects, not on whole domain controllers, domains, or forests.
- **Recovery for Entra ID module**—Recovers and rolls back Entra ID users and groups.

Each recovery module is licensed separately from Cygna Auditor's auditing.

Choosing the right product

Situation	Use
A user, group, or attribute was changed or deleted by mistake	The Recovery for Active Directory or Recovery for Entra ID module of Cygna Auditor. A directory restore is not needed to undo a few object changes.
You need to know who made a change, or to be alerted to risky	Cygna Auditor and Risk Radar.

Situation	Use
activity	
A domain controller is corrupted or will not start	Cygna Forest Recovery. See Choosing a Recovery Approach.
A domain or the whole forest is compromised or lost	Cygna Forest Recovery. After the recovery, Cygna Auditor's change history and Risk Radar's posture assessment can help you review the recovered directory.

For more about these products, see the Cygna Auditor documentation or contact Cygna Labs.

Planning Your Deployment

A Cygna Forest Recovery deployment is small: one Manager, one storage machine, and an agent on each domain controller. The decisions you make before installing matter, though, because they decide whether your backups stay out of an attacker's reach and whether the product still works when the directory does not. This chapter covers those decisions and the requirements behind them.

For a description of each component, see Components and Architecture.

Machines you need

- **Manager machine**—A Windows Server machine that is not joined to any domain. It runs the Manager service, the console, and the `cdr` command-line tool.
- **Storage machine**—A Windows Server machine that is not joined to any domain, with a volume large enough for your backups. It runs the storage agent and holds the vault, your Windows Server installation media, and the recovery images.
- **Domain controllers**—Every domain controller you protect runs the DC agent.
- **Operator workstations**—Any machine with a current web browser that can reach the Manager.
- **Isolated Recovery Host (optional)**—A Hyper-V host that is not joined to any domain, on which test restores of your backups run. A dedicated machine is recommended. For sizing, see System Requirements. Without one, everything else works and backups are shown as not verified.
- **Candidate machines (optional)**—Machines that can receive a recovered domain controller when the original cannot start. You can prepare these in advance or when you need them.

Keep the Manager and storage machine out of the domain

Neither the Manager machine nor the storage machine may be joined to a domain. This is the basis of the product's protection:

- An attacker who controls the directory gains no rights on either machine, because no domain account has any.
- Both machines keep working, and you can sign in to the console, while every domain controller is offline.

Manage both machines with local accounts that are not used anywhere else, and keep the number of people who can sign in to them small. Avoid management tools that would give someone with domain rights a path to them.

Never use a domain controller

A domain controller is always a domain member, so it can never be the Manager machine or the storage machine. Running everything on one domain controller is not a supported arrangement, even for testing.

Separate or shared Manager and storage machine

Separate machines are recommended. You can run the Manager and the storage agent on one machine if you need to save a server; this is supported.

The choice is about how much an attacker gains from one compromised machine. With separate machines, the Manager has no way to delete or read stored backups: it has no access to the vault at all. On a shared machine, anyone who gains control of the Manager is already on the machine that holds the backups. If you share a machine, protect it as carefully as you would protect the vault on its own.

Where to put the Isolated Recovery Host

If you use an Isolated Recovery Host, give it a machine of its own. The rules for the Manager and the storage machine apply to it too: it must not be joined to a domain, and it can never be a domain controller or share a machine with one. Its disks briefly hold copies of the directory database during each test restore.

Installing the host agent on the storage machine or the Manager machine is supported, but it has costs:

- A restored domain controller runs on that machine. If code in a restored backup ever escaped its virtual machine, it would land on the machine that holds every backup, or beside the Manager's certificate authority key and database.
- On the storage machine, test restores compete with backups for disk and network.
- If the shared machine is itself a virtual machine, Hyper-V needs nested virtualization.
- The agent does not restart a shared machine to finish installing Hyper-V; you restart it yourself.

Place the host on the same network as the Manager and the storage machine. Test restore virtual machines reach both through the host while they restore, so the storage machine must accept SMB from the host's address. The restored systems themselves run on a private network that reaches nothing. For the setup, see Setting Up an Isolated Recovery Host.

Keep the host on current Windows updates for as long as you use it. A host that is behind on updates can fail to start a restored domain controller that is more current than the host.

Moving a machine later

A trial is often set up on temporary machines and moved into production afterward. Plan for the difference between the two machines that can move:

- **Storage machine**—Restore points do not move. Replacing the storage machine ends every restore point on the old one, and backups start again from zero on the new one, where every domain controller with a backup policy is backed up at once. It needs both a System Admin and a Backup Operator, or one operator holding both roles. If you can, put the storage machine where it will stay from the start.
- **Isolated Recovery Host**—Nothing is lost. Approve the new host, drain the old one, and remove it. Any number of approved hosts can share the test restores.

For the procedures, see Moving the Storage Machine or an Isolated Recovery Host.

Give both machines fixed IP addresses

Give the Manager machine and the storage machine static IP addresses, or DHCP reservations that never change. Agents are configured with the Manager's IP address when you install them, and the storage agent gives domain controllers its own IP address as the staging location. Neither is updated automatically if the address changes. If an address changes, agents keep connecting to the old one and backups fail until the agents are reconfigured.

IP addresses are used instead of names because neither machine is in the domain's DNS, because DNS may be down when you need to recover, and because a name can be redirected to another machine.

If the storage machine has more than one network address, set the address that domain controllers should use with the **Advertised host** setting on the **Storage agents** tab of the **Configuration** screen. For details, see Settings Reference.

Network placement and isolation

Place the Manager and storage machine where they are reachable from every domain controller you protect, but not from the rest of your network more than necessary:

- Every domain controller must reach the Manager on TCP 5211 and the storage machine on TCP 445.
- Operator workstations must reach the Manager on TCP 5210, and optionally on TCP 80 for the redirect to HTTPS.

- Nothing else needs to reach the storage machine. Limit inbound SMB on it to the networks your domain controllers and recovery targets use.

Plan for recovery as well as daily operation. During a forest recovery, you isolate the domain controllers being recovered from the rest of the network. The Manager and storage machine must stay reachable from those domain controllers throughout, and from the operator workstation you will use. A candidate machine needs the same access as the domain controller it replaces.

For the complete list of connections, see Ports and Protocols.

Scale

One Manager and one storage machine serve the whole deployment.

As the environment grows, the main constraints are the size of the storage volume and how many backups run at the same time. Every backup is a full copy, so storage grows with the number of domain controllers you back up and the number of versions you keep for each. You don't have to back up every domain controller: a few per domain can be restored and the others rebuilt during a recovery. To decide which ones to back up, see Choosing Which Domain Controllers to Back Up. The **Maximum concurrent backups** setting limits how many backups run at once. To work out the storage you need, see Sizing Backup Storage.

Deployment checklist

Use this checklist to prepare before you install.

- Choose the Manager machine and the storage machine. Confirm that neither is domain-joined and neither is a domain controller.
- Assign fixed IP addresses to both machines.
- Confirm that every machine meets the System Requirements, including the Windows Server Backup feature on each domain controller.
- Run `cdr repository estimate` and provision a storage volume of at least the recommended size. See Sizing Backup Storage. Plan to turn on BitLocker for it. See Encrypting the vault volume.
- Open the network paths listed in Ports and Protocols.
- Choose an enrollment token: a long random secret that you will give to the Manager installer and to every agent installer. See Certificates and Trust.
- Decide how you will install the DC agent: from the console, with `cdr agent deploy`, or with your own software distribution tools. Prepare the accounts and target settings described in Accounts and Permissions.

- Collect Windows Server installation media (ISO files) that match the build, edition, installation type, and architecture of your domain controllers. The storage agent builds recovery images from them.
- Decide where you will store, offline, the secrets created during setup: the break-glass account password, the recovery passphrase, the enrollment token, and the storage volume's BitLocker recovery key. See Manager Day-One Checklist.
- If you want backups verified by test restores, choose a machine for the Isolated Recovery Host that meets the Isolated Recovery Host (optional) requirements, and a private network prefix for its recovery network that is not used anywhere else.
- Plan who needs operator accounts and which roles they need. See Managing Operators and Roles.

Install the Manager and sign in to the console for the first time, then install the storage agent, and then the DC agents. If you use an Isolated Recovery Host, set it up last, once backups and recovery images exist. For the steps, see Installing and Setting Up.

In this chapter

- System Requirements
- Ports and Protocols
- Accounts and Permissions
- Certificates and Trust
- Choosing Which Domain Controllers to Back Up
- Sizing Backup Storage

System Requirements

This topic lists the requirements for each machine in a Cygna Forest Recovery deployment. Requirements that apply only when you recover a server are in Recovery Requirements.

Machines at a glance

Machine	Operating system	Minimum	Recommended
Manager	64-bit Windows Server 2019 or later	2 processor cores, 4 GB of memory	4 processor cores, 8 GB of memory
Storage	64-bit Windows Server	4 processor	4 processor cores,

Machine	Operating system	Minimum	Recommended
machine	2019 or later	cores, 4 GB of memory	8 GB of memory
Isolated Recovery Host (optional)	64-bit Windows Server 2019 or later, with Hyper-V	4 processor cores, 8 GB of memory	8 processor cores, 32 GB of memory, on a dedicated machine
Candidate machines (optional)	64-bit Windows Server 2019 or later	Enough memory to start the recovery image: its size plus about 1 GB	The processors and memory of the domain controller it replaces
Domain controllers	Windows Server 2016 through Windows Server 2025, Server Core or Desktop Experience	—	—

All components

- No .NET runtime is needed. Each component includes everything it needs to run.
- No database server is needed. The Manager uses an embedded database.
- The installers run silently from the command line, with `msiexec /qn`. They have no setup wizard.
- Install the current Windows updates on the Manager, the storage machine, and the Isolated Recovery Host before you install, and keep them current. Their services need them and can fail to start on a system that is behind. The DC agent does not need them, so a domain controller rebuilt from installation media during a recovery can take its agent before it is updated.

Manager machine

Requirement	Details
Operating system	64-bit Windows Server 2019 or later.
Domain membership	Not joined to any domain. Never a domain controller.

Requirement	Details
IP address	A fixed IP address: static, or a DHCP reservation that does not change.
Ports	TCP 5210 (console and REST API) and TCP 5211 (agent connections) must be free. TCP 80 is used for the optional redirect to HTTPS; if another program already uses it, the Manager starts without the redirect. You can choose other ports at installation.
Other software	None. IIS, SQL Server, and a .NET runtime are not required, and neither is a PDF converter: the Manager installs its own PDF renderer.
Disk space	The database, its daily snapshots, and the service logs are kept under `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager`. Backups are not stored on the Manager machine. The program folder includes the PDF renderer and the browser engine it uses, which take more than 250 MB.

Storage machine

Requirement	Details
Operating system	64-bit Windows Server 2019 or later.
Domain membership	Not joined to any domain. Never a domain controller.
IP address	A fixed IP address: static, or a DHCP reservation that does not change. If the machine has more than one address, set the **Advertised host** setting to the address domain controllers should use.
Storage volume	A local folder for the storage root, such as `E:\CygnaDR`, on an NTFS or ReFS volume large enough for your backups. The storage agent creates the `staging`, `vault`, and `manager-backups` folders in it. To work out the size, see Sizing Backup Storage. The storage root must be available when the service starts, or the storage agent does not start. BitLocker on this volume, with auto-unlock, is recommended. See Encrypting the vault volume.
Staging share	An SMB share over the `staging` folder under the storage root,

Requirement	Details
	which you create. Never share the storage root itself, and don't use an administrative share such as `E$`. The storage agent checks the share's permissions and adds what it needs, but does not create the share. Success auditing for Logon, File Share, Detailed File Share, and File System on the storage machine is recommended, so that every connection to the share is seen. See Audit policy on the storage machine.
Inbound SMB	TCP 445 open from your domain controllers and recovery targets, and from the Isolated Recovery Host if you have one. The storage agent installer does not create this firewall rule.
Installation media and recovery images	Folders for your Windows Server installation media (ISO files) and for the recovery images built from them. Recovery images can be built only when a recovery image folder is set.
Service account	The storage agent runs as LocalSystem and needs local administrator rights. Don't change it to another account.

Domain controllers

Requirement	Details
Operating system	Windows Server 2016 through Windows Server 2025, Server Core or Desktop Experience.
SYSVOL replication	SYSVOL must replicate with DFS Replication (DFSR) in every domain you protect. A domain whose SYSVOL still replicates with FRS cannot be recovered: its coverage shows **FRS SYSVOL — unsupported**, and preflight refuses a restore of its backups. To check a domain, run `dfsrmig /getglobalstate` on one of its domain controllers; a domain on DFSR reports **Eliminated**. To move a domain from FRS to DFSR, follow Microsoft's Migrate SYSVOL replication from FRS to DFS Replication.
Windows Server Backup	The Windows Server Backup feature must be installed. See Windows Server Backup later in this topic.
Network	Outbound access to the Manager on TCP 5211 and to the storage machine on TCP 445.
Free disk space	For a DC state recovery, free space on the domain controller of

Requirement	Details
	about 1.15 times the size of the backup being restored. For details, see Recovery Requirements.
Remote installation (optional)	To install the DC agent from the console or with `cdr agent deploy`, the target needs the `ADMIN$` share, the Remote Registry service, and either WinRM or DCOM. See Accounts and Permissions.

Candidate machines

A candidate machine receives a recovered domain controller when the original cannot start. It must meet these requirements before a recovery can use it:

Requirement	Details
Operating system	64-bit Windows Server 2019 or later, able to start normally.
DC agent	The DC agent installed and enrolled with the Manager.
Role	Not a domain controller.
Hardware	At least as many disks as the backed-up domain controller, each the same size or larger, and the same firmware type (UEFI or BIOS). Enough memory and free space to start the recovery image. With Secure Boot on, firmware that trusts Microsoft's *Windows UEFI CA 2023* certificate, which a domain controller with current updates boots with. A virtual machine gets this from a Hyper-V host on current Windows updates.
Network	The same access as a domain controller: the Manager on TCP 5211 and the storage machine on TCP 445.

The full hardware requirements, and the conditions that must hold at the time of recovery, are in Recovery Requirements.

Isolated Recovery Host (optional)

An Isolated Recovery Host is a Hyper-V host on which Cygna Forest Recovery runs test restores of your backups. You need one only if you want backups verified by test restores. A dedicated machine is recommended.

Requirement	Details
Operating system	64-bit Windows Server 2019 or later, kept on current Windows updates. A host that is behind on updates can fail to start a restored domain controller that is more current than the host, so its test restores fail. See Keeping the host current.
Processor and memory	See *Machines at a glance* at the start of this topic. Each test restore virtual machine is given the memory its backup recorded, reduced to what the host has free, and the host warns when it has less than 8 GB.
Virtualization	Hyper-V. The agent installs it, with its PowerShell module, when it is missing. On a physical machine, processor virtualization (Intel VT-x or AMD-V) must be turned on in the firmware. On a virtual machine, nested virtualization must be turned on.
Disk space	Room on an NTFS or ReFS drive for the disks of each domain controller you test, at the sizes its backup recorded. Each test restore checks the free space before it starts. The host warns when less than 100 GB is free where the virtual machines go. A data drive separate from the system drive is recommended.
Domain membership	Not joined to any domain. Never a domain controller, and never on the same machine as one.
Other roles	A machine of its own is recommended. The storage agent or the Manager may run on the same machine; this is supported, with the costs described in Planning Your Deployment.
Network	On the same network as the Manager and the storage machine. Outbound access to the Manager on TCP 5211 and to the storage machine on TCP 445. DNS servers configured on the host, which test restores use.
Recovery network	A private IPv4 network, from a /16 to a /29, that is used nowhere the host needs to reach. The agent creates it behind Windows NAT on the host. The default is `172.31.250.0/24`.

Operator workstation

Requirement	Details
Web browser	A current web browser.
Network	Access to the Manager on TCP 5210, and optionally TCP 80 for the redirect to HTTPS.
Certificate trust	The Manager's certificate authority trusted on the workstation, so the browser does not warn about the console's certificate. See Trusting the Manager Certificate.
Command-line tool	`cdr.exe` runs on Windows. It is installed on the Manager machine; to use it elsewhere, copy it from the Manager's program folder.

Windows Server Backup

Cygna Forest Recovery backs up domain controllers with Windows Server Backup (`wbadmin`). The Windows Server Backup feature is not installed on Windows Server by default, so a backup fails on any domain controller that does not have it.

You can install the feature in either of these ways:

- Install it yourself on each domain controller, for example with the PowerShell command `Install-WindowsFeature Windows-Server-Backup`, or through your usual configuration management tools.
- Let the DC agent install it when it is missing. The **Install the Windows Server Backup feature** setting on the **Domain controller agents** tab of the **Configuration** screen does this, and is on by default. Where adding a Windows feature to a domain controller must go through change control, turn it off and install the feature yourself. See Settings Reference.

Ports and Protocols

This topic lists the network connections Cygna Forest Recovery uses. Each row names the side that opens the connection; replies on the same connection need no separate rule. Agents only open outbound connections, so they need no inbound firewall rules.

Connections

From	To	Port and protocol	Purpose
Operator browser, `cdr`	Manager	TCP 5210, HTTPS	Console and REST API.
Operator browser	Manager	TCP 80, HTTP (optional)	Redirects the browser to the console on HTTPS. Serves nothing else. You can turn it off at installation.
DC agent, storage agent, Isolated Recovery Host agent, recovery agent	Manager	TCP 5211, HTTPS over HTTP/2 with mutual TLS	Agent enrollment and agent sessions, which carry jobs, recovery steps, configuration, and status.
Domain controller, recovery environment, Isolated Recovery Host	Storage machine	TCP 445, SMB	Writing backups to staging, and reading a backup during a restore. A test restore reads its backup through the Isolated Recovery Host, so the connection comes from the host's address.
DC agent	Domain controllers	TCP and UDP 389, LDAP; TCP 3268, global catalog	Reading the directory topology.
Manager or `cdr agent deploy`	Target domain controller	TCP 445, SMB (`ADMIN$` share and Remote Registry); and TCP 5985, WinRM, or TCP 135 and dynamic RPC ports, DCOM	Installing or updating the DC agent remotely (optional).
`cdr repository estimate`	Domain controllers	TCP 5985, WinRM, or TCP	Measuring domain controller volumes to size

From	To	Port and protocol	Purpose
		135 and dynamic RPC ports, DCOM	backup storage (optional).
Manager	Mail server	TCP 587, SMTP with STARTTLS (default); TCP 465, SMTP over TLS; or TCP 25, unencrypted relay. The port is the **SMTP port** setting.	Sending recovery plan notifications and the recovery report by email (optional).

The Manager ports 5210 and 5211 are defaults. You can change them, and the redirect port, when you install the Manager. For details, see Installing the Manager.

Plain `http://` addresses are refused everywhere except the redirect port. Agents and `cdr` must use `https://`.

Firewall rules created by the Manager installer

The Manager installer adds these inbound rules to Windows Defender Firewall on the Manager machine. Each rule allows TCP traffic to `CygnaDR.Manager.exe` on its port, on every network profile.

Rule name	Port	Profile
Cygna DR Manager (agent gRPC)	5211	All
Cygna DR Manager (console)	5210	All
Cygna DR Manager (console redirect)	80	All

If you install with different ports, the rules use those ports. The redirect rule is not created when the redirect is turned off. Uninstalling the Manager removes all of these rules.

Notes

- **Network profiles**—The Manager's rules apply on every profile because Windows classifies the network of a machine outside a domain as Public until someone changes it. The rules are not limited by source address. What protects the ports is the Manager itself: the console and API require a signed-in operator, and the agent port admits only

agents holding a certificate from the Manager's CA. To narrow the rules further, add a remote address scope to them or replace them with your own.

- **SMB on the storage machine**–The storage agent installer does not create any firewall rule. Allow inbound TCP 445 on the storage machine from your domain controllers and recovery targets, and from the Isolated Recovery Host if you have one, and limit it to those networks.
- **Isolated Recovery Host**–The host agent installer creates no firewall rules, and the host needs no inbound connections. A test restore virtual machine reaches the Manager and the storage machine through the host's NAT while it restores. After the restore, the restored system runs on a private switch with no network at all and reports only through Hyper-V, so it opens no connections. See Test Restores.
- **Remote installation**–For WinRM, when the machine that runs the installation is not in the target's domain, as is always the case for the Manager, the target must be in that machine's WinRM TrustedHosts list. For DCOM, the target must allow the **Windows Management Instrumentation (WMI-In)** firewall rule group. See Accounts and Permissions.
- **Listening address**–By default, the Manager listens on all of its network interfaces. On a machine with several interfaces, you can bind it to one at installation.
- **During a forest or domain recovery**–Restore targets are cut off from production, and only the paths a recovery needs stay open to the Manager and the storage machine. See Isolating Recovery Targets.

Accounts and Permissions

Cygna Forest Recovery needs no domain service account, and it stores no domain credentials. When a task needs domain rights, such as installing an agent remotely or seizing operations master roles during a recovery, you supply the credential for that task only. This topic describes every account involved, so you can prepare them before you install.

Service accounts

Service	Runs as	Notes
`CygnaDRManager`	LocalSystem	On the Manager machine.
`CygnaDRStorageAgent`	LocalSystem	Needs local administrator rights on the storage machine, because it creates and deletes local accounts and sets permissions on the vault. Don't

Service	Runs as	Notes
		change it to another account.
`CygnaDRDcAgent`	LocalSystem	On each domain controller and candidate machine.
`CygnaDRHostAgent`	LocalSystem	On the Isolated Recovery Host, if you have one. LocalSystem gives it the rights to manage Hyper-V on the host. No credential is stored for the host: the agent's authority to the Manager is its enrollment certificate, like every other agent, and the Manager never connects to the host.

The installers configure these accounts. You don't need to create anything.

Storage machine local groups

The storage agent creates these local groups on the storage machine. Access to staging and the vault is granted through the groups, so you never need to edit the permissions on those folders.

Group	Purpose
CygnaDR Administrators	Has write access to the vault, the staging folder, and the folder that holds the Manager's configuration backups. Add the local accounts of anyone who must be able to manage the backup files directly on the storage machine. This group must have at least one member before you can turn on the **Remove BUILTIN\Administrators from the vault** setting.
CygnaDR Staging Writers	Holds the local accounts that the storage agent creates for backup jobs, and nothing else. The staging share grants this group access. Membership alone grants no access to any backup.
CygnaDR Vault Readers	Holds the local accounts that the storage agent creates for restores, and nothing else. Membership alone grants no access to any backup.

Accounts created for each job

For each backup job, the storage agent creates a local account named `cdr-` followed by 12 hexadecimal characters, with write access to that job's staging folder only. For each restore, it creates a read-only account named `cdr-r-` followed by 12 hexadecimal characters. Each account is deleted when its job or restore ends. You may see these accounts in the storage machine's local users while a job is running; don't change or delete them.

Audit policy on the storage machine

The storage agent watches the staging share for connections that no backup job explains, and the vault for changes that retention did not make. It reads the storage machine's audit policy to know what the Security log can tell it, and it never changes that policy. Without the auditing below, both watches still run, but they see less: short connections to the staging share can be missed, and a change to a backup is reported without saying who made it. The Manager raises **Storage machines not fully watching the staging share** while the auditing the share watch needs is off.

For complete coverage, turn on success auditing for these subcategories in the storage machine's Advanced Audit Policy, in its local security policy or with `auditpol`:

Subcategory (success)	What it adds
Logon/Logoff > **Audit Logon**	Event 4624: every logon of a job's account, with its client address, however briefly its session lasts.
Object Access > **Audit File Share**	Event 5140: every connection to the staging share by any account.
Object Access > **Audit Detailed File Share**	Event 5145: the staging folder reached through another share, such as an administrative share, and the client address of a change made to the vault over SMB. This subcategory logs every access to every share on the machine, so size the Security log for it.
Object Access > **Audit File System**	Events 4663 and 4660: the account and process that changed a committed backup. The storage agent adds the audit entry on each committed backup's folder that these events need; it audits nothing else on the machine.

Turning on the auditing

The storage machine is not joined to a domain, so set this in its local policy. You can use a command prompt or Local Security Policy.

To turn on the auditing from a command prompt:

1. On the storage machine, open a command prompt as an administrator.
2. Turn on success auditing for the four subcategories. The subcategories are given by their GUIDs, which work in every Windows language:

   ```
   auditpol /set /subcategory:"{0CCE9215-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE9224-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE9244-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE921D-69AE-11D9-BED3-505054503030}" /success:enable
   ```

   In order, these are Logon, File Share, Detailed File Share, and File System.
3. Check the result:

   ```
   auditpol /get /subcategory:"{0CCE9215-69AE-11D9-BED3-505054503030},{0CCE9224-69AE-11D9-BED3-505054503030},{0CCE9244-69AE-11D9-BED3-505054503030},{0CCE921D-69AE-11D9-BED3-505054503030}"
   ```

   Each of the four lines should show **Success** (or **Success and Failure**).

To turn on the auditing in Local Security Policy:

1. On the storage machine, run `secpol.msc`.
2. Go to **Security Settings** > **Advanced Audit Policy Configuration** > **System Audit Policies - Local Group Policy Object**.
3. Under **Logon/Logoff**, open **Audit Logon**, select **Configure the following audit events** and **Success**, and select **OK**.
4. Under **Object Access**, do the same for **Audit File Share**, **Audit Detailed File Share**, and **Audit File System**.

If the machine also has basic audit policies set under **Local Policies** > **Audit Policy**, check that **Local Policies** > **Security Options** > **Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings** is **Enabled**, which is the default. Otherwise Windows ignores the subcategory settings.

You do not need to add audit entries to any folder: the storage agent adds the one that File System auditing needs on each committed backup. Detailed File Share logs every access to every share on the machine, so increase the maximum size of the Security log to keep enough history.

The storage agent reads the audit policy every 10 minutes, so **Storage machines not fully watching the staging share** clears without a restart once it sees the change, unless the alert gives another reason. To have it read the policy at once, restart the storage agent service, `CygnaDRStorageAgent`, while no backup is running.

For what the storage agent watches, see Vault Protection and Integrity.

Account for installing DC agents remotely

When you install or update the DC agent from the console or with `cdr agent deploy`, you supply an account for that run. The password is used only for the run and is never stored.

- The account must be a local administrator on each target domain controller.
- Enter it as `DOMAIN\user` or `user@domain`.
- To install from the console, your operator account needs the System Admin role.

Each target must have:

- The `ADMIN$` administrative share, which is enabled by default.
- The Remote Registry service, which Windows Server starts on demand by default.
- WinRM, which is enabled by default on Windows Server, or, if you use DCOM, the **Windows Management Instrumentation (WMI-In)** firewall rule group enabled.

WinRM also needs a setting on the machine that runs the installation. When that machine is not in the target's domain, the target must be in its WinRM TrustedHosts list. Console installations run from the Manager, which is never in the domain, so for WinRM add your domain controllers to the Manager machine's TrustedHosts list. DCOM needs no such setting. You choose the transport with the **Remote management transport** setting on the **Manager** tab of the **Configuration** screen, or with the `--transport` option of `cdr agent deploy`.

The enrollment token is also needed for a new agent. A console installation uses the Manager's own token; `cdr agent deploy` asks you for it. You can also install agents manually or with your own software distribution tools, which needs no remote access. For the steps, see Deploying DC Agents.

Credentials needed during a recovery

Some recovery steps need directory credentials. You supply them when the recovery asks for them; they are never stored.

Credential	When it is needed
An Enterprise Admins or Domain Admins account	Seizing operations master roles and promoting rebuilt domain controllers.
An account in both Enterprise Admins and Schema Admins	Recovering the forest root domain.
The password of each domain's built-in Administrator account, as it was at the time of the backup	Signing in to a restored domain controller, including in Directory Services Restore Mode.

Record where these credentials are kept as part of your recovery planning. The built-in Administrator password in particular must be the one that was valid when the backup was taken, not the current one. For details, see Recovery Requirements.

Accounts for sizing and directory cleanup

- **`cdr repository estimate`**—An account that is an administrator on the domain controllers. Give the domain by its DNS name, as `user@corp.example.com` or `corp.example.com\user`; the NetBIOS form, such as `CORP\user`, is refused because the command authenticates only with Kerberos. If you omit the account, the command uses your own sign-in on a domain-joined machine. See Sizing Backup Storage.
- **`cdr cleanup`**—An account with domain administrator rights, to remove the directory data of a domain controller, domain, or trust that no longer exists. See Directory Cleanup Commands.

`cdr` never accepts a password as a command-line argument. It reads passwords from an environment variable or asks for them at a prompt, so they don't appear in your command history.

Cygna Forest Recovery operator accounts

People sign in to the console and `cdr` with Cygna Forest Recovery operator accounts. These accounts are kept in the Manager's database. They are not Windows or Active Directory accounts, so you can sign in while the directory is down. Each operator can also turn on multi-factor authentication, with a code from an authenticator app; it needs no internet access on the Manager or the phone. See Multi-Factor Authentication.

The first account, the break-glass account, is created when you first sign in to the console after installing the Manager. It holds every permission. You then create named accounts for each person and assign them roles: Viewer / Auditor, Backup Operator, Restore Operator, or System Admin. Approval and execution of a recovery are separate permissions, which among the roles only Restore Operator grants, and the System Admin role has no backup or restore permissions.

For details, see Signing In for the First Time and Managing Operators and Roles.

Certificates and Trust

Every connection to the Manager is encrypted with TLS, and every certificate involved is issued by the Manager's own certificate authority (CA). Cygna Forest Recovery does not use a domain certificate service or a public CA, so secure connections keep working when the directory is down. This topic explains the certificates, the two values you handle when installing agents—the CA fingerprint and the enrollment token—and how browsers and `cdr` come to trust the Manager.

The Manager CA

The Manager creates its CA the first time its service starts. The installer creates no certificates. The CA uses an ECDSA P-256 key and is valid for 20 years. Its private key is kept in the Manager's database, protected for that machine. When you set the recovery passphrase at first sign-in, a copy of the key is protected with that passphrase so that the Manager can be restored on another machine. For more information, see Protecting the Manager.

The Manager uses the CA to issue:

- **Its own server certificate**–Valid for 365 days and renewed automatically. It covers the Manager's host name, its fully qualified name, `localhost`, and every IP address of the machine. If the machine's names or addresses change, the Manager issues a new certificate when it starts.
- **A client certificate for each agent**–Issued when the agent enrolls, valid for 365 days, and renewed automatically when two-thirds of its lifetime has passed.

Connections use TLS 1.2 or TLS 1.3. Plain HTTP is refused, except on the optional redirect port, which only sends browsers to the HTTPS address.

CA fingerprint

The CA fingerprint is the SHA-256 hash of the Manager's CA certificate. It is shown as 16 groups of four hexadecimal characters separated by hyphens. Agents use it to make sure they are talking to your Manager and not to an impostor: an agent connects only to a Manager whose CA matches the fingerprint it was installed with.

The fingerprint is public, not secret. It identifies a certificate that the Manager presents to anyone who connects, so you can copy, email, or print it freely. You can find it in these places:

- In the Windows Application event log on the Manager machine, every time the Manager service starts, in a line that begins `Cygna DR manager CA fingerprint:`. This is the most reliable source, because you read it on the Manager machine itself. The same line appears in the current `manager-yyyyMMdd.log` file in `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\logs`.
- On the **Agents** screen of the console, in the installation command for each agent, including before any agent has enrolled. The **Manager recovery** card on the **License** screen also shows it.
- In the output of `cdr trust list`, after `cdr` has connected to the Manager.

Before you rely on a fingerprint, compare it from two of these sources. A manual agent installation requires it as the `MANAGERCAFINGERPRINT` property. When you install agents from the console or with `cdr agent deploy`, the fingerprint is supplied automatically.

The fingerprint changes only if you replace the Manager CA. For what that involves, see Re-Keying the Manager CA.

Enrollment token

The enrollment token is a secret that you choose and give to the Manager installer as the `ENROLLMENTTOKEN` property. Every agent presents the same token the first time it connects. An agent with a valid token is enrolled immediately, with no approval step; the enrollment is recorded in the audit trail.



Note: Important: Treat the enrollment token like a password. Anyone who has it and can reach the Manager can enroll an agent. Windows Installer masks it in installer logs, including a verbose log (`/l*v`), but not on the `msiexec` command line. Where process-creation auditing records command lines (event 4688), the token is in that event: install DC agents from the console or with `cdr agent deploy`, which put no token on a command line, and keep commands that contain it out of scripts.

- The console never shows the token.
- Each installation keeps its token in the registry, readable only by SYSTEM and Administrators, so an agent can enroll again if its certificate expires or the Manager no longer recognizes it. Upgrades keep the stored token.
- Store a copy offline with your other recovery secrets. You need it to rebuild the Manager on a new machine without re-enrolling every agent. See Manager Day-One Checklist.

Why IP addressing works

Agents check the Manager's certificate against the pinned CA and against a marker that only the Manager's certificate carries. They don't compare it with the address they connected to. This means:

- Agents can reach the Manager by IP address, without DNS.
- A stolen agent certificate cannot be used to impersonate the Manager, because it lacks the Manager marker.
- A Manager restored onto a machine with a different address is still trusted by every agent, with nothing reissued.

The Manager identifies each agent by its client certificate in the same way, without checking its address.

Browser trust

The console uses a certificate issued by the Manager CA. Browsers don't know this CA, so they show a certificate warning until you trust the Manager CA on the operator's

workstation. Trust it on each workstation used to operate Cygna Forest Recovery, and confirm the fingerprint when you do. For the steps, see Trusting the Manager Certificate.

Because the server certificate covers every address of the Manager, a browser accepts it whether you open the console by IP address or by name, once the CA is trusted.

How `cdr` trusts the Manager

`cdr` checks the Manager's certificate the same way agents do: by CA fingerprint, not by address. The first time you connect to a Manager from an interactive prompt, `cdr` shows the fingerprint and asks you to confirm it against the console or the event log. If you confirm, it remembers the fingerprint for that Manager address and your Windows user, and checks it on every later connection.

A script or scheduled task has no prompt to answer, so it must supply the fingerprint explicitly with the `--manager-ca` option or the `CYGNADR_MANAGER_CA` environment variable. If the fingerprint is missing or does not match, `cdr` refuses to connect. For details, see Connecting and Signing In.

Choosing Which Domain Controllers to Back Up

You install the DC agent on every domain controller, but you don't need to back up every one. A recovery restores one writable domain controller in each domain and rebuilds the others by promoting them again. This topic explains which domain controllers to back up, what you give up for the ones you don't, and how to keep the dashboard accurate.

Why not back up every domain controller

Every backup is a full copy, so storage grows with each domain controller you back up and each version you keep. In a large forest, backing up every domain controller multiplies the storage you need many times over, for backups a recovery would never use.

What Microsoft recommends

Microsoft's Active Directory Forest Recovery Guide advises you to "Back up at least two writeable DCs for each domain regularly," so that you have more than one backup to choose from. It also recommends a dedicated domain controller in each domain that is the preferred one to restore. When several backups are available, you restore only one of them. See Determine how to recover the forest (Microsoft Learn, updated September 11, 2025).

A recommended backup set

For each domain, back up:

- **At least two writable domain controllers**–If one backup turns out to be unusable or compromised, you have another to restore from.
- **The PDC emulator**–Its backup is typically the best source for SYSVOL.
- **One dedicated restore domain controller**–Choose one domain controller as the preferred source for a restore, and keep its backups current.
- **Full system backups**–A full system backup supports every recovery method, including a restore onto different hardware. See Backup Types.

Where you can, follow Microsoft's other criteria for the domain controllers you choose: prefer ones that are DNS servers, so that name resolution comes back with the directory, and ones that are global catalog servers.

Never rely on backups of read-only domain controllers (RODCs). An RODC backup cannot restore a writable domain controller, so a recovery always starts from a writable one, and RODCs are rebuilt.

What a backup of a domain controller gives you

Domain controller	What you can do
Backed up	Restore it with DC state recovery. Recover the whole server in place or onto a candidate machine. Use it as the source for a domain or forest recovery.
Not backed up	Rebuild it by promotion during a domain or forest recovery. It cannot be restored by DC state recovery or full system recovery, because both need that domain controller's own backup.

For the recovery methods, see Recovery Scopes.

A sizing example

Consider a forest of 10 domains in which you back up two domain controllers per domain, each full system backup is about 50 GB, and the vault keeps 2 versions of each, the product default:

- **Retained backups**–10 domains × 2 domain controllers × 50 GB × 2 versions = about 2 TB.
- **Backups in progress**–4 backups running at once, the default, at 50 GB each plus the 20% margin = about 240 GB.
- **Free-space reserve**–10 GB.

That gives a required size of about 2.25 TB, and a recommended size, with the 25% growth allowance, of about 2.8 TB. Every domain controller you add to the backup set adds its full size for each version kept. For how the figures are calculated, see Sizing Backup Storage.

Estimating your own backup set

To size a backup set of a fixed number of domain controllers per domain, run `cdr repository estimate` with `--per-domain`. Each domain is sized as that many domain controllers at the average backup size of the domain's domain controllers. A domain with fewer domain controllers backs up all it has. For example, for two domain controllers per domain:

```
cdr repository estimate --per-domain 2
```

`--per-domain` cannot be combined with `--dc`. For every option, see Storage and Media Commands.

How the console treats the domain controllers you do not back up

A domain controller with an enabled backup policy is a backup target. One with no enabled policy and nothing in the vault has the **Rebuilt on recovery** verdict: it is information, not a finding, and raises no alert. You do not need to exclude it. The counts show it beside the protected domain controllers, for example `2 protected · 4 rebuilt on recovery`, not as a shortfall.

Whether a domain can be recovered is decided by the domain's own verdict, which turns critical when no writable domain controller has a restorable backup, whatever the other domain controllers read.

A global catalog in a forest with more than one domain needs the DC agent even when you do not back it up. A recovery of another domain in the forest reconciles each global catalog through its agent, and the console raises **Global catalogs with no agent** until each has one.

Keep exclusions for objects that are really out of scope, such as a forest reached across a trust, not for domain controllers that recovery rebuilds.

For the full rules, see Backup Targets and Rebuilt Domain Controllers. For how verdicts are decided, see Coverage and Readiness.

Sizing Backup Storage

The storage machine's volume holds every backup that retention keeps, plus room for the backups that are running. This topic explains what drives the size and how to measure it from your own domain controllers with `cdr repository estimate`.

What drives the size

- **Every backup is a full copy**—Backups are not incremental. A full system backup copies the used space on each domain controller's critical volumes: the volumes that hold Windows, the boot files, the directory database and its logs, SYSVOL, and the files of installed services.
- **How many domain controllers you back up**—Each one you back up adds its full backup size for every version kept. You don't have to back up every domain controller; a recovery can restore some and rebuild the rest. For which ones to choose, see Choosing Which Domain Controllers to Back Up, and size that set with `--per-domain`.
- **How many versions you keep**—Retention settings decide how many backups of each domain controller stay in the vault. See Retention.
- **Backups in progress**—Staging and the vault share one volume, so backups being written take space alongside the stored history. The storage agent reserves room for each backup before it starts.
- **Free space reserve**—The storage agent keeps a minimum amount of free space that no backup may use.
- **The Manager's configuration backups**—The storage machine also keeps the Manager's daily configuration backups, on the same volume, beside the vault. Each is roughly the size of the Manager's database, compressed, and the storage machine caps the folder with its own size limit.

DC state backups are smaller than full system backups, but they support only DC state recovery. The estimate assumes full system backups, which support every recovery method. For the difference, see Backup Types.

How the estimate works

`cdr repository estimate` reads the list of domain controllers from the directory, then connects to each one and reads its volumes. It changes nothing on any machine and does not contact the Manager, so you can run it before you deploy any agents.

For each domain controller, it calculates the backup size as the used space on the critical volumes, minus the files that Windows Server Backup leaves out (the paging file, the hibernation file, and shadow copy storage), plus an allowance for backup image overhead (15% by default).

It then calculates the volume size:

- **Retained backups**—The sum of every domain controller's backup size, multiplied by the number of versions kept for each.
- **Backups in progress**—The largest domain controllers' backups, as many as can run at once, plus the margin the storage agent adds when it reserves room for a backup.

- **Free-space reserve**–The storage agent's reserve.
- **Required**–The total of the three. This is the smallest volume that can accept the next backup when the stored history is at full depth.
- **Recommended**–Retained backups and backups in progress increased by a growth allowance (25% by default), plus the reserve.

Unless you override them, the number of versions, the number of concurrent backups, the margin, and the reserve come from the product's default settings. The output always states the assumptions it used. For the defaults, see Settings Reference.

Requirements for running the estimate

- **The `cdr` tool on Windows**–`cdr.exe` is installed in the Manager's program folder. It runs without installation, so you can copy it to the machine where you run the estimate.
- **A domain-joined machine**–Run the command from a Windows machine joined to a domain in the forest. You can run it from a machine outside the domain if you give an account with `--user` and a domain controller with `--server`, and the machine's DNS resolves the domain.
- **An administrator account on the domain controllers**–By default, the command uses your own sign-in. To use another account, give it with `--user`, naming the domain by its DNS name, such as `admin@corp.example.com` or `corp.example.com\admin`. The password is read from the `CYGNADR_ESTIMATE_PASSWORD` environment variable, or you are prompted for it.
- **Kerberos**–The command authenticates only with Kerberos. The machine's clock must be within five minutes of the domain's, and its DNS must resolve the domain's own zone.
- **Remote access to the domain controllers**–WinRM on TCP 5985 (the default), or DCOM on TCP 135 and the dynamic RPC ports with `--transport dcom`.

When you name an account with `--user` and list domain controllers with `--dc`, the command tries the account on one domain controller first. If that domain controller refuses it, the command stops instead of sending a possibly wrong password to every domain controller and locking the account out.

Running the estimate

To estimate backup storage:

1. On a domain-joined Windows machine, open a command prompt in the folder that contains `cdr.exe`.
2. Run the estimate for the whole forest:

```
cdr repository estimate
```

The command lists each domain controller as it is measured, then prints the results.

3. Optionally, run it again with your planned settings. For example, to keep 7 daily versions of 3 domain controllers in each domain, with an administrator account in the forest root domain:

```
cdr repository estimate --versions 7 --per-domain 3 --user admin@corp.example.com
```

Common options

Option	Description
`--dc <FQDN>`	Measures only this domain controller. Repeat for more. Without it, every domain controller in the forest is measured.
`-d`, `--domain <FQDN>`	Measures only the domain controllers of this domain.
`-s`, `--server <DC>`	The domain controller to read the list of domain controllers from.
`-u`, `--user <ACCOUNT>`	An administrator on the domain controllers, with the domain given by its DNS name.
`--versions <N>`	Backups kept per domain controller, as in the **Maximum versions per DC** setting. 0 means no count limit.
`--days <N>`	The age limit for backups in days, as in the **Maximum version age** setting. 0 means no age limit.
`-e`, `--every <MINUTES>`	How often each domain controller is backed up, used with `--days`. The default is 1440 (daily).
`--per-domain <N>`	Backs up N domain controllers in each domain instead of every one, each sized at the domain's average.
`--concurrent <N>`	Backups running at once, as in the **Maximum concurrent backups** setting.
`--overhead <PERCENT>`	Image overhead added to the copied space. The default is 15.
`--growth <PERCENT>`	Growth allowance in the recommended size. The default is 25.
`--volumes`	Lists every volume of every domain controller, with what was

Option	Description
	left out and why.
`--json`	Writes the results as JSON instead of tables.

If you give both `--versions` and `--days`, the stricter limit applies, as it does in the vault. For the complete list of options, see Storage and Media Commands.

Reading the output

The first table has one row per domain controller:

Column	Shows
DC	The domain controller's name. Read-only domain controllers are tagged `RODC`.
DOMAIN, SITE, OS	Where the domain controller is and what it runs.
CRITICAL VOLUMES	The volumes a full system backup includes.
USED	The used space on those volumes.
LEFT OUT	Space that Windows Server Backup does not copy: the paging file, the hibernation file, and shadow copy storage.
BACKUP	The estimated size of one full backup. A domain controller that could not be read shows `not measured` and the reason.

Below the table, a line starting with `Assumes` states the retention, concurrency, margin, reserve, and overhead used. The summary table then shows **Retained backups**, **Backups in progress**, **Free-space reserve**, **Required**, and **Recommended**. Provision at least the recommended size.

Pay attention to the warnings after the summary:

- **Domain controllers that were not measured**—They are not in the main totals. An extra summary row shows the recommended size if each one is as large as the largest domain controller that was measured. Fix the cause and run the estimate again, or use that row.
- **`NOT IN ANY TOTAL`**—Domains where no domain controller could be reached. This is usually a domain that no longer exists but is still defined in the directory. If the domain is live, fix access and run the estimate again.

- **`NOT MEASURED`**—Domains whose domain controllers answered but could not be read, usually because the account is not an administrator there. Run the estimate again with `--user` and an account that is.

The command exits with code 7 if any domain controller was not measured, and 0 if all were.



Note: The estimate does not include your Windows Server installation media or the recovery images. If you keep them on the same volume, allow a few gigabytes for each Windows Server version.

After you deploy

Once backups are running, the **Repository** screen shows the vault's real capacity, free space, and retention depth for each domain controller, and `cdr repository status` shows the same from the command line. See Repository.

Installing and Setting Up

A Cygna Forest Recovery deployment has three installed components: the Manager, the storage agent on the storage machine, and a DC agent on each domain controller. An optional fourth, the Isolated Recovery Host agent, runs test restores of your backups on a Hyper-V host. All of them install from silent MSI packages with no setup wizard. After installation you finish the setup in the console: you claim the Manager, apply the license key, connect storage, provide installation media for full system recovery, and create backup policies.

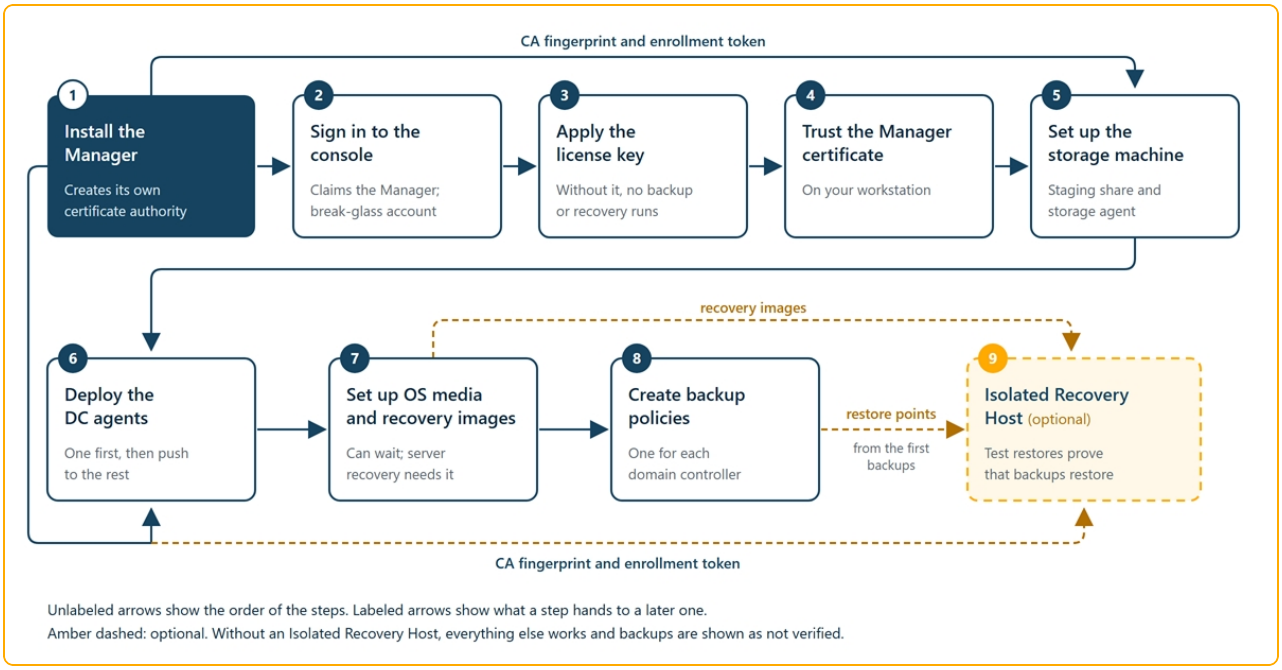
This chapter assumes you have planned the deployment. For requirements, ports, and accounts, see Planning Your Deployment.

Order of operations

Follow this order. Each step depends on something the step before it creates.

1. **Install the Manager.** The Manager creates its own certificate authority (CA) the first time its service starts. Every agent is installed with that CA's fingerprint and with the Manager's enrollment token, so you cannot install an agent until the Manager is running.
2. **Sign in to the console straight away.** A new Manager has no operator accounts. The first person to open the console completes the one-time setup and receives the break-glass account, which holds every capability. Do this immediately after installing, so that nobody else can claim the Manager.
3. **Apply the license key.** A new Manager has no license, and without one it refuses every backup and recovery operation. Apply the key Cygna sent you on the **License** screen as soon as you have signed in.
4. **Trust the Manager certificate on your workstation.** The console uses a certificate issued by the Manager's own CA, so browsers warn until you trust that CA.
5. **Set up the storage machine.** Backups need somewhere to go. Prepare the storage volume and the staging share, and then install the storage agent.
6. **Deploy the DC agents.** Install the agent on one domain controller first. It reports every domain controller in its forest, and you can then push the agent to the rest from the console.
7. **Set up OS media and recovery images.** Restoring a whole server needs a recovery image for its operating system, which the storage agent builds from Windows Server ISOs that you supply. DC state recovery does not need one, so you can do this step later, but full system recovery is unavailable until you do.

8. **Create backup policies.** No backup policy exists by default. Nothing is backed up until you create one for each domain controller.
9. **Set up an Isolated Recovery Host (optional).** Test restores need backups to restore and recovery images to start from, so set up the host once both exist. Without a host, everything else works and backups are shown as not verified.



What you need on hand

Gather these before you start:

- **Machines**–A machine for the Manager and a machine for storage, both running 64-bit Windows Server 2019 or later, neither joined to a domain, each with a fixed IP address (static or a reserved DHCP lease). One machine can hold both. A domain controller can never be the storage machine.
- **Installer packages**–`CygnaDR.Manager.msi`, `CygnaDR.StorageAgent.msi`, `CygnaDR.DcAgent.msi`, and, if you use an Isolated Recovery Host, `CygnaDR.HostAgent.msi`. The Manager package carries a copy of each agent package: the DC agent package for pushing from the console, and the others for downloading from the Agents screen and for updating agents. Every package and program file is signed by Cygna Labs.
- **A Hyper-V host (optional)**–A machine for the Isolated Recovery Host, not joined to a domain. A dedicated machine is recommended. For its sizing and other requirements, see Isolated Recovery Host (optional).
- **A license key**–The Cygna Forest Recovery license key that Cygna sent you. See Licensing.

- **An enrollment token**–A long random secret that you generate. The Manager is installed with it and every agent presents it to enroll. See Installing the Manager.
- **IP addresses**–The Manager's address, which agents dial, and the storage machine's address, which domain controllers write backups to. Agents use addresses rather than names so that they keep working when DNS is down.
- **A storage path**–An absolute path on the storage volume for the storage root, for example `E:\CygnaDR`. See Sizing Backup Storage.
- **An administrator account for the domain controllers**–An account that is a local administrator on each domain controller (Domain Admin in most environments), to push the DC agent. It is used for the installation only and never stored.
- **Windows Server ISOs**–Installation media matching each Windows Server build, edition, and installation type your domain controllers run, for full system recovery.
- **A safe place offline**–Somewhere to record the break-glass credential, the recovery passphrase, and the enrollment token. You need these to sign in and to rebuild the Manager when nothing else works. See Manager Day-One Checklist.

In this chapter

- Installing the Manager
- Signing In for the First Time
- Trusting the Manager Certificate
- Setting Up the Storage Machine
- Deploying DC Agents
- Setting Up OS Media and Recovery Images
- Creating Your First Backup Policies
- Setting Up an Isolated Recovery Host
- Upgrading and Uninstalling

Quick Install Guide

This guide lists the machines a Cygna Forest Recovery for Active Directory deployment needs and the steps to install it, with the commands for each step. Cygna Forest Recovery for Active Directory is part of the Cygna Disaster Recovery Suite, so the services and commands it installs carry the suite's short name, Cygna DR, and its installer packages the full name, Cygna Disaster Recovery. It is for administrators who know Windows Server and Active Directory and want to install without reading the background first. Each step names the topic with the full procedure, the options, and the reasons behind each choice.

For the decisions to make before you start, see Planning Your Deployment and Pre-Deployment Checklist.

Machines

Machine	Operating system	Minimum	Recommended
Manager	64-bit Windows Server 2019 or later	2 processor cores, 4 GB of memory	4 processor cores, 8 GB of memory
Storage machine	64-bit Windows Server 2019 or later	4 processor cores, 4 GB of memory	4 processor cores, 8 GB of memory
Isolated Recovery Host (optional)	64-bit Windows Server 2019 or later, with Hyper-V	4 processor cores, 8 GB of memory	8 processor cores, 32 GB of memory, on a dedicated machine
Candidate machines (optional)	64-bit Windows Server 2019 or later	Enough memory to start the recovery image: its size plus about 1 GB	The processors and memory of the domain controller it replaces
Domain controllers	Windows Server 2016 through Windows Server 2025, Server Core or Desktop Experience	–	–

In addition:

- **Manager and storage machine**–not joined to any domain, never a domain controller, each with a fixed IP address (static, or a DHCP reservation). They can share one machine, but separate machines are recommended. The storage machine needs a volume large enough for the backups you keep, with BitLocker turned on. No IIS, SQL Server, or .NET runtime is needed.
- **Domain controllers**–every domain controller runs the DC agent and needs the Windows Server Backup feature. Back up at least two writable domain controllers in each domain.
- **Isolated Recovery Host**–for test restores only. Not joined to any domain, never a domain controller. Processor virtualization turned on (nested virtualization if it is a

virtual machine), and a private IPv4 prefix that is not used elsewhere, for its recovery network.

- **Candidate machines**–spare machines that can receive a full system recovery. Not a domain controller, with the DC agent installed. At least as many disks as the domain controller they replace, each the same size or larger, and the same firmware type (UEFI or BIOS).
- **Operator workstations**–a current web browser, with the Manager's certificate authority (CA) trusted.

Network

These are the default ports. You can change the Manager's ports when you install it.

From	To	Port
Domain controllers, storage machine, candidate machines, Isolated Recovery Host	Manager	TCP 5211. The Manager installer opens it.
Domain controllers, recovery targets, Isolated Recovery Host	Storage machine	TCP 445 (SMB). You add this firewall rule yourself.
Operator workstations	Manager	TCP 5210 (console, REST API, and `cdr`), and optionally TCP 80 for the redirect to HTTPS.
Manager, or the machine that runs `cdr agent deploy`	Domain controllers, for a push installation	TCP 445 (`ADMIN$`) with the Remote Registry service, plus either WinRM (TCP 5985, with the targets in the installing machine's TrustedHosts) or DCOM (TCP 135 and dynamic RPC ports, with the **Windows Management Instrumentation (WMI-In)** firewall rules enabled).

Have on hand

- **Installer packages**–`CygnaDR.Manager.msi`. The Manager installs a copy of each agent package in the `redist` folder of its program folder, and offers them for download

on the **Agents** screen.

- **The license key**–the key Cygna sent you.
- **IP addresses**–of the Manager and the storage machine. Agents use addresses rather than names, so they keep working when DNS is down.
- **A storage path**–an absolute path on the storage volume, for example `E:\CygnaDR`. It cannot be changed later from the Manager.
- **A domain controller administrator account**–a local administrator on each domain controller (Domain Admin in most environments), used to push the DC agent and never stored.
- **Windows Server ISOs**–matching the build, edition, and installation type of each domain controller, for full system recovery.
- **A safe place offline**–for the enrollment token, the break-glass credential, and the recovery passphrase that you create during installation.

Install

Follow the steps in order. Each step needs something that the step before it creates. Run every command in an elevated session.

Step 1: Install the Manager

1. Generate an enrollment token, a random secret that every agent presents to enroll, and record it offline:

```
$bytes = New-Object byte[] 32
[System.Security.Cryptography.RandomNumberGenerator]::Create().GetBytes($bytes)
([System.BitConverter]::ToString($bytes) -replace '-', '')
```

2. On the Manager machine, install the Manager:

```
msiexec /i CygnaDR.Manager.msi /qn ENROLLMENTTOKEN=<token>
```

3. Read the CA fingerprint from the Application event log. You compare against it in step 3. It is public, so keep a copy with your recovery notes rather than in the safe:

```
Get-WinEvent -LogName Application -MaxEvents 500 |
  Where-Object Message -like '*Cygna DR manager CA fingerprint*' |
  Select-Object -First 1 -ExpandProperty Message
```

For details, see Installing the Manager.

Step 2: Sign in, claim the Manager, and apply the license

Note: Important: Whoever completes the setup page first owns the Manager. Do this step immediately after installing.

1. On the Manager machine, open `https://localhost:5210`.

   The **Set up this manager** page opens.

2. Enter a recovery passphrase of at least 16 characters twice, and select **Create the break-glass account**.

3. Record the break-glass username and password offline with the passphrase, select both confirmations, and select **I have stored both — continue**.

   The password is shown only once.

4. Select **Administration** > **License** > **Apply a license key…**, paste the key, and select **Apply this license**.

   Without a license, the Manager refuses every backup and recovery operation.

5. Select **Administration** > **Operators & Roles**, and create a named account for each person who operates Cygna Forest Recovery. The people setting up the deployment need System Admin and Backup Operator.

6. Sign out, and sign in with your named account. Keep the break-glass account for emergencies.

For details, see Signing In for the First Time.

Step 3: Trust the Manager certificate on workstations

On each operator workstation, download the CA certificate, compare its hash with the fingerprint from step 1, and import it only if they match. In PowerShell 7:

```
$trust = Invoke-RestMethod -Uri https://<manager>:5210/api/trust -SkipCertificateCheck
Set-Content -Path .\manager-ca.pem -Value $trust.caCertificatePem
certutil -decode .\manager-ca.pem .\manager-ca.cer
certutil -hashfile .\manager-ca.cer SHA256
certutil -user -addstore Root .\manager-ca.cer
```

Restart the browser. To trust the CA for every user of a workstation, run `certutil -addstore Root .\manager-ca.cer` in an elevated session instead, or distribute it with Group Policy.

For details, see Trusting the Manager Certificate.

Step 4: Set up the storage machine

1. Turn on BitLocker with auto-unlock for the storage volume.
2. Create the staging folder, share it, and allow SMB from your domain controllers' networks. Share only the staging folder, never the storage root:

   ```
   New-Item -ItemType Directory -Path E:\CygnaDR\staging
   New-SmbShare -Name CygnaDRStaging -Path E:\CygnaDR\staging -FullAccess
   "BUILTIN\Administrators"
   New-NetFirewallRule -DisplayName "Cygna DR staging (SMB)" -Direction Inbound -Protocol
   TCP -LocalPort 445 -RemoteAddress 10.0.1.0/24,10.0.2.0/24 -Action Allow
   ```

3. Copy `CygnaDR.StorageAgent.msi` from `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager\redist` on the Manager machine to the storage machine.
4. Install the storage agent with the Manager's IP address and CA fingerprint, the storage root, and the enrollment token:

   ```
   msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
   MANAGERCAFINGERPRINT=<fingerprint> STORAGEROOT=E:\CygnaDR
   ENROLLMENTTOKEN=<token>
   ```

   On **Administration** > **Agents**, the storage machine shows **Connected**. Within a few minutes, **Protection** > **Repository** shows its capacity.

For details, see Setting Up the Storage Machine.

Step 5: Deploy the DC agents

1. Install Windows Server Backup on each domain controller:

   ```
   Install-WindowsFeature Windows-Server-Backup
   ```

   Alternatively, leave **Install the Windows Server Backup feature** on (the default) on the **Domain controller agents** tab of **Administration** > **Configuration**, and the agent installs it.
2. Install the agent on one domain controller in each forest. Either run the installer on the domain controller, with the command from **Administration** > **Agents** > **Install agent**:

   ```
   msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
   MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
   ```

   Or push it from the Manager machine, in `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager.cdr` prompts for the password and the enrollment token:

```
cdr agent deploy dc01.corp.example.com --user CORP\admin --manager-address
https://10.0.0.10:5211
```

The first agent reports every domain controller in its forest.

3. Select **Active Directory** > **Domain Controllers**, select the domain controllers that have no agent, and select **Deploy the agent to n…**.
4. Enter the administrator account and password, check the Manager address, choose WinRM or DCOM, and select **Install on n machines**.

   Each domain controller shows **Connected** in the **Agent** column.

Candidate machines need the DC agent too. Install it with the same `msiexec` command.

For details, see Deploying DC Agents.

Step 6: Set up OS media for full system recovery

1. On the storage machine, create a folder for ISOs and one for recovery images, outside the storage root, and copy your Windows Server ISOs into the first:

   ```
   New-Item -ItemType Directory -Path E:\os-media, E:\recovery-images
   ```

2. Select **Administration** > **Configuration** > **Storage agents**. Set **Media folder** and **Recovery image folder**, and select **Save**.
3. Restart the storage agent:

   ```
   Restart-Service CygnaDRStorageAgent
   ```

The storage agent builds a recovery image for each operating system that appears in a backup, so images appear after the first backups complete. DC state recovery works without them; full system recovery does not.

For details, see Setting Up OS Media and Recovery Images.

Step 7: Create backup policies

No backup policy exists by default. Back up at least two writable domain controllers in each domain, including the PDC emulator.

1. Select **Active Directory** > **Backup Policies**, and select the domain controllers to back up.
2. Select **Edit *n* schedules…**. Set **State** to **Enabled**, **Interval** to **Daily**, and **Backup type** to **Full system**, and select **Save *n* schedules**.

The first backups are due immediately. To run them outside business hours instead, set **Start time** to **At a set time** and enter a time in the Manager's local time, such as 22:00.

3. Domain controllers you don't back up are shown as **Rebuilt on recovery** and raise no alert; you do not need to exclude them. Make sure every global catalog has the DC agent, because a recovery of another domain in the forest needs it.

For details, see Creating Your First Backup Policies.

Step 8: Set up an Isolated Recovery Host (optional)

1. On the storage machine, allow SMB from the host's address, if you limit TCP 445 to your domain controllers' networks:

```
New-NetFirewallRule -DisplayName "Cygna DR Isolated Recovery Host (SMB)" -Direction Inbound -Protocol TCP -LocalPort 445 -RemoteAddress 10.0.0.30 -Action Allow
```

2. From **Administration** > **Agents** > **Install agent**, download `CygnaDR.HostAgent.msi` to the host, and install it:

```
msiexec /i CygnaDR.HostAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211 MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
```

The agent installs Hyper-V if it is missing and, on a dedicated host, restarts the host five minutes later. It then creates the isolated recovery network.

3. On **Administration** > **Agents**, check that the host panel shows **Ready to run verification**. If it does not, follow the instructions in the panel.

For details, see Setting Up an Isolated Recovery Host.

Verify the deployment

When the first backups have completed, check that:

- **Administration** > **Agents** lists every agent as **Connected**.
- **Protection** > **Restore Points** lists a restore point for each domain controller you back up.
- The **Dashboard** shows each domain and forest as covered.
- On **Active Directory** > **OS Media**, the **Operating systems** tab shows **Available** in the **Recovery image** column for each operating system.
- If you set one up, the Isolated Recovery Host is ready.
- The **Configuration backups** card on the **License** screen shows a backup stored on the storage machine.

- The enrollment token, break-glass credential, and recovery passphrase are recorded together offline, and the CA fingerprint is in your recovery notes.

For everything else to do on the first day, see Manager Day-One Checklist.

Installing the Manager

Install the Manager first. Its certificate authority (CA) is created the first time the service starts, and every agent you install afterwards needs that CA's fingerprint and the enrollment token you give the Manager now.

Before you begin

- Use a machine running 64-bit Windows Server 2019 or later that is not joined to a domain and has a fixed IP address. See System Requirements.
- Make sure TCP ports 5210 and 5211 are free on the machine, and port 80 if you want the HTTP redirect. See Ports and Protocols.
- Sign in to the machine with an account that is a local administrator.
- Plan to complete the first console sign-in as soon as the installation finishes. See Signing In for the First Time.

Choose an enrollment token

The enrollment token is a shared secret. The Manager is installed with it, and each agent presents it to enroll. Any agent that presents a valid token enrolls immediately, with no approval step, so treat the token like a password:

- Make it a long random value, not a word or phrase. Use only letters and digits so that it needs no quoting on a command line.
- Record it offline with the other Manager credentials. You need it for every agent you install by hand, and to bring the Manager back on new hardware.
- Do not put it anywhere other people can read, such as a script on a file share.

For example, this PowerShell command generates a 64-character random hexadecimal token:

```
$bytes = New-Object byte[] 32
[System.Security.Cryptography.RandomNumberGenerator]::Create().GetBytes($bytes)
([System.BitConverter]::ToString($bytes) -replace '-', '')
```

Install the Manager

To install the Manager:

1. Copy `CygnaDR.Manager.msi` to the Manager machine.

   The package is signed by Cygna Labs Corp. To check it, run `Get-AuthenticodeSignature CygnaDR.Manager.msi` in PowerShell and expect the status `Valid` with `O=Cygna Labs Corp.` in the signer, or open the file's **Properties** > **Digital Signatures** tab.

2. Open an elevated command prompt in the folder that holds the package.
3. Run the installer with your enrollment token:

   ```
   msiexec /i CygnaDR.Manager.msi /qn ENROLLMENTTOKEN=<token>
   ```

   The installer runs silently, installs the `CygnaDRManager` service, and starts it.

Note: Windows Installer masks the enrollment token in installer logs, so a log option such as `/l*v` does not record it. Other property values are still logged. Keep the command with the token out of scripts and command history.

Installer properties

Add any of these properties to the command line to change the defaults. Only `ENROLLMENTTOKEN` is required.

Property	Default	Description
`ENROLLMENTTOKEN`	None (required)	The secret that agents present to enroll. Kept across upgrades. Giving a new value on an upgrade replaces the stored one.
`RESTPORT`	`5210`	The HTTPS port for the console, the REST API, and the `cdr` command-line tool. Kept across upgrades.
`GRPCPORT`	`5211`	The port agents connect to. Agents are installed with this port in their Manager address. Kept across upgrades.
`LISTENADDRESS`	`0.0.0.0`	The local address the Manager listens on. The default is every interface. On a machine with more than one network adapter, set it to the one address agents and operators use.

Property	Default	Description
		Kept across upgrades.
`REDIRECTPORT`	`80`	A plain HTTP port that only redirects a browser to the HTTPS console. It serves nothing else. `0` turns it off. Kept across upgrades. If another program already uses the port, the Manager logs a warning and starts without the redirect.

For example, to use different ports and listen on one address only:

```
msiexec /i CygnaDR.Manager.msi /qn ENROLLMENTTOKEN=<token> RESTPORT=8443
GRPCPORT=8444 LISTENADDRESS=10.0.0.10 REDIRECTPORT=0
```

What gets installed where

Item	Location and details
Program folder	`%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager`. Holds `CygnaDR.Manager.exe`, the `cdr.exe` command-line tool, the console files in `wwwroot`, the PDF renderer in `pdf`, and, in `redist`, the agent packages `CygnaDR.DcAgent.msi`, `CygnaDR.StorageAgent.msi`, and `CygnaDR.HostAgent.msi`. The Manager uses them to push DC agents, to send updates to agents that update themselves, and to offer the packages for download on the **Agents** screen and with `cdr agent package download`. `cdr.exe` is not added to the `PATH`; run it from this folder. The Manager runs `pdf\CygnaDR.PdfRenderer.exe` as a separate process for each PDF it makes (the recovery report and audit trail exports). It is not a service and opens no port.
Data folder	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager`, readable only by SYSTEM and Administrators. Holds the database `cygnadr.db` (including the CA), the daily database snapshots in `snapshots`, and the service logs in `logs`. A configuration backup is built in `config-backup-work` beside the database while it is sent, and removed afterwards.
Service	`CygnaDRManager`, display name Cygna DR Manager. Runs as LocalSystem, starts automatically (delayed start), and

Item	Location and details
	restarts after a failure.
Install-time settings	`HKLM\SOFTWARE\Cygna Labs\Disaster Recovery Suite\Manager\Configuration`, including the enrollment token, readable only by SYSTEM and Administrators.
Firewall rules	Inbound rules for the agent port, the console port, and the redirect port (unless it is off), scoped to `CygnaDR.Manager.exe`: Cygna DR Manager (agent gRPC), Cygna DR Manager (console), and Cygna DR Manager (console redirect). The rules apply to every network profile, so the console and agents can connect even when Windows classifies the network of a machine outside a domain as Public.

Find the CA fingerprint

The CA fingerprint identifies this Manager. Every agent is installed with it, and the `cdr` command-line tool asks you to confirm it the first time it connects. The fingerprint is the SHA-256 hash of the CA certificate, shown as 16 groups of four hexadecimal digits separated by hyphens. It is public, so it does not need to be kept in a safe. Keep a copy with your recovery notes, such as your runbook. After a recovery, it is how you confirm that a rebuilt Manager has the same CA.

The most reliable source of the fingerprint is the Windows Application event log on the Manager machine. At every start, the Manager writes an entry that contains `Cygna DR manager CA fingerprint:`. To find it with PowerShell:

```
Get-WinEvent -LogName Application -MaxEvents 500 |
  Where-Object Message -like '*Cygna DR manager CA fingerprint*' |
  Select-Object -First 1 -ExpandProperty Message
```

For the other places the fingerprint is shown, see Certificates and Trust.

Verify the installation

To verify that the Manager is running:

1. Check the service status:

   ```
   Get-Service CygnaDRManager
   ```

The status is `Running`. The service shows `StartPending` until the Manager has finished its startup work and is listening, which takes longer than usual the first time after an upgrade. The installer does not wait for it, so a Manager that fails to start does not fail the installation. If the service stops instead, the System log has its error code, the Application log has the reason, and the Manager log names the startup step that failed.

2. Confirm that the CA fingerprint line is in the Application event log, as described above.
3. Open `https://localhost:5210` in a browser on the Manager machine, or `https://<manager address>:5210` from your workstation.

   The console shows the **Set up this manager** page. Continue with Signing In for the First Time now.

If the service does not start or the console does not respond, see Installer Messages and Logs and Diagnostics.

Signing In for the First Time

A new Manager has no operator accounts. The first time anyone opens the console, it shows a one-time setup page that creates the **break-glass account** and sets the **recovery passphrase**:

- **Break-glass account**–An account that holds every capability and depends on nothing outside the Manager. You keep it for the day nothing else works, and use named operator accounts for everyday work. No administrator can change its roles or reset its password.
- **Recovery passphrase**–A passphrase that protects a copy of the Manager's CA key in its database. With a copy of the database and this passphrase, you can bring the Manager back on another machine. It also opens the Manager's configuration backups on the storage machine.



Note: **Important:** Whoever completes the setup first owns the Manager. Complete it immediately after you install the Manager. If you open the console of a Manager you did not just install and it still shows the setup page, treat that as a security finding.

Before you begin

- Install the Manager. See Installing the Manager.
- Choose a recovery passphrase of at least 16 characters. A few unrelated words are easier to keep than a short, complex string.
- Have a way to record the break-glass credential offline, such as paper you can put in a safe. The password is shown only once.

Open the console

The console is served by the Manager at `https://<manager>:5210`, where `<manager>` is the Manager's IP address or host name. If the HTTP redirect is on, `http://<manager>/` takes you there too. If you changed the port at installation, use that port instead.

Until you trust the Manager's CA on your workstation, the browser warns that the connection is not private. The certificate is genuine, but it is issued by the Manager's own CA, which your browser does not yet know. To remove the warning, see Trusting the Manager Certificate.

For the first sign-in, the safest choice is a browser on the Manager machine itself, at `https://localhost:5210`. The break-glass password then never crosses the network before you have confirmed the Manager's certificate.

On a wide enough window, a panel beside the setup and sign-in forms shows the product name, **Cygna Disaster Recovery Suite**, with the module it carries, **Cygna Forest Recovery for Active Directory**. Once setup is complete, the sign-in form is headed **Sign in to Cygna Disaster Recovery Suite**, at any window width.

Complete the setup

To set up the Manager:

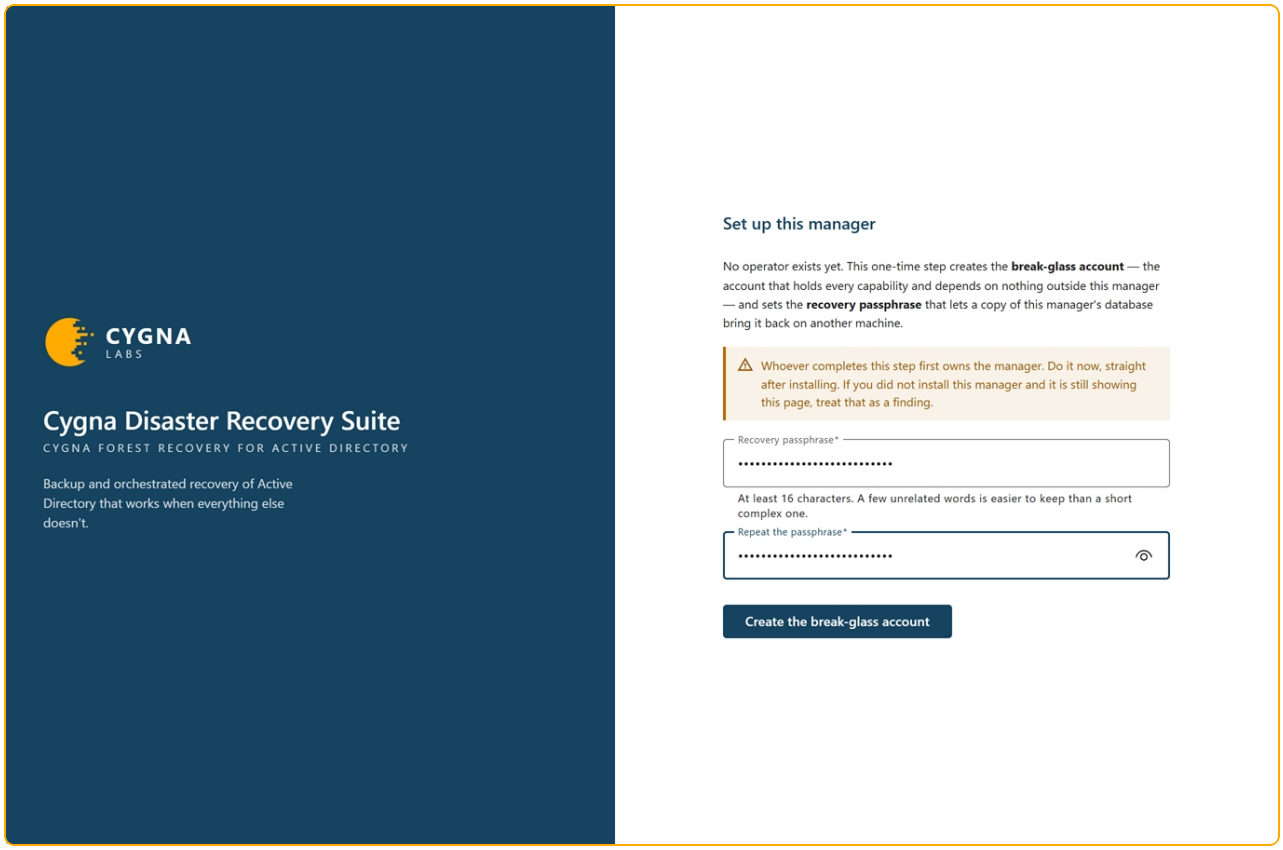
1. Open the console.

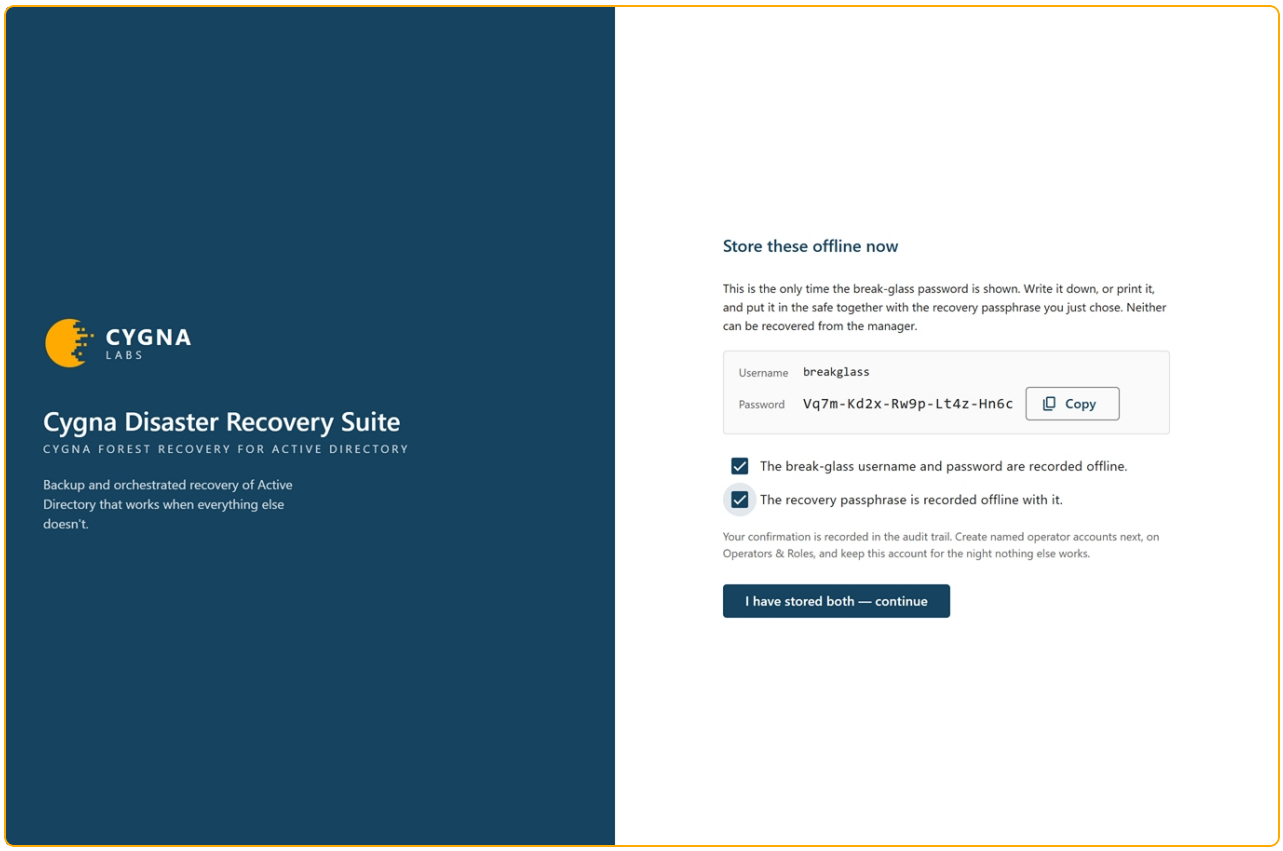
   The **Set up this manager** page opens.
2. In **Recovery passphrase**, enter the passphrase you chose.
3. In **Repeat the passphrase**, enter it again.
4. Select **Create the break-glass account**.

   The **Store these offline now** page shows the break-glass **Username** and **Password**, which the Manager generated.
5. Record the username and password offline, together with the recovery passphrase. Select **Copy** if you want to copy the password, for example to print it.

   Neither the password nor the passphrase can be recovered from the Manager.
6. Select **The break-glass username and password are recorded offline.**
7. Select **The recovery passphrase is recorded offline with it.**
8. Select **I have stored both — continue**.

   Your confirmation is recorded in the audit trail, and the console opens, signed in as the break-glass account.


CYGNA
LABS
Cygna Disaster Recovery Suite
CYGNA FOREST RECOVERY FOR ACTIVE DIRECTORY
Backup and orchestrated recovery of Active Directory that works when everything else doesn't.
Set up this manager
No operator exists yet. This one-time step creates the break-glass account — the account that holds every capability and depends on nothing outside this manager — and sets the recovery passphrase that lets a copy of this manager's database bring it back on another machine.
Whoever completes this step first owns the manager. Do it now, straight after installing. If you did not install this manager and it is still showing this page, treat that as a finding.
Recovery passphrase*
At least 16 characters. A few unrelated words is easier to keep than a short complex one.
Repeat the passphrase*
Create the break-glass account


CYGNA
LABS
Cygna Disaster Recovery Suite
CYGNA FOREST RECOVERY FOR ACTIVE DIRECTORY
Backup and orchestrated recovery of Active Directory that works when everything else doesn't.
Store these offline now
This is the only time the break-glass password is shown. Write it down, or print it, and put it in the safe together with the recovery passphrase you just chose. Neither can be recovered from the manager.
Username breakglass
Password Vq7m-Kd2x-Rw9p-Lt4z-Hn6c
Copy
The break-glass username and password are recorded offline.
The recovery passphrase is recorded offline with it.
Your confirmation is recorded in the audit trail. Create named operator accounts next, on Operators & Roles, and keep this account for the night nothing else works.
I have stored both — continue

If the page was closed before you confirmed, the next visit says that the break-glass account was created but its password cannot be shown again. If you recorded it, select **It was recorded — continue**. If you did not, select **Issue a new password**: the Manager shows a new password, and the old one stops working.

If the console says **Setup has already been completed on this manager. Sign in instead.** and you did not complete it, someone else holds the break-glass account. Treat this as a security finding, and do not enroll agents with this Manager.

Store the Manager's credentials offline

Keep these together in a safe or another offline location that does not depend on Active Directory:

- The break-glass username and password
- The recovery passphrase
- The enrollment token the Manager was installed with
- The break-glass account's recovery codes, if you turn on multi-factor authentication for it

The CA fingerprint is public, so it does not belong in the safe. Keep a copy with your recovery notes instead.

For everything else to do on the first day, see Manager Day-One Checklist. You can change the recovery passphrase later from the **Manager recovery** card on the **License** screen; replace the offline record when you do.

Apply the license key

A new Manager has no license, and without one it refuses every backup, restore, recovery plan, and test restore. Until you apply a key, a banner on every screen says that backups and recovery operations are stopped. Apply the key now, while you are signed in as the break-glass account, which can apply it.

To apply the license key:

1. Select **Administration** > **License**.
2. Select **Apply a license key…**.
3. In **License key**, paste the key exactly as Cygna sent it.

   The dialog shows what the key says: the licensee, type, number of people, and end date.
4. Select **Apply this license**.

The License page shows the license's standing. The banner disappears, unless the license ends within 30 days, as a short trial may; it then reads **License ending soon**.

You can also run `cdr license apply <KEY>`. For what the license enforces and how it warns before it ends, see Licensing.

Create named operator accounts

Use the break-glass account only to set up named accounts, and then keep it in the safe. Every action it takes is recorded in the audit trail as break-glass, and named accounts let you see who did what.

Operator accounts belong to Cygna Forest Recovery itself. They are stored in the Manager database, not in Active Directory or Windows, so they work when the directory is down. Each account gets one or more roles: **Viewer / Auditor**, **Backup Operator**, **Restore Operator**, and **System Admin**. A **System Admin** manages agents, accounts, and configuration but cannot run backups or restores, so the people setting up the deployment usually need more than one role.

To create the first named accounts:

1. Select **Administration** > **Operators & Roles**.
2. Create an account for each person, with the roles they need.

   The Manager generates a temporary password for each new account and shows it once. The person chooses their own password at their first sign-in.
3. Sign out of the break-glass account and sign in with your named account.

Each operator can then turn on multi-factor authentication for their own account from the user menu, so that signing in also needs a code from an authenticator app on their phone. It does not need internet access. The break-glass account can use it too; if you turn it on, its recovery codes go in the safe with its password. See Multi-Factor Authentication.

For the details of each role and how to create accounts, see Managing Operators and Roles.

Trusting the Manager Certificate

The Manager creates its own certificate authority (CA) when it first starts, and uses it to issue the certificate for the console. Browsers do not know this CA, so they warn about the connection until you add the CA to the trusted root certificates on each workstation that operators use. Agents need no action: they are installed with the CA fingerprint and trust it from the start.

Trusting the CA is a one-time task for each workstation. The CA is valid for 20 years. The Manager reissues its own server certificate automatically, and the new certificate is trusted

because it comes from the same CA. You trust the CA again only if the Manager is re-keyed. See Re-Keying the Manager CA.

For background, see Certificates and Trust.

Before you begin

- Have the CA fingerprint you read from the Manager's event log. See Installing the Manager.

Download the CA certificate

Get the CA certificate in either of these ways.

From the Manager's trust endpoint

The Manager publishes its CA certificate at `https://<manager>:5210/api/trust`. The endpoint needs no sign-in and returns the fingerprint (`fingerprint`) and the CA certificate in PEM format (`caCertificatePem`). For example, in PowerShell 7:

```
$trust = Invoke-RestMethod -Uri https://<manager>:5210/api/trust -SkipCertificateCheck
Set-Content -Path .\manager-ca.pem -Value $trust.caCertificatePem
certutil -decode .\manager-ca.pem .\manager-ca.cer
```

The certificate check is skipped here because the CA is not trusted yet. That is why you must verify the fingerprint before you import the certificate.

From the browser

Open the console, view the site's certificate from the address bar, select the top certificate in the certification path (the Manager's CA), and export it to a file.

Verify the fingerprint

Before you import the certificate, confirm that it is your Manager's CA. Compare its SHA-256 hash with the fingerprint you took from the Manager's event log on the Manager machine itself. A certificate fetched over the network could come from an impostor, so do not compare it with a value fetched over the same connection.

To verify the fingerprint:

1. Calculate the SHA-256 hash of the certificate file. The file must be in DER format, like the `.cer` file that `certutil -decode` writes:

   ```
   certutil -hashfile .\manager-ca.cer SHA256
   ```

2. Compare the result with the fingerprint from the event log, ignoring letter case and the hyphens between the groups of four digits.

   If the two values differ, do not import the certificate. You are not connected to your Manager.

Import the certificate

Import the CA into the **Trusted Root Certification Authorities** store. Choose the store that fits how many workstations need it.

For your own account

Run this in a command prompt as yourself. No administrator rights are needed; Windows asks you to confirm the installation.

```
certutil -user -addstore Root .\manager-ca.cer
```

For every user of a workstation

Run this in an elevated command prompt:

```
certutil -addstore Root .\manager-ca.cer
```

For many workstations

Distribute the certificate with Group Policy or your device management tool. In a Group Policy Object, import the certificate under **Computer Configuration** > **Policies** > **Windows Settings** > **Security Settings** > **Public Key Policies** > **Trusted Root Certification Authorities**.

Note: Keep a way to reach the console that does not depend on Group Policy. In a directory outage, workstations that rely on a policy for the CA still have it, but a newly built workstation does not. The per-user `certutil` command works on any machine.

Microsoft Edge and Google Chrome use the Windows certificate store. Firefox uses its own store by default; import the certificate in Firefox's certificate settings under **Authorities**, or configure Firefox to trust the Windows root store.

After the import, restart the browser and open the console again. The warning no longer appears. The Manager's certificate covers its host name, its fully qualified name, `localhost`, and all its IP addresses, so you can connect by any of them.

Trust the Manager in the cdr command-line tool

The `cdr` command-line tool keeps its own list of trusted Managers for each Windows user, separate from the Windows certificate store. The first time it connects to a Manager, it shows the CA fingerprint and asks you to confirm it:

```
cdr has not connected to the manager at https://localhost:5210/ before.
Its CA fingerprint is:
  3F2A-91C0-...
Compare it with the fingerprint on the manager's console, or in its event log. Trust it only if they
match: a different value means you are not talking to your manager.
Trust this manager and remember it for this address? [y/n] (n):
```

Compare the value with the fingerprint from the event log, and answer `y` only if they match. `cdr` remembers the fingerprint for that address and checks it on every later connection.

- To trust a Manager without running another command, run `cdr trust pin --manager https://<manager>:5210`.
- In a script, where nobody can answer the prompt, give the fingerprint with `--manager-ca <fingerprint>` or the `CYGNADR_MANAGER_CA` environment variable.
- To see the Managers you trust, run `cdr trust list`. To forget one, for example after it is re-keyed, run `cdr trust remove --manager https://<manager>:5210`.

For details, see Agent and Trust Commands.

Setting Up the Storage Machine

The storage machine holds every backup. Domain controllers write each backup into a staging area over SMB, and the storage agent then verifies it and moves it into the vault, which has no share and cannot be reached over the network. Under the storage root, the agent keeps three folders:

- `<StorageRoot>\staging`—Where domain controllers write backups in progress. You share this folder.
- `<StorageRoot>\vault`—Where committed backups are kept. It is never shared.
- `<StorageRoot>\manager-backups`—Where the Manager's encrypted configuration backups are kept, the copies from which a lost Manager is rebuilt. It has the vault's permissions and is never shared. See Manager Configuration Backups.

Setting up the machine takes four tasks, in this order: prepare the storage folder, share the staging folder, open the firewall for SMB, and install the storage agent. Turning on auditing is also recommended. The agent looks for its staging share when it starts, so create the share before you install it.

For how backups move through staging into the vault, see How Backup Works.

Before you begin

- Use a machine running 64-bit Windows Server 2019 or later that is not joined to a domain and has a fixed IP address. It can be the Manager machine, but never a domain controller. See System Requirements.
- Provide a volume with enough space for the backups you plan to keep. See Sizing Backup Storage.
- Put the storage root on a local NTFS or ReFS volume. The storage agent watches the vault through the volume's change journal, which other file systems and network shares do not have. See Detecting changes to committed backups.
- Turn on BitLocker for that volume, with auto-unlock, so that the backups are encrypted at rest and the volume is available after a restart. See Encrypting the vault volume.
- Install the Manager and complete the first sign-in. See Installing the Manager and Signing In for the First Time.
- Have the enrollment token the Manager was installed with.
- Sign in to the storage machine with a local administrator account.

Prepare the storage folder

Choose an absolute path on the storage volume for the storage root, for example `E:\CygnaDR`. You give this path to the installer as `STORAGEROOT`, and it cannot be changed from the Manager.

Create the storage root and its `staging` folder now, so that you can share the folder before the agent starts:

```
New-Item -ItemType Directory -Path E:\CygnaDR\staging
```

The agent creates the `vault` and `manager-backups` folders itself and sets the permissions on all three folders when it starts.

Note: The storage agent does not start if the storage root is unreachable. If the volume is attached after the agent is installed, restart the agent once the volume is present.

Share the staging folder

Create an SMB share over `<StorageRoot>\staging`. Share this folder only:

- Do not share the storage root. A share over the root also exposes the vault beside the staging folder.

- Do not rely on an administrative share such as `E$`. The agent ignores administrative shares.

For example, in an elevated PowerShell session:

```
New-SmbShare -Name CygnaDRStaging -Path E:\CygnaDR\staging -FullAccess "BUILTIN\Administrators"
```

You do not need to grant domain controllers access to the share. Each backup job uses a local account that the storage agent creates for that job only. When the agent starts, it checks the share's permissions and grants its **CygnaDR Staging Writers** group the **Change** permission if it is missing. If share permissions are controlled by someone else in your organization, you can turn this off with the **Manage the staging share's permissions** setting; the agent then only reports what is missing.

Don't open the staging share, or the staging folder through another share, with any other account, for example to look at it. The storage agent reports any account other than a job's own as **Unexpected SMB connections to the staging share**, a critical alert. Work in the folder locally on the storage machine instead.

Open the firewall for SMB

Domain controllers reach the staging share over TCP port 445, and so do recovery environments when they read a backup during a full system recovery. The storage agent installer does not open this port. Allow inbound TCP 445 from your domain controllers' addresses, for example:

```
New-NetFirewallRule -DisplayName "Cygna DR staging (SMB)" -Direction Inbound -Protocol TCP -LocalPort 445 -RemoteAddress 10.0.1.0/24,10.0.2.0/24 -Action Allow
```

Replace the addresses with the subnets your domain controllers use. If you plan server recoveries onto replacement machines, include the networks those machines are on. For the full list of ports, see Ports and Protocols.

Turn on auditing (recommended)

The storage agent reads the storage machine's audit policy and never changes it. For it to see every connection to the staging share, however short, and to say who changed a committed backup, turn on success auditing for **Logon**, **File Share**, **Detailed File Share**, and **File System** in the Advanced Audit Policy of the storage machine. See Audit policy on the storage machine.

```
auditpol /set /subcategory:"Logon" /success:enable
auditpol /set /subcategory:"File Share" /success:enable
auditpol /set /subcategory:"Detailed File Share" /success:enable
auditpol /set /subcategory:"File System" /success:enable
```

Install the storage agent

Install the storage agent by hand on the storage machine, with the command line the console builds for you. After that first install, it updates itself from the Manager, unless a policy on the storage machine turns that off. See Maintaining Agents.

An installation has one storage machine: the first storage agent to enroll. A storage agent installed on another machine after that is a candidate and is given no work until the storage machine is replaced with it, which ends every restore point. To move the storage machine later, for example from a trial setup into production, see Moving the Storage Machine or an Isolated Recovery Host.

To install the storage agent:

1. In the console, select **Administration** > **Agents**.

   Before any agent has enrolled, the screen shows an install command for each kind of agent on the **Agent packages** card, already filled in with the Manager's address and CA fingerprint. Once agents are enrolled, select **Install agent** in the page header to see the same commands.

2. Under **Storage agent**, select **Copy command**.

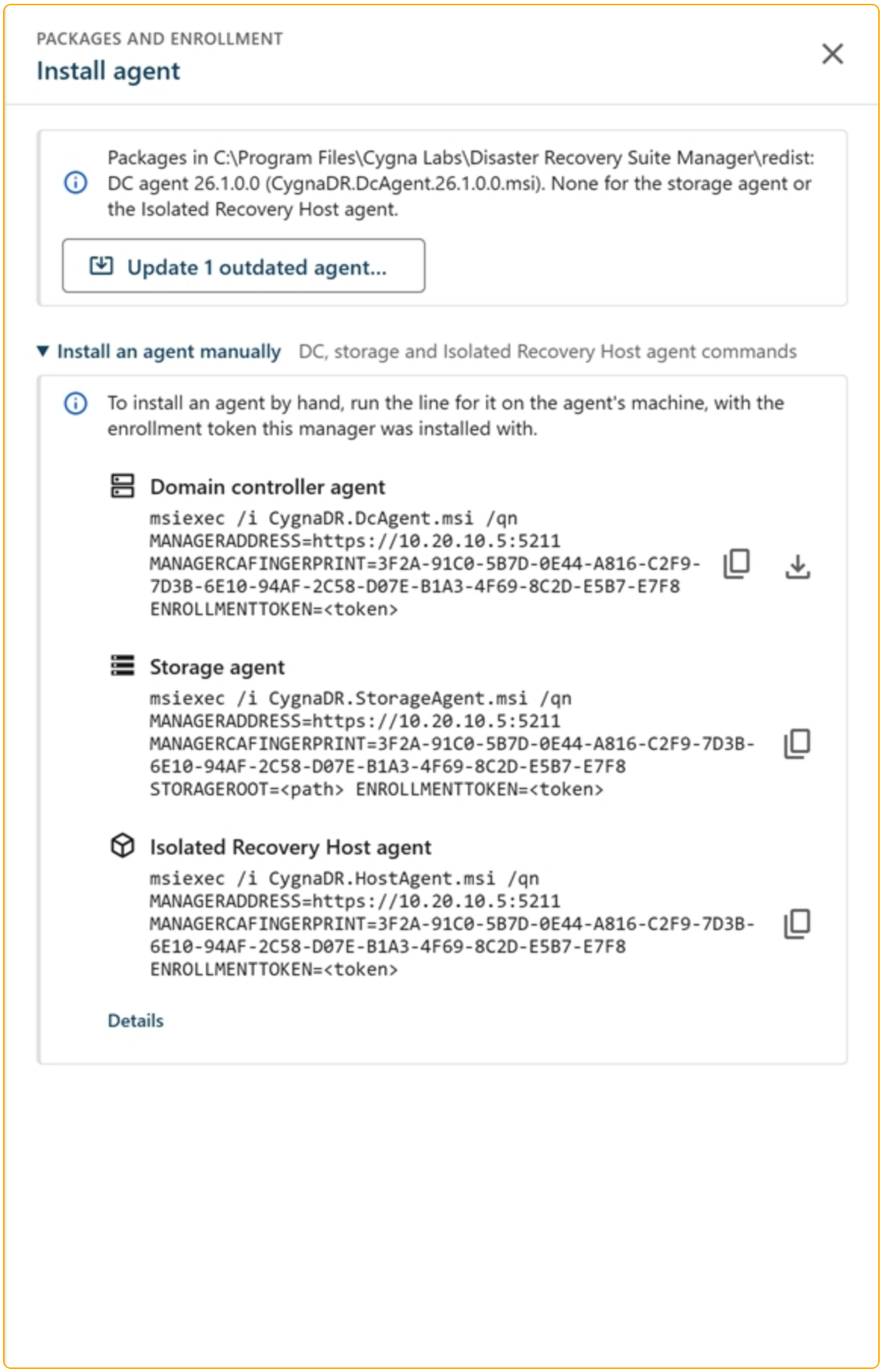
3. Copy `CygnaDR.StorageAgent.msi` to the storage machine, and open an elevated command prompt in that folder.

   The Manager holds the package in the `redist` folder in its program folder. You can also download it with the download button beside the storage agent command, which needs the `agents.manage` capability, or with `cdr agent package download storage`.

4. Paste the command, replace `<path>` with your storage root and `<token>` with the enrollment token, and run it. The command looks like this:

   ```
   msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
   MANAGERCAFINGERPRINT=<fingerprint> STORAGEROOT=E:\CygnaDR
   ENROLLMENTTOKEN=<token>
   ```

   The installer installs and starts the `CygnaDRStorageAgent` service. The agent connects to the Manager and enrolls.


PACKAGES AND ENROLLMENT
Install agent
Packages in C:\Program Files\Cygna Labs\Disaster Recovery Suite Manager\redist: DC agent 26.1.0.0 (CygnaDR.DcAgent.26.1.0.0.msi). None for the storage agent or the Isolated Recovery Host agent.
Update 1 outdated agent...
Install an agent manually
DC, storage and Isolated Recovery Host agent commands
To install an agent by hand, run the line for it on the agent's machine, with the enrollment token this manager was installed with.
Domain controller agent
msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 ENROLLMENTTOKEN=<token>
Storage agent
msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 STORAGEROOT=<path> ENROLLMENTTOKEN=<token>
Isolated Recovery Host agent
msiexec /i CygnaDR.HostAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 ENROLLMENTTOKEN=<token>
Details

Before you rely on the fingerprint in the command, compare it with the one in the Manager's event log. Use the Manager's IP address in `MANAGERADDRESS`, not its name: the Manager is not joined to the domain, so its name is often not in DNS, and DNS may be down during a recovery. If the Manager has more than one address, the Agents screen lists the others; choose the one the storage machine can reach.

Note: The enrollment token is a credential that can enroll new agents. Windows Installer masks it in installer logs, but do not leave the command in scripts or command history.

The package is signed by Cygna Labs Corp. To check it before you install, run `Get-AuthenticodeSignature CygnaDR.StorageAgent.msi` in PowerShell and expect the status `Valid`, or open the file's **Properties** > **Digital Signatures** tab.

The program files go to `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Storage Agent`, and the agent's state and logs to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent`. The installer also places the recovery agent beside the storage agent; it is copied into each recovery image and is not installed or enrolled separately.

Verify the storage agent

To verify the storage agent:

1. In the console, select **Administration** > **Agents**.

   The storage machine is listed with **Agent type Storage** and **Connection Connected**.

2. Select **Protection** > **Repository**.

   Within a few minutes, the screen shows the storage machine's capacity.

If the agent does not appear, see Agent Enrollment and Connectivity.

Machines with more than one network address

The storage agent tells domain controllers to write to `\\<address>\<share>`, using an address it detects on its own machine, so that backups do not depend on DNS. On a machine with more than one plausible address, the agent cannot tell which network domain controllers arrive on, so it advertises its machine name instead. Backups then depend on the domain controllers resolving that name through DNS.

On a multi-homed storage machine, set the address yourself: select **Administration** > **Configuration**, select the **Storage agents** tab, and in the **Staging** group set **Advertised host** to the address domain controllers should use. Save the change and restart the storage agent:

```
Restart-Service CygnaDRStorageAgent
```

Local groups the agent creates

When it starts, the storage agent creates three local groups on the storage machine. CygnaDR Administrators has access to the vault, the staging folder, and the Manager's configuration backups. CygnaDR Staging Writers and CygnaDR Vault Readers hold only the accounts the agent creates for backup jobs and restores. To give a local account access to the backups, add it to CygnaDR Administrators rather than editing folder permissions. Do not add accounts to the other two groups yourself. For what each group is for, see Accounts and Permissions.

Optional: remove local administrators from the vault

By default, the local Administrators group keeps access to the vault, staging, and manager-backups folders. You can remove it, leaving only SYSTEM and CygnaDR Administrators. This is not a hard barrier, because a local administrator can take ownership of the folders. It does mean that reading the backups requires a deliberate ownership change that leaves evidence, instead of looking like normal administration.

To remove local administrators from the vault:

1. Add at least one local account to the CygnaDR Administrators group, for example:

   ```
   Add-LocalGroupMember -Group "CygnaDR Administrators" -Member storageadmin
   ```

   The setting is refused while the group is empty, so that you cannot lock yourself out of your own vault.
2. Select **Administration** > **Configuration**, and then select the **Storage agents** tab.
3. In the **Vault** group, turn on **Remove BUILTIN\Administrators from the vault**, and select **Save**.
4. Restart the storage agent.

   The agent rewrites the folder permissions when it starts.

For the other storage agent settings, see Settings Reference.

Deploying DC Agents

The DC agent runs on every domain controller, including read-only domain controllers. It reports the directory's topology, runs the backups the Manager schedules, and carries out restores and recovery steps. The agent only connects out to the Manager, so it needs no inbound firewall rule.

You can deploy the agent in three ways:

- **From the console**—The Manager pushes the agent to domain controllers it knows about, using an administrator account you supply for that run.
- **With the cdr command-line tool**—`cdr agent deploy` pushes the agent from the machine where you run it, to named domain controllers or to a selection from the Manager's inventory.
- **With the MSI package**—Run the installer on each domain controller yourself, or with your software distribution tool.

The console lists only domain controllers that an agent has reported, so it cannot push the first agent. Install the agent on one domain controller in each forest by hand or with `cdr agent deploy`. That agent enumerates its forest and reports every domain controller it finds. The rest then appear on the **Domain Controllers** screen with no agent, ready for a push from the console.

Push installation is for a domain controller's first agent. Once installed, the DC agent updates itself from the Manager: it receives the package over its own connection, so an update needs no credential and nothing connects into the domain controller. The Manager falls back to a push install only for a DC agent that cannot update itself, because it is not connected or a policy on its machine switches updates from the Manager off. To update agents, use **Administration** > **Agents** or `cdr agent update`; see Maintaining Agents, which also describes the policy.

Before you begin

- Install the Manager and set up the storage machine. See Installing the Manager and Setting Up the Storage Machine.
- Make sure each domain controller can reach the Manager on TCP 5211 and the storage machine on TCP 445. See Ports and Protocols.
- Make sure each domain controller has the Windows Server Backup feature, or let the agent install it. See Windows Server Backup below.

Prerequisites for a push installation

The console and `cdr agent deploy` install the agent the same way and have the same prerequisites:

- **An account**—An account that is a local administrator on each target domain controller (Domain Admin in most environments), given as `DOMAIN\user` or `user@domain`. The password is used for that run only and never stored.
- **On each target**—The administrative share `ADMIN$` (TCP 445), the Remote Registry service, and either WinRM (TCP 5985) or DCOM (TCP 135 and dynamic RPC ports, with the **Windows Management Instrumentation (WMI-In)** firewall rule group enabled). All of these are on by default on Windows Server, except the WMI-In rules.

- **For WinRM from outside the domain**–The machine that runs the installation must list each target in its WinRM TrustedHosts. For a console push, that machine is the Manager; for `cdr agent deploy`, it is the machine where you run `cdr`. To avoid TrustedHosts, use DCOM instead.

To add targets to TrustedHosts, run this in an elevated PowerShell session on the machine that runs the installation:

```
Set-Item WSMan:\localhost\Client\TrustedHosts -Value "dc01.-
corp.example.com,dc02.corp.example.com" -Concatenate -Force
```

For the accounts and rights involved, see Accounts and Permissions.

Deploy from the console

To push from the console, you need the `agents.manage` capability, which the **System Admin** role has. The Manager deploys the DC agent package from its package folder. By default, this is the `redist` folder installed with the Manager, which holds the agent builds that shipped with it. You can change the folder, or the default transport (WinRM or DCOM), in the **Agent deployment** group on the **Manager** tab of the **Configuration** screen.

To deploy the DC agent from the console:

1. Select **Active Directory** > **Domain Controllers**.
2. Start the deployment in one of these ways:
   - For one domain controller, select the row's actions menu (⋮), and then select **Install agent…**.
   - For several, select their checkboxes, and then select **Deploy the agent to *n*…** in the bar at the bottom of the screen. To find the ones that still need an agent, filter the list to those without an agent first.

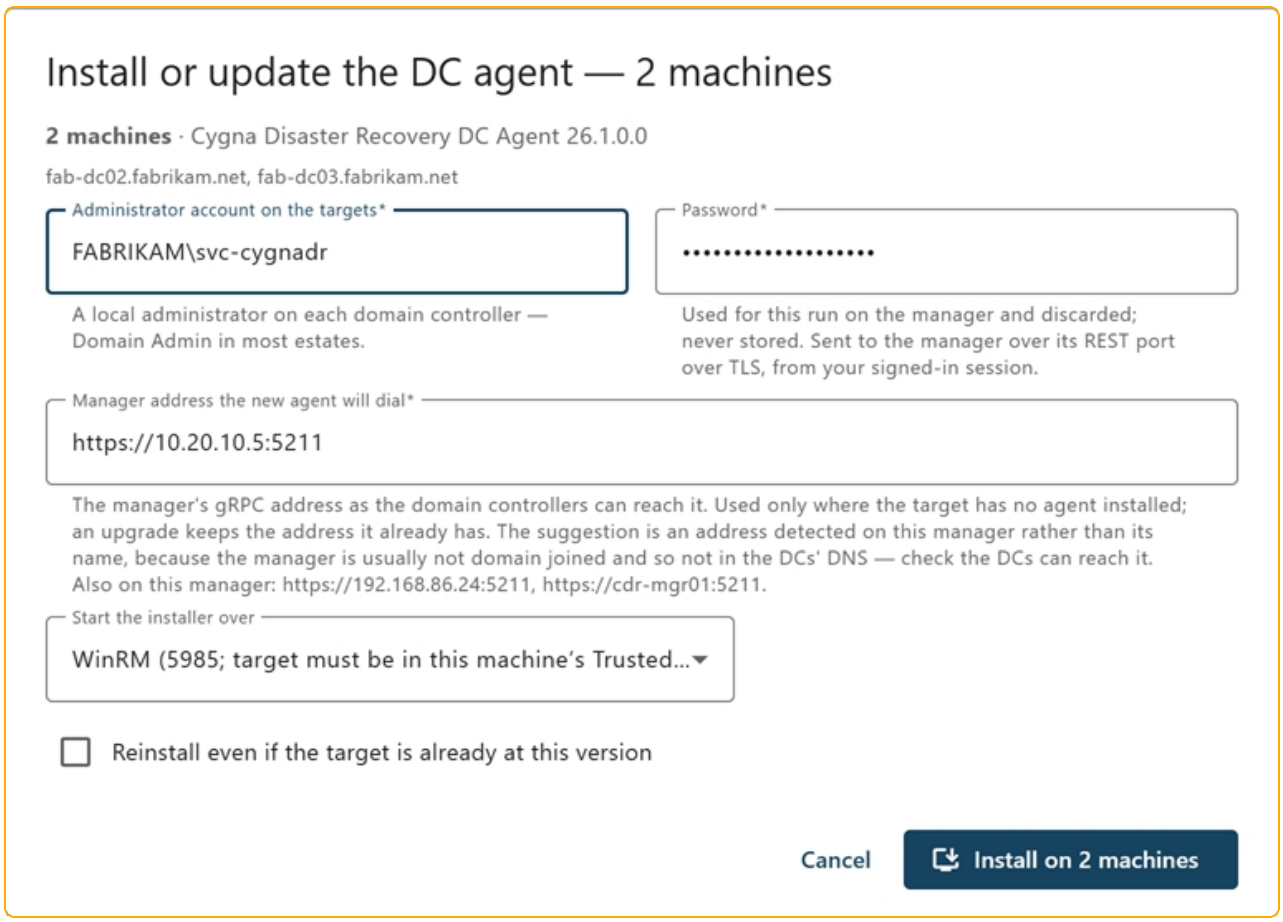
   A dialog titled **Install the DC agent**, or **Install or update the DC agent** for several machines, opens and lists the targets and the package version.
3. In **Administrator account on the targets**, enter the account, for example `CORP\admin`.
4. In **Password**, enter its password.
5. Check **Manager address the new agent will dial**.

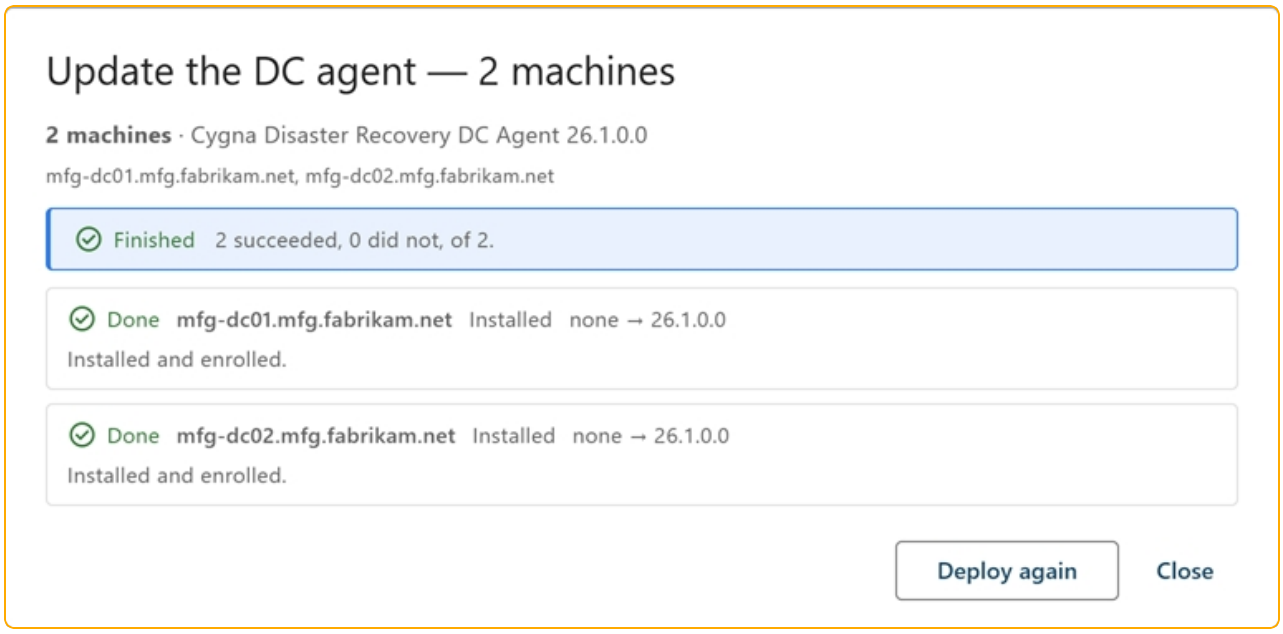
   The console suggests an address detected on the Manager. Make sure the domain controllers can reach it. Use an IP address rather than a name, because the Manager is usually not in the domain's DNS.
6. In **Start the installer over**, choose **WinRM (5985; target must be in this machine's TrustedHosts)** or **DCOM (135 + RPC; no TrustedHosts needed)**.

7. Select **Install on this machine** or **Install on *n* machines**.

   The dialog shows each target's progress. You can select **Close – the run continues** to let it finish in the background. When it finishes, the dialog shows how many targets succeeded, and the version each one moved from and to.

Note: To update agents without entering a credential, use the **Agents** screen. **Update agent…** on the **Domain Controllers** screen asks for an administrator account, for any target that must be pushed to.





The dialog fields:

Field	Description
Administrator account on the targets	Required. A local administrator on each target, as `DOMAIN\user` or `user@domain`.
Password	Required. Sent to the Manager over TLS from your signed-in session, used for this run, and discarded.
Manager address the new agent will dial	The Manager's agent address as the domain controllers reach it, for example `https://10.0.0.10:5211`. Used only where the target has no agent; an update keeps the address the agent already has. When you update agents, the field is labeled **Manager address if this turns out to be a fresh install** and is optional.
Start the installer over	WinRM (the default) or DCOM. See the prerequisites above.
Reinstall even if the target is already at this version	Off by default. Turn it on to repair an agent that is already current.

You do not enter the enrollment token or the CA fingerprint. The Manager supplies its own.

Deploy with cdr agent deploy

`cdr agent deploy` runs on Windows. It reaches each domain controller directly over SMB and WinRM or DCOM, and it never sends the administrator password to the Manager. `cdr.exe` is installed in the Manager's program folder, and when you run it there it uses the DC agent package in the `redist` folder beside it. Elsewhere, give the package with `--package`.

The password is read from the `CYGNADR_DEPLOY_PASSWORD` environment variable or prompted for. For new installations, the enrollment token is read from `CYGNADR_ENROLLMENT_TOKEN` or prompted for. Neither is accepted as an argument, so neither lands in your command history. Each new agent is given the CA fingerprint of the Manager that `cdr` trusts.

Name the targets, or select them from the Manager's inventory with `--all`, `--without-agent`, or `--outdated`, optionally restricted to one domain with `-d`. The selectors read the inventory from the Manager, so sign in with `cdr login` first. See Connecting and Signing In.

On the Manager machine, open a command prompt in `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager`. Then, for example:

- Install the first agent in a forest, on a named domain controller:

  ```
  cdr agent deploy dc01.corp.example.com --user CORP\admin --manager-address https://10.0.0.10:5211
  ```

- See what would happen to every domain controller in one domain that has no agent, without changing anything:

  ```
  cdr agent deploy --without-agent -d corp.example.com --user CORP\admin --manager-address https://10.0.0.10:5211 --plan
  ```

- Install the agent on every domain controller that has none:

  ```
  cdr agent deploy --without-agent --user CORP\admin --manager-address https://10.0.0.10:5211
  ```

- Install or update the agent on every domain controller in the inventory, over DCOM:

  ```
  cdr agent deploy --all --user CORP\admin --manager-address https://10.0.0.10:5211 --transport dcom
  ```

- Push the package to every agent that is older than it:

  ```
  cdr agent deploy --outdated --user CORP\admin
  ```

To bring installed agents up to date, `cdr agent update --outdated` is usually the better choice: agents that can update themselves do so with no credential, and only domain controllers that cannot are pushed to.

When you run `cdr` on the Manager, it connects to the Manager as `localhost`, so you must give `--manager-address` for any new installation. A new agent cannot dial `localhost`.

`cdr agent deploy` exits with code `0` when every target ended where it was asked to be, including targets that were already current, and `7` when at least one did not. The table it prints says which and why. For every option, see Agent and Trust Commands.

Install with the MSI package

Use the MSI package directly when a push is not possible, or when you deploy software with your own tools. The installer is silent and takes these properties:

Property	Description
`MANAGERADDRESS`	Required. The Manager's agent address, for example `https://10.0.0.10:5211`. Use the IP address; a name works but the agent logs a warning. A restored domain controller's agent dials this address from the isolated recovery network, where the name may not resolve. See Isolating Recovery Targets.
`MANAGERCAFINGERPRINT`	Required on a first installation. The Manager's CA fingerprint. Public, not secret.
`ENROLLMENTTOKEN`	The Manager's enrollment token. The agent needs it to enroll.

To install the DC agent by hand:

1. In the console, select **Administration** > **Agents**, and select **Copy command** next to the DC agent command.
2. Copy `CygnaDR.DcAgent.msi` to the domain controller, and open an elevated command prompt in that folder.

   Take the package from the `redist` folder in the Manager's program folder, from the download button beside the DC agent command (which needs the `agents.manage` capability), or with `cdr agent package download dc`.
3. Paste the command, replace `<token>` with the enrollment token, and run it:

```
msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
```

The installer installs and starts the `CygnaDRDcAgent` service, which enrolls with the Manager.

To deploy the package with a software distribution tool such as Configuration Manager, run the same command line as a silent installation under the local SYSTEM account. Keep the enrollment token out of places other users can read. In particular, do not put it in a Group Policy startup script: scripts in SYSVOL are readable by every domain user. Windows Installer masks the token in installer logs, so a log option such as `/l*v` does not expose it. The package is signed by Cygna Labs Corp., which application control policies such as WDAC and AppLocker can allow by publisher.

The program files go to `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite DC Agent`, and the agent's state, job logs, and service logs to `%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent`.

Windows Server Backup

The DC agent runs backups with Windows Server Backup (`wbadmin`), which is not installed on Windows Server by default. Install it on each domain controller in either of these ways:

- Install the feature yourself, for example with `Install-WindowsFeature Windows-Server-Backup`.
- Let the agent install it: on the **Domain controller agents** tab of **Administration** > **Configuration**, leave **Install the Windows Server Backup feature** in the **Windows Server Backup** group on (the default). The agent then installs the feature at startup when `wbadmin` is missing. Turn the setting off, for the fleet or for one domain controller, where adding a Windows feature must go through change control; that domain controller's backups then fail until you install the feature. A newly installed agent also acts on the setting when it receives its first configuration from the Manager; an agent that is already running acts on a change at its next restart.

A backup that starts while the agent is installing the feature waits for the install to finish instead of failing. A backup that reaches a newly installed agent before its first configuration waits up to 2 minutes for that configuration, which decides whether the agent installs the feature. If the install fails, a restart still pending on the domain controller, for example after its promotion, is the usual cause: restart the domain controller, and the agent tries again at startup.

Verify enrollment

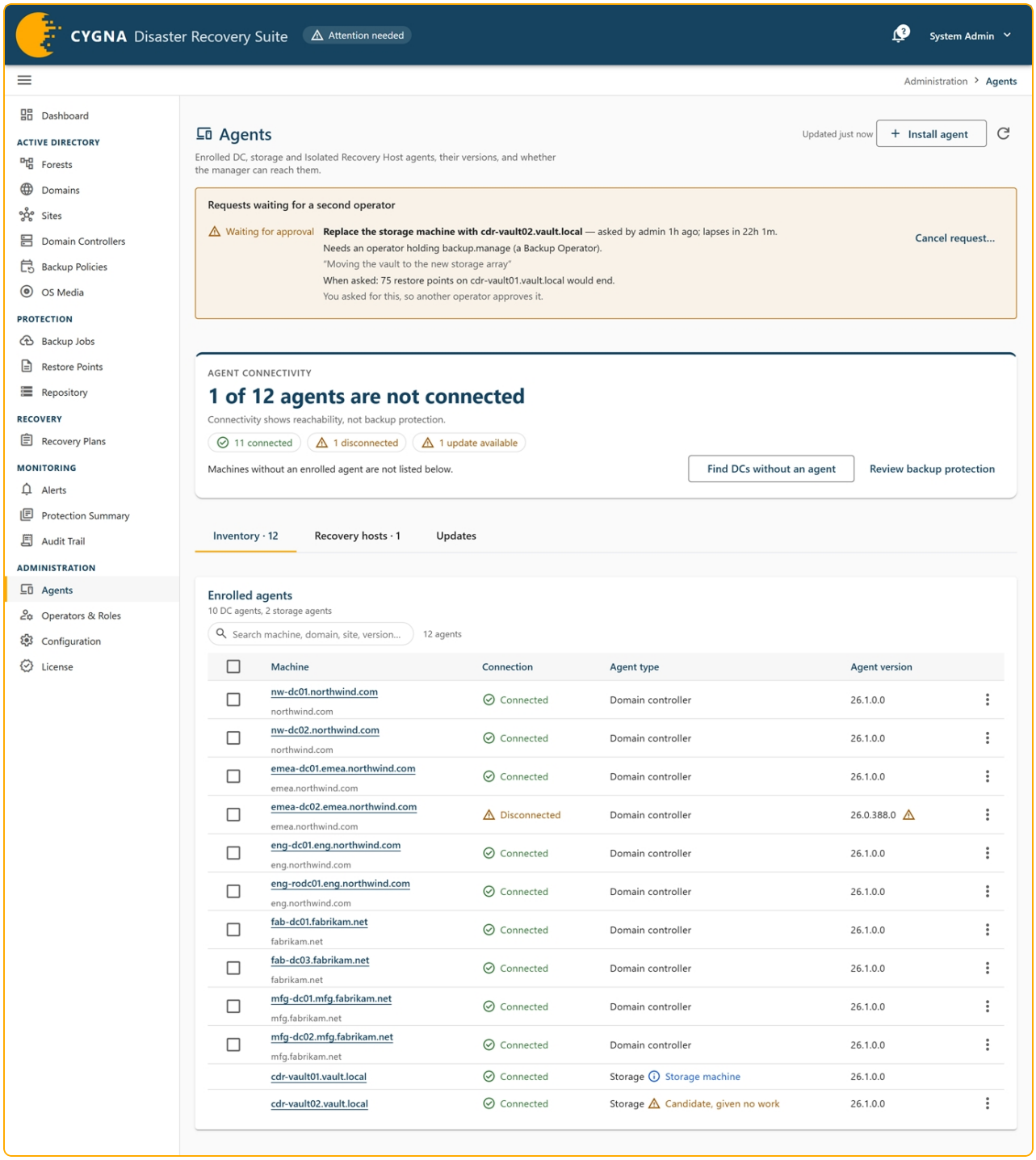
To verify that the agents enrolled:

1. Select **Administration** > **Agents**.

   Each domain controller is listed with **Kind Domain controller**, its agent version, and **Connection Connected**.

2. Select **Active Directory** > **Domain Controllers**.

   Each domain controller shows **Connected** in the **Agent** column. Domain controllers that still have no agent show **No agent**.



If an agent does not appear or stays disconnected, see Agent Enrollment and Connectivity.

Install the agent on candidate machines

A candidate is a machine that is not a domain controller and is set aside to receive a full system recovery when a domain controller's own hardware will not start. The candidate needs the DC agent too, enrolled with the same Manager, so that the Manager can survey it and stage the recovery on it. Install it by hand with the same DC agent command line.

The candidate appears on the **Agents** screen once it enrolls. Because it is not a domain controller, it is not listed on the **Domain Controllers** screen and is not counted in coverage figures or alerts.

For how candidates are used, see Full System Recovery onto a Candidate.

To keep agents up to date later, see Upgrading and Uninstalling. To stop a machine accepting updates from the Manager, set the `AllowRemote` policy described in Maintaining Agents.

Setting Up OS Media and Recovery Images

A full system recovery restores a whole domain controller from a full system backup, in place or onto a replacement machine. To do that, the machine starts into a recovery image: a small Windows PE environment that restores every critical volume from the vault. The storage agent builds one recovery image for each operating system variant your domain controllers run, from Windows Server installation media that you supply. It builds them ahead of time, because a build takes minutes and needs the ISO, and both are scarce during a disaster.

Until this is set up, domain controllers cannot use full system recovery. DC state recovery, which restores the directory on a domain controller that still starts Windows, does not need a recovery image.

No Windows Assessment and Deployment Kit (ADK) is needed. The storage agent builds each image with the DISM tool built into Windows, from the Windows Recovery Environment inside your own ISO, and adds the recovery agent to it.

For how recovery images are used, see Recovery Images.

Before you begin

- Set up the storage machine. See Setting Up the Storage Machine.
- Get a Windows Server ISO for each operating system variant your domain controllers run. See the matching rules below.
- Make sure the storage machine has room for the ISOs, the built images, and a build working folder roughly the size of one image.
- To change these settings, you need the `config.manage` capability, which the **System Admin** role has.

How media is matched

For each domain controller, the operating system that matters is the one recorded in its newest backup, because that is what a restore puts back. An ISO covers a domain controller when it contains a Windows image that matches:

Property	How it is compared
Build	Must match exactly, for example 20348 for Windows Server 2022.
Edition	Must match exactly, for example Standard or Datacenter.
Installation type	Must match exactly: Server Core or Desktop Experience. Desktop Experience media cannot produce a Server Core machine.
Language	Should match. Media in another language is reported as a warning: it works, but the display language and the tool output during a recovery differ from what your administrators expect.
Architecture	Must match, for example x64 or ARM64.
Update revision	Not compared. An ISO is a base image, and your domain controllers have been patched since; the console shows the difference for information only.

One ISO can cover several variants if it holds several images, for example both Standard and Datacenter, each in Server Core and Desktop Experience.

Set up the folders and settings

Use two separate folders on the storage machine: one for your ISOs, which Cygna Forest Recovery only reads, and one for the recovery images it builds. Keep both outside the storage root's `staging` and `vault` folders.

To set up OS media and recovery images:

1. On the storage machine, create the two folders, for example:

   ```
   New-Item -ItemType Directory -Path E:\os-media, E:\recovery-images
   ```

2. Copy your Windows Server ISOs into the media folder. Subfolders are scanned too.
3. In the console, select **Administration** > **Configuration**, and then select the **Storage agents** tab.
4. In the **OS media library** group, set **Media folder** to the ISO folder, for example `E:\os-media`.
5. In the **Recovery images** group, set **Recovery image folder** to the image folder, for example `E:\recovery-images`.
6. Optionally, give the storage agent a folder of drivers to add to every image. Set it on the storage machine: add `"RecoveryImages": { "DriverFolder": "E:\\drivers" }` to

the `CygnaDR` section of the storage agent's `appsettings.json`, and restart the service. See Settings Reference.

Virtual machines are usually served by the drivers already in Windows PE. Physical servers often need the vendor's storage controller and network adapter drivers. Drivers in this folder are added even if they are unsigned.

7. Select **Save**.
8. Restart the storage agent so that the new folders take effect:

```
Restart-Service CygnaDRStorageAgent
```

The agent scans the media folder. It reads the image information inside each ISO without mounting it.

The media folder can also be a network share given as a UNC path. A local folder is usually the better choice: a share depends on a network, a server, and a credential that the same disaster may take away.

How images are built

The storage agent builds the images the Manager asks for: one for each operating system variant found in restore points still held in the vault. Images are therefore built only after domain controllers have been backed up at least once, and only for variants that the media folder covers. The agent checks for missing or out-of-date images whenever that requirement changes and on a schedule (every 24 hours by default), and builds what is missing. A new build is added beside existing images. The Manager then decides which images each library keeps: per operating system variant, the newest image and the newest image proven by a completed restore, and never an image a restore is using. The storage agent deletes the rest. For the full rule, see Recovery Images.

When you upgrade the storage agent and the recovery agent it ships changes, the storage agent builds new images.

A new image is untested until a full system recovery or a test restore completes with it. If you have set up an Isolated Recovery Host and the **Prove new recovery images** setting is on (the default), untested images are test-restored automatically. See Setting Up an Isolated Recovery Host.

If you set up OS media before the first backups run, expect the images to appear shortly after the first backups complete.

Check the results

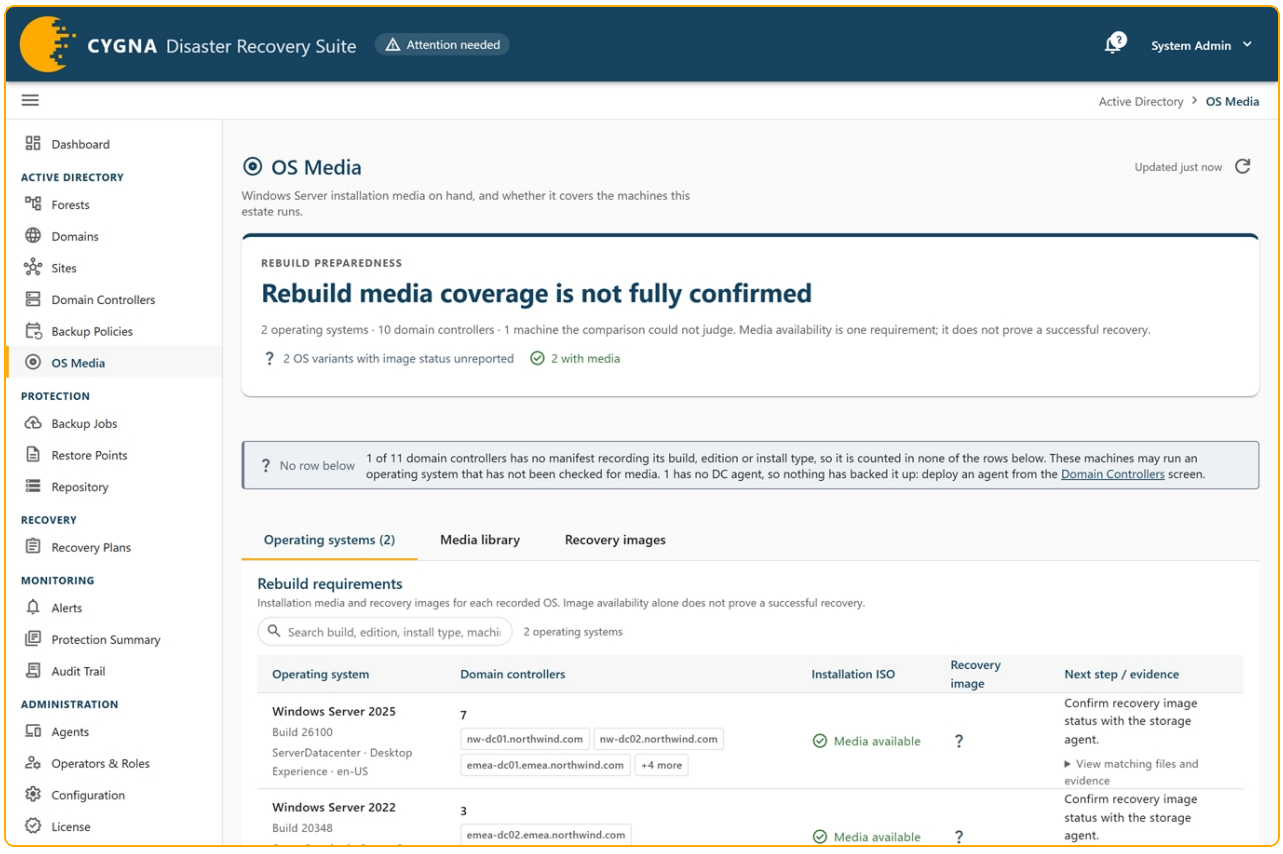
To check media coverage and recovery images:

1. Select **Active Directory** > **OS Media**.
2. Select the **Media library** tab.

   The **Media folders** card shows your folder, with **Read successfully** under **Last scan**. The **ISO files** card lists each ISO; any that show **Could not be read** should be checked against their published hash and replaced.
3. Select the **Operating systems** tab.

   Each operating system variant your domain controllers run is listed. The **Installation ISO** column shows **Media available** where an ISO matches. The **Recovery image** column shows whether domain controllers on that variant can use full system recovery.
4. Select the **Recovery images** tab.

   Each storage agent's library lists the images it holds, with each image's verdict and what the Manager decided for it under **Retention**, for example **Kept, newest**.



An **Available** value in the **Recovery image** column means full system recovery is possible. For what each **Unavailable** value means and what to do about it, and for a full description

of the screen, see OS Media. For the other OS media and recovery image settings, see Settings Reference.

Creating Your First Backup Policies

No backup policy exists by default, so nothing is backed up until you create one. A backup policy belongs to one domain controller and sets how often it is backed up and with which backup type. This topic shows how to protect the domain controllers you choose with one bulk edit. You can refine individual policies afterwards.

Before you begin

- Deploy the DC agent to your domain controllers and set up the storage machine. See Deploying DC Agents and Setting Up the Storage Machine.
- Make sure the Windows Server Backup feature is installed on the domain controllers, or that the agent is set to install it.
- To create policies, you need the `backup.manage` capability, which the **Backup Operator** role has.
- Decide which domain controllers to back up in each domain: at least two writable domain controllers, including the PDC emulator and the one you prefer to restore. See Choosing Which Domain Controllers to Back Up.

Choose the settings

- **Backup type**—Choose **Full system**. It captures critical volumes, including Windows, applications and DC state, and supports both DC state recovery and full system recovery. **DC state** backups are smaller and faster to take but support DC state recovery only, on a domain controller that still boots, so a domain controller whose hardware is lost would have to be rebuilt. See Backup Types.
- **Interval**—**Daily** is the default. Choose a shorter interval if you need recent restore points, and make sure your storage can hold the backups. Intervals range from 5 minutes to 90 days.
- **Start time**—**Any time** is the default: each policy runs at its interval, with a small random spread so that domain controllers do not all start at once. To keep backups outside business hours, choose **At a set time** and enter a time in the Manager's local time, such as 22:00. Each backup is due shortly after that time, never before, and then waits for a free backup slot, so with many domain controllers on one time the last start later. The editor estimates how late. See Managing Backup Policies.

Create policies for the domain controllers you back up

To create backup policies in bulk:

1. Select **Active Directory** > **Backup Policies**.

   Each domain controller is listed. Those without a policy show **None** in the **Schedule** column and are listed first.

2. Select the checkbox of each domain controller you chose to back up in each domain.
3. In the bar at the bottom of the screen, select the edit button, for example **Edit *n* schedules…**.

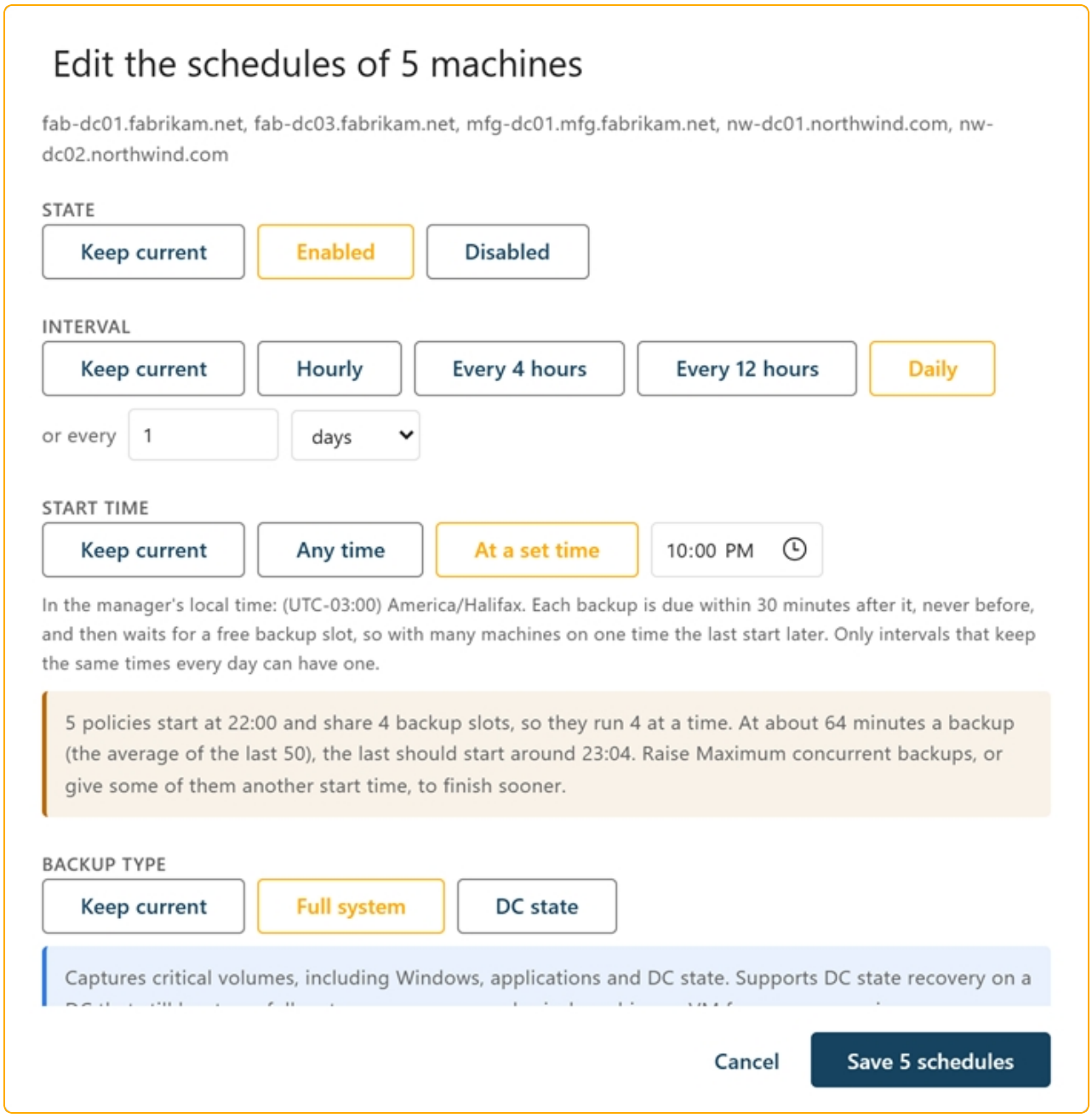
   The bulk editor opens.

4. In **State**, choose **Enabled**.
5. In **Interval**, choose **Daily** or another preset, or enter a custom interval.
6. In **Start time**, leave **Any time**, or choose **At a set time** and enter the time.

   If more domain controllers share the time than there are backup slots, the editor warns you how late the last should start.

7. In **Backup type**, choose **Full system**.
8. Select **Save *n* schedules**.

   Each new policy's first backup is due immediately, or at the start time if you set one. For a domain controller whose DC agent has not come online yet, the policy waits, shown as **Waiting for agent**, and its first backup runs as soon as the agent connects.



Backups do not all run at once. The Manager runs a limited number at a time, set by **Maximum concurrent backups** on the **Configuration** screen, and queues the rest, so the first round can take a while in a large environment.

Domain controllers you don't back up need no further action. With no enabled policy and nothing in the vault, a domain controller has the **Rebuilt on recovery** verdict: information, not a finding, and it raises no alert. See Backup Targets and Rebuilt Domain Controllers.

Run a backup now

To back up a single domain controller without waiting for its policy, select **Back up now** (play icon) on its row on the **Backup Policies** screen. The backup is of the type its policy takes. See Running a Backup Now.

Confirm the first backups

To confirm that the first backups completed:

1. Select **Protection** > **Backup Jobs**.

   A job is listed for each domain controller you scheduled. Watch the jobs until they complete. If a job fails, the job shows why; see Backup Job Errors.

2. Select **Protection** > **Restore Points**.

   A restore point is listed for each completed backup.

3. Select **Dashboard**.

   The dashboard shows whether each domain and forest is covered.

Once restore points exist, the storage agent can build recovery images for the operating systems they record. Check the **OS Media** screen after the first backups. See Setting Up OS Media and Recovery Images.

For changing, disabling, and removing policies, see Managing Backup Policies.

Setting Up an Isolated Recovery Host

An Isolated Recovery Host is a Hyper-V host that you dedicate to Cygna Forest Recovery. The product uses it for test restores: it restores a domain controller's backup into a new virtual machine on the host, starts the restored system on a network that reaches nothing, checks that it comes up as a working domain controller, and then deletes the virtual machine. For how a test restore works, see Test Restores.

The host is optional. Backup and recovery never depend on it. Without one, everything else works the same, and the console shows your backups as not verified.

The host runs the Isolated Recovery Host agent, which you install by hand. The agent checks the host, installs Hyper-V if it is missing, creates the network that test restore virtual machines use, and reports to the Manager whether the host is ready.

Before you begin

- Choose a machine for the host. A dedicated machine is recommended. For its sizing and all other requirements, see Isolated Recovery Host (optional) in System

Requirements.

- Install the current Windows updates on the machine, and keep it current afterward. A host that is behind on updates can fail to start a restored domain controller that is more current than the host. See Keeping the host current.
- Make sure the machine is not joined to a domain and is not a domain controller. Its disks briefly hold copies of the directory database, so no domain account may have rights on it.
- If the machine is a virtual machine, turn on nested virtualization for it. If it is a physical machine, turn on processor virtualization (Intel VT-x or AMD-V) in its firmware.
- Install the Manager and the storage agent, and deploy the DC agents. A test restore needs backups to restore and recovery images to start from. See Setting Up OS Media and Recovery Images.
- Have the enrollment token the Manager was installed with.
- Sign in to the host with a local administrator account.

Dedicated or shared machine

A dedicated machine is recommended, but installing the host agent on the storage machine or on the Manager machine is supported. A machine that runs the host agent or the storage agent is never offered as a candidate machine for a full system recovery, even if a DC agent is installed on it. A test restore builds its own virtual machine, of the backup's own generation, on the host. Consider what sharing costs before you choose it:

- A restored domain controller runs on the shared machine. If code in a restored backup ever escaped its virtual machine, it would land on the machine that holds every backup, or beside the Manager's certificate authority key and database.
- Test restores compete with backups for disk and network on a shared storage machine.
- If the storage machine or the Manager is itself a virtual machine, Hyper-V needs nested virtualization there.
- The agent does not restart a shared machine to finish installing Hyper-V, because the restart would also stop the storage agent or the Manager. You restart it yourself at a time that suits you.

The host can never be a domain controller or share a machine with one.

Allow the host to reach the storage machine

During a test restore, the virtual machine reads the backup from the storage machine over SMB and reports to the Manager. It reaches both through the host, so the connections

come from the host's IP address. If you limit inbound TCP 445 on the storage machine to your domain controllers' networks, add the host's address, for example:

```
New-NetFirewallRule -DisplayName "Cygna DR Isolated Recovery Host (SMB)" -Direction Inbound -Protocol TCP -LocalPort 445 -RemoteAddress 10.0.0.30 -Action Allow
```

The host itself needs outbound access to the Manager on TCP 5211. The host agent installer creates no firewall rules, and the agent needs no inbound connections. For the full list, see Ports and Protocols.

Install the Isolated Recovery Host agent

Install the host agent by hand, with the command line the console builds for you. The Manager carries a copy of the package, `CygnaDR.HostAgent.msi`, which you can download from the Agents screen.

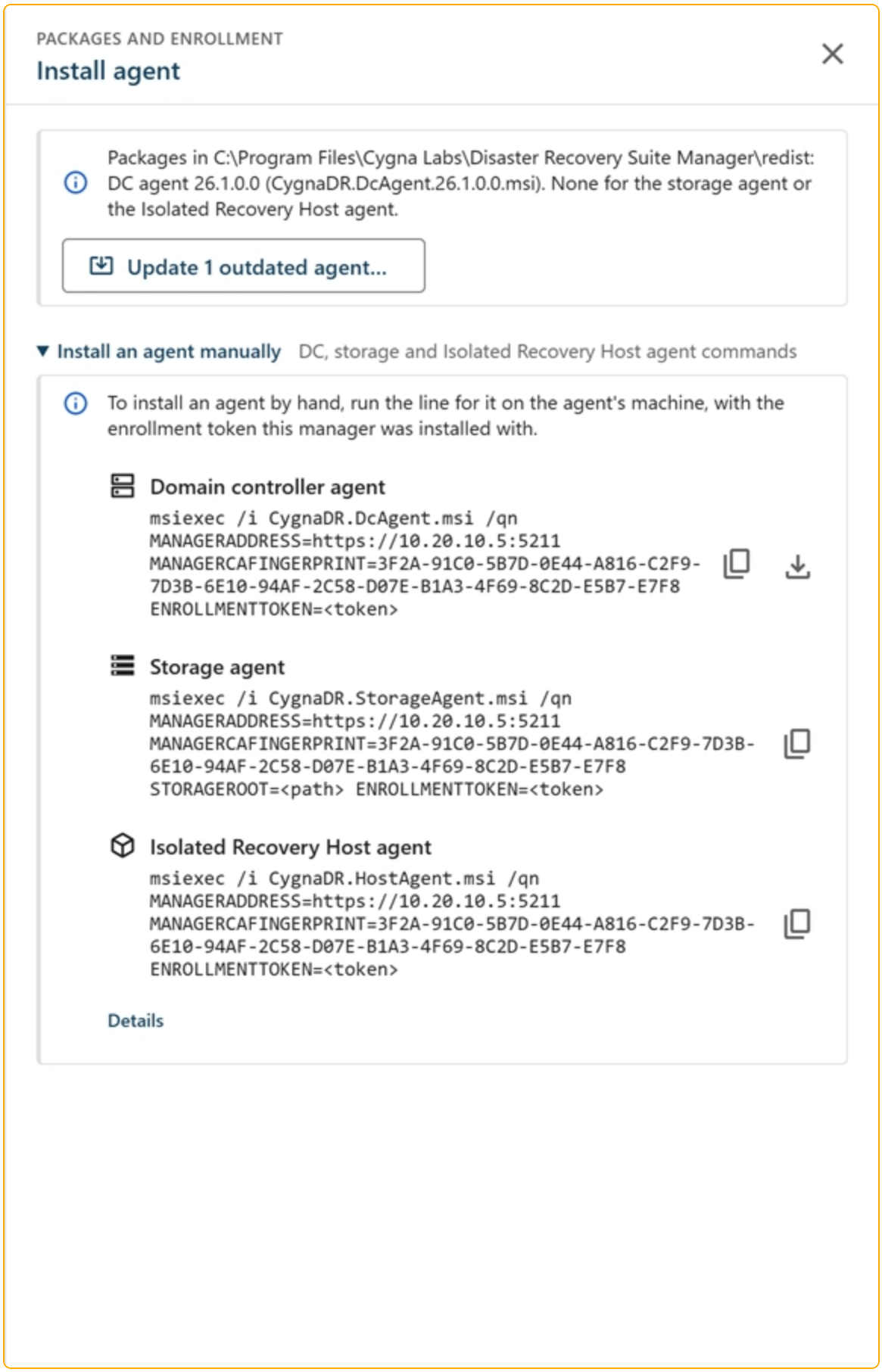
To install the Isolated Recovery Host agent:

1. In the console, select **Administration** > **Agents**, and then select **Install agent** in the page header.
2. On the Isolated Recovery Host agent line, select the download button to save `CygnaDR.HostAgent.msi`, and copy it to the host.

   Downloading a package requires the `agents.manage` capability. You can also save it with `cdr agent package download host`.
3. Select **Copy command** on the same line.
4. On the host, open an elevated command prompt in the folder that holds the package. Paste the command, replace `<token>` with the enrollment token, and run it. The command looks like this:

   ```
   msiexec /i CygnaDR.HostAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211 MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
   ```

   The installer installs and starts the `CygnaDRHostAgent` service. The agent connects to the Manager and enrolls. It enrolls awaiting approval, and is given no test restores until you approve it.


PACKAGES AND ENROLLMENT
Install agent
Packages in C:\Program Files\Cygna Labs\Disaster Recovery Suite Manager\redist: DC agent 26.1.0.0 (CygnaDR.DcAgent.26.1.0.0.msi). None for the storage agent or the Isolated Recovery Host agent.
Update 1 outdated agent...
Install an agent manually
DC, storage and Isolated Recovery Host agent commands
To install an agent by hand, run the line for it on the agent's machine, with the enrollment token this manager was installed with.
Domain controller agent
msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 ENROLLMENTTOKEN=<token>
Storage agent
msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 STORAGEROOT=<path> ENROLLMENTTOKEN=<token>
Isolated Recovery Host agent
msiexec /i CygnaDR.HostAgent.msi /qn MANAGERADDRESS=https://10.20.10.5:5211 MANAGERCAFINGERPRINT=3F2A-91C0-5B7D-0E44-A816-C2F9-7D3B-6E10-94AF-2C58-D07E-B1A3-4F69-8C2D-E5B7-E7F8 ENROLLMENTTOKEN=<token>
Details

The package takes no other properties. Use the Manager's IP address in `MANAGERADDRESS`, as for the other agents. The installer keeps the enrollment token out of its log.

The program files go to `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Isolated Recovery Host Agent`, and the agent's state and logs to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Isolated Recovery Host Agent`.

What the agent does after it enrolls

The agent waits for its configuration from the Manager and then prepares the host. Each step is controlled by a setting on the **Isolated Recovery Host agents** tab of the Configuration screen, and each is on by default.

- **Installs Hyper-V**—If Hyper-V or its PowerShell module is missing, the agent installs the Hyper-V role with its management tools. Turn off **Install Hyper-V** to install it yourself instead. When the agent leaves a missing Hyper-V alone, for example because the setting is off or the processor cannot virtualize, its service log says why, once when the service starts and again whenever the reason changes.
- **Restarts the host to finish**—Hyper-V needs a restart to finish installing. A host where Windows reports the role installed but the Hyper-V service has not appeared yet counts as waiting for that restart. When nothing else of Cygna Forest Recovery's runs on the host, the agent asks Windows to restart it five minutes later, once per install. To cancel the restart, run `shutdown /a` on the host within those five minutes. On a shared host, or with **Restart to finish installing** turned off, the agent reports that a restart is needed and why it did not make one.
- **Creates the recovery network**—The agent creates an internal Hyper-V switch named **CygnaDR Recovery**, gives the host the first address of the **Recovery network prefix** (by default `172.31.250.0/24`) on it, and puts Windows NAT behind it. Test restore virtual machines reach the Manager and the storage machine through the host, and nothing on your network can reach them. The agent never creates an external switch, and it refuses a prefix that overlaps an address or a NAT the host already has.
- **Chooses a folder for virtual machines**—Unless you set **Folder for verification VMs**, the agent uses a `CygnaDR\Recovery VMs` folder on the data drive with the most free space. Hyper-V's own default folders are not changed.

Choose a recovery network prefix that is not used anywhere the host or the virtual machines need to reach. While it restores, each test restore's virtual machine gets a static address in this network and the host's own DNS servers, so the host must have DNS servers configured. Once isolated, the restored domain controller uses itself for DNS.

Check that the host is ready

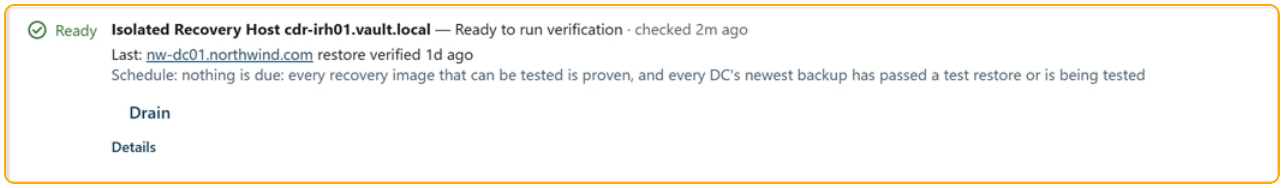
The agent checks the host when it connects and every 15 minutes after that, and reports a verdict: **Ready**, **Not ready**, or **Unknown**.

To check the host:

1. In the console, select **Administration** > **Agents**, and then select the **Recovery hosts** tab.

   A panel for the host shows its verdict and a headline, such as **Ready to run verification**, followed by any problem the host found and what to do about it.

2. If the host is not ready, follow the instructions in the panel, for example restarting the host to finish the Hyper-V install. The host reports again on its own.



From the command line, run `cdr recovery-host list`. For what each check means, see Agents.

Approve the host

A host is given test restores only once an operator approves it, so that a machine nobody meant to enroll never reads your backups. Until then, its panel and its row on the **Agents** screen read **Awaiting approval**, and the notice **Isolated Recovery Hosts awaiting approval** is raised. Approving needs the `agents.manage` capability.

To approve the host:

1. In the console, select **Administration** > **Agents**, and then select the **Recovery hosts** tab.
2. In the host's panel, select **Approve…**.
3. Check that the dialog names the host you installed, and select **Approve the host**.

From the command line, run `cdr recovery-host approve` with the host agent's ID, which `cdr recovery-host list` shows under the host.

When the host is approved and ready, scheduled test restores start on their own. To change when they run, see the **Test restores** settings in Settings Reference. You can approve more than one host; test restores go to whichever approved host can run one.

Keeping the host current

Keep the Isolated Recovery Host on current Windows updates. This is a requirement, not general good practice: without it, test restores of up-to-date domain controllers fail, and so would a real recovery onto that host. The same applies to any Hyper-V host you recover a domain controller onto as a new virtual machine.

Microsoft is replacing the Secure Boot certificates that Windows has started with since 2011. As a domain controller takes its Windows updates, its boot files move to the new *Windows UEFI CA 2023* certificate. A backup holds the boot files, but not the list of certificates the machine's firmware trusts; for a Hyper-V virtual machine, that list comes from the host. A host that is behind on updates gives new virtual machines a list without the 2023 certificate, so the restored domain controller is refused at its first start.

The restore itself completes; only the first start of the restored system fails:

- The test restore restores the domain controller and then ends with no verdict, because the restored domain controller reports nothing.
- The virtual machine's console shows **Virtual Machine Boot Summary**, with *The image's hash and certificate are not allowed (DB)* against the system disk.
- The host logs event 18604 in **Microsoft-Windows-Hyper-V-Worker-Admin**, saying that at least one boot source failed Secure Boot validation.

To fix it, install the current cumulative update for the host's Windows Server version, restart the host, and then run a new test restore. Do not turn Secure Boot off on the virtual machine to get past it: the restored domain controller would then start on a machine set up differently from the original, and the test would not show that the host can recover it as it is.

Uninstalling the host agent

Before you uninstall the host agent, drain the host on the **Agents** screen or with `cdr recovery-host drain`, so that no test restore is cut short, and wait until it reads **Drained**. To move test restores to another host, see Moving an Isolated Recovery Host.

Uninstall the host agent from **Apps** in Windows Settings, or with `msiexec /x CygnaDR.HostAgent.msi /qn`. The uninstall removes the service and the program folder, and leaves the data folder. Hyper-V, the **CygnaDR Recovery** switch, its address, and the NAT named `CygnaDRRecovery` stay on the host; remove them yourself if you no longer need them, for example with `Remove-NetNat -Name CygnaDRRecovery` and `Remove-VMSwitch -Name "CygnaDR Recovery"`.

Upgrading and Uninstalling

Each Cygna Forest Recovery package upgrades in place. Running the installer over an existing installation stops the service, replaces the program, and starts it again. Data, configuration, and enrollment are kept. Uninstalling removes the program but deliberately leaves the data behind.

Upgrade order

Upgrade the Manager first, then the agents:

1. **Manager**–The Manager package carries the DC agent, storage agent, and Isolated Recovery Host agent packages of the same release. Upgrading the Manager replaces the agent builds it holds, so the console can then flag every agent that is behind and update it.
2. **Storage agent**–Update it from the console or with `cdr agent update`, between backups. It then rebuilds its recovery images with the new recovery agent, so full system recoveries run the current release.
3. **DC agents**–Update them from the console or with `cdr agent update --outdated`, before any recovery plan runs.
4. **Isolated Recovery Host agents**–If you have any, update them the same way, when no test restore is running on their host.

Agents of every kind update themselves from the Manager, with no credential. Update all outdated agents in one run with **Update *n* outdated agents…** on the **Agents** screen or `cdr agent update --outdated`. An agent that cannot update itself is upgraded by hand or, if it is a DC agent, pushed to. Use packages from the same release for every component.

Every release file and package is signed by Cygna Labs Corp. To check a package before you install it, run `Get-AuthenticodeSignature` on it in PowerShell and expect the status `Valid` with `O=Cygna Labs Corp.` in the signer, or open the file's **Properties** > **Digital Signatures** tab.

Note: **Important:** The Manager does not send a recovery plan step to a DC agent whose build is behind its package, and every Manager upgrade puts every DC agent behind, even when the agent has not changed. Update the DC agents before any recovery plan runs, and avoid upgrading the Manager while a recovery is under way. If a step is refused, its message names both builds; update the agent on that domain controller, then retry the step. See Maintaining Agents.

Upgrade the Manager

Before you upgrade, take a database snapshot so that you have a copy of the database from just before the change. See Protecting the Manager.

```
"%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager\CygnaDR.Manager.exe" db snap-shot
```

To upgrade the Manager:

1. Copy the new `CygnaDR.Manager.msi` to the Manager machine, and open an elevated command prompt in that folder.

2. Run the installer. No properties are needed: the installer keeps the enrollment token, ports, listen address, and redirect port the Manager was installed with.

   ```
   msiexec /i CygnaDR.Manager.msi /qn
   ```

   A property you do give replaces the kept value. Give `RESTPORT`, `GRPCPORT`, or `LISTENADDRESS` only to move the Manager: agents and operators go on using the old values until you change them.

3. Reload the console in your browser.

   The agents reconnect on their own.

The enrollment token and `REDIRECTPORT` are kept from the previous installation. A value you give on the command line replaces the stored one; for example, giving `ENROLLMENTTOKEN` on an upgrade changes the token.

Update the agents

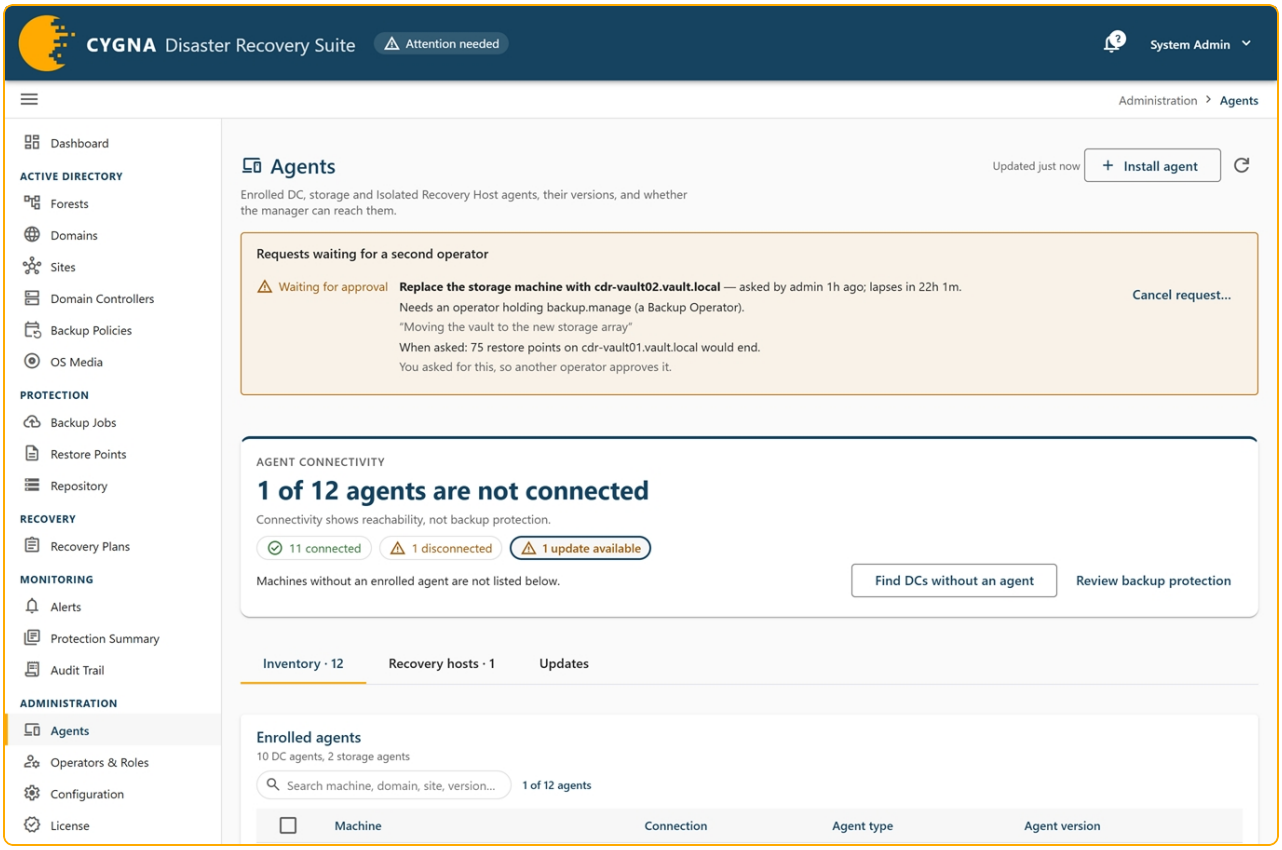
The **Agents** screen compares each agent with the package for its kind in the Manager's package folder and marks agents that are behind with an update icon. The ***n* updates available** filter above the table shows only those agents. With **Update agents automatically** on, the Manager updates them itself as each one can take an update, and you need only deal with those it lists as needing an operator. See Updating agents automatically.

To update agents from the console:

1. Select **Administration** > **Agents**.
2. Start the update in one of these ways:
   - To update every outdated agent, select **Update *n* outdated agents…** beside the package line on the **Updates** tab.
   - To update one agent, open its row's actions menu (⋮) and select **Update the agent on *machine* to *version*…**.
   - To update a selection, select the rows' checkboxes, and then select **Update *n* agents…** (**Update 1 agent…** for one).
3. Start the update.

   Agents that can update themselves need nothing more. Each receives the package over its own connection, checks that the package is signed by Cygna Labs Corp., is its own package, and is newer, and then installs it and reports back. If any target is a domain controller that must be pushed to, the dialog also asks for an administrator account and password and the transport, as for a first installation. See Deploying DC Agents. An update keeps each agent's Manager address.

The Manager does not update an agent while it has work in flight, such as a backup or restore on that machine. For details, see Maintaining Agents.



With the `cdr` command-line tool, run this from the Manager's program folder after signing in with `cdr login`:

```
cdr agent update --outdated
```

Add `--user CORP\admin` to include domain controllers whose agent cannot update itself and must be pushed to.

Any agent can also be upgraded by hand, by running its package on the machine in an elevated command prompt. This is the only way to upgrade a storage agent or Isolated Recovery Host agent that cannot update itself. Download the package from the **Agents** screen or with `cdr agent package download`, or copy it from the `redist` folder in the Manager's program folder:

```
msiexec /i CygnaDR.DcAgent.msi /qn
msiexec /i CygnaDR.StorageAgent.msi /qn
msiexec /i CygnaDR.HostAgent.msi /qn
```

The Manager address, CA fingerprint, enrollment token, and, for the storage agent, the storage root are kept, so no properties are needed. To change one of them, give it on the command line.

The machine-local policy that turns off updates from the Manager, under `HKLM\SOFTWARE\Policies\Cygna Labs\Disaster Recovery Suite`, is not touched by an upgrade or an uninstall. See Maintaining Agents.

Note: Windows Installer refuses to install an older release over a newer one, with a message that a newer version is already installed. To go back to an older release, uninstall first.

Uninstall

To uninstall a component, run its package with `/x` in an elevated command prompt on the machine:

```
msiexec /x CygnaDR.DcAgent.msi /qn
msiexec /x CygnaDR.StorageAgent.msi /qn
msiexec /x CygnaDR.HostAgent.msi /qn
msiexec /x CygnaDR.Manager.msi /qn
```

A connected agent can also be uninstalled from the Manager, with **Uninstall the agent from *machine*...** on the **Agents** screen or `cdr agent uninstall`; see Maintaining Agents. You can also remove Cygna Disaster Recovery DC Agent, Cygna Disaster Recovery Storage Agent, Cygna Disaster Recovery Isolated Recovery Host Agent, or Cygna Disaster Recovery Manager from **Programs and Features** in Control Panel. Their services keep the short names, such as Cygna DR DC Agent.

Uninstalling removes the service, the program folder, the install-time registry settings, and, for the Manager, its firewall rules. It leaves these behind:

What remains	Details
Manager data folder	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager`: the database, which includes the Manager's CA, operator accounts, inventory, and restore point index; the database snapshots; and the logs.
Agent data folders	`%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent`, `%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent`, and `%ProgramData%\Cygna Labs\Disaster Recovery Suite Isolated Recovery Host Agent`: agent state, backup job transcripts, and logs.
Vault and staging	Everything under the storage root, including every backup in the vault. The installer never touches the storage root.

What remains	Details
Staging share and firewall rule	The SMB share and the inbound TCP 445 rule you created on the storage machine.
Trusted certificates	The Manager CA certificate you imported on workstations, and the Managers that `cdr` users have trusted.

These are kept on purpose: an upgrade is a remove-then-install, and the data must survive it. If you are removing Cygna Forest Recovery for good, decide what to keep before you delete anything. The vault holds your backups, and the Manager database is the only index of them.

Note: **Important:** Deleting the vault folder permanently deletes every backup in it. Deleting the Manager data folder permanently deletes the Manager's CA and its record of every restore point.

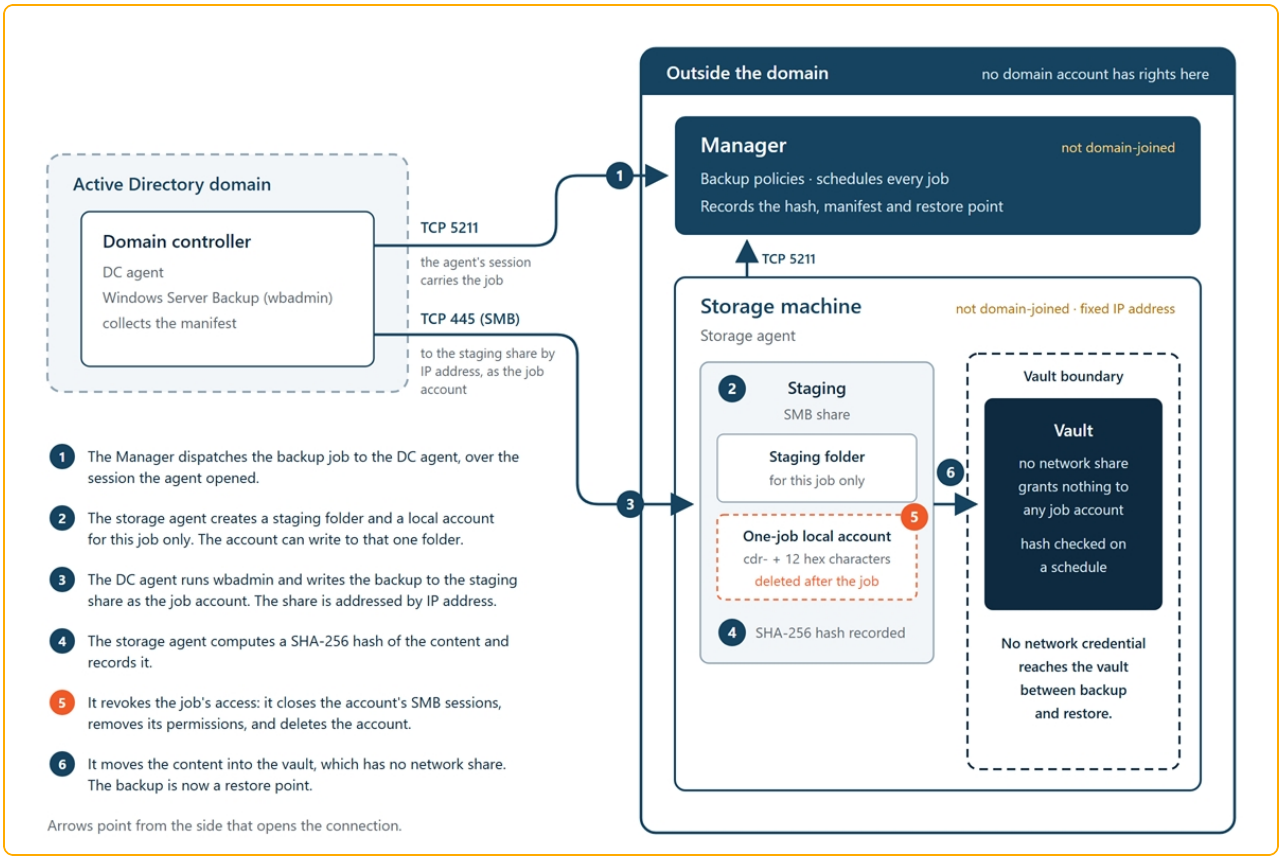
How Backup Works

Cygna Forest Recovery backs up each domain controller on a schedule you set, and moves every completed backup into a vault on a machine that is not joined to the domain. This chapter explains how a backup travels from the domain controller to the vault, what is recorded about it, how it is protected while it is stored, and how long it is kept.

The path of a backup

Every backup follows the same sequence. The order of the steps is what keeps a backup out of reach of the directory it protects.

1. **The Manager starts the job.** Each domain controller has a backup policy with an interval and a backup type. When a policy falls due, the Manager dispatches the job. Agents never start a backup on their own. A policy can also have a start time in the Manager's local time, which is the earliest each backup is due. A small random spread, around the interval or only after the start time, keeps a large estate from backing up all at once, and the Manager limits how many backups run at the same time.
2. **The storage agent prepares a staging folder and a one-time account.** The storage agent reserves room in the vault for the incoming backup, creates a staging folder for this job only, and creates a local account on the storage machine for this job only. The account can write to that one folder and nothing else.
3. **The domain controller writes the backup to staging.** The DC agent runs Windows Server Backup (`wbadmin`) against the staging share, using the per-job account. The staging share is addressed by the storage machine's IP address, not by its name, so a poisoned DNS record cannot send the backup to another machine. While the backup runs, the DC agent also records what it needs to know about the machine: its operating system, hardware, network settings, and directory role. That record becomes the backup's manifest.
4. **The storage agent hashes the content.** When the job finishes, the storage agent computes a SHA-256 hash of the backup. The hash is recorded with the manifest and in the Manager's database, and is used later to prove that the backup has not changed.
5. **The job's access is revoked.** The storage agent closes the account's network sessions, removes every permission the account held, and then deletes the account.
6. **The backup moves into the vault.** The storage agent moves the content into the vault, which has no network share and grants nothing to any job account. It then checks that no permission for the deleted account came along with the content. The backup is now a restore point.



Every backup is a full backup. There are no incremental or differential backups, so each restore point stands on its own and does not depend on any other.

Why this design is safe

- **No domain account has rights to backup storage**–The storage machine is not joined to the domain, so domain credentials, including Domain Admins and Enterprise Admins, have no access to it.
- **Each write credential lives only as long as its job**–An account is created for one job, can reach one folder, and is deleted when the job ends. No account ever covers two backups.
- **Nothing on the network can reach the vault**–Between a backup and a restore, no credential exists that can read, change, or delete a backup over the network. A restore gets its own short-lived, read-only access to the one backup it needs.
- **Nothing on the network can delete a backup**–Backups are deleted only by the retention rules that run on the storage machine itself.
- **Changes are detectable**–Each backup's hash is checked again on a schedule and before it is restored.

- **Backups can be proven**–With an optional Isolated Recovery Host, backups are test-restored onto isolated virtual machines and checked as working domain controllers, on a schedule and on demand. See Test Restores.

Who does what

Component	Role in backup
Manager	Holds the backup policies and schedules every job. Records each job's outcome, its manifest, and its restore point. Never touches the vault directly.
DC agent	Runs `wbadmin` on the domain controller, writes the backup to its staging folder, collects the manifest, and captures the backup's version identifier.
Storage agent	Owns staging and the vault on the storage machine. Creates and deletes the per-job accounts, hashes and commits backups, verifies stored backups, applies retention, and reports vault capacity.

Operators with the Backup Operator role create and change backup policies and start backups on demand. For the tasks, see Backup Tasks.

In this chapter

- Backup Types
- Restore Points
- Vault Protection and Integrity
- Retention
- Recovery Images
- Test Restores
- Coverage and Readiness

Backup Types

Each backup policy sets one of two backup types. The type decides which kinds of restore the backup can support later, so choose it with recovery in mind rather than storage alone.

Full system

A full system backup (shown in the console as **Full system**) captures every critical volume of the domain controller: the operating system volume and every volume that holds the directory database, its logs, or SYSVOL. That includes Windows, applications, and DC state. It is taken with `wbadmin start backup -allCritical`. A full system backup supports both kinds of restore:

- **DC state recovery**—Puts Active Directory back on a domain controller whose Windows still starts.
- **Full system recovery**—Puts the whole server back, operating system included, on a physical machine or VM from a recovery image: either the same machine or a replacement machine.

Full system is the default type for a new backup policy.

DC state

A DC state backup (**DC state**) captures Active Directory and the domain controller's Windows system state: the directory database, SYSVOL, the registry, and the boot files. It is smaller and faster than a full system backup, but it supports DC state recovery only, on a domain controller that still boots. It cannot restore a server whose operating system is gone, and it cannot be restored onto another machine.

Comparing the types

	Full system	DC state
DC state recovery	Yes	Yes
Full system recovery, in place or onto a candidate	Yes	No
First restored DC in a forest or domain recovery	Yes, by either method	Yes, by DC state recovery in place only
Relative size	Larger: the used space on every critical volume	Smaller
Counts toward a domain's full system recovery coverage	Yes	No

A full system backup is roughly the used space on the machine's critical volumes, less the page file, hibernation file, and shadow copy storage, plus some image overhead. To estimate the storage your estate needs before you install, see Sizing Backup Storage.

Choosing a type

- **Use full system for the domain controllers you would restore first**—A lost or encrypted machine needs a full system recovery, and only a full system backup provides one. A domain whose writable domain controllers have only DC state backups shows **No full system recovery** on the **Domains** screen.
- **Cover at least two writable domain controllers in each domain**—The coverage rule is two full system backups per domain, so that one bad or post-compromise backup still leaves another to restore from. A domain with only one shows **Single recovery option**.
- **Consider DC state where storage is tight and full system recovery is not needed**—For example, for additional domain controllers that you would rebuild by promotion rather than restore if they were lost.



Note: A DC state recovery can only put a backup back onto the same Windows build and revision the machine runs now. After Windows Update patches a domain controller, its older backups can no longer be restored onto it this way until a new backup is taken. Back up soon after patching. A full system recovery is not affected, because it restores the operating system along with the directory.

For how each type is used during recovery, see Restore Methods.

Read-only domain controllers

A read-only domain controller (RODC) is never restored from backup. It is rebuilt instead: its old metadata is removed and a fresh RODC is promoted. A forest or domain recovery plan can only mark an RODC for rebuild or exclusion, and a single-DC restore of an RODC is refused. A backup of an RODC therefore cannot start or contribute to a recovery, and coverage ignores it: a domain whose only restorable backups are of RODCs is reported as **Unrecoverable**.

Install the DC agent on RODCs anyway, so that the inventory, coverage figures, and recovery plans include them.

Restore Points

A restore point is one committed backup of one domain controller, together with the record of what that backup captured. Every backup that reaches the vault becomes a restore point. Recovery plans are built from restore points: you choose one for each domain controller you restore, and preflight checks it against the machine it will be restored onto.

The manifest

The record that travels with each backup is its manifest. The DC agent collects it while the backup runs, and the storage agent stores it beside the backup in the vault. The Manager

keeps a copy in its database. The manifest is covered by the backup's hash, so a manifest that has been changed is detectable.

The manifest describes the machine as it was at backup time, which is what a restore would put back:

Group	What it records	Used for
Identity	FQDN, NetBIOS name, domain, forest, site, domain SID, DC object GUID	Matching the backup to its machine, domain, and forest
Operating system	Version, build, update build revision (UBR), edition, install type (Desktop Experience or Server Core), language, install volume	The build check for DC state recovery; choosing the recovery image and installation media
Active Directory role	Global catalog, read-only flag, SYSVOL replication engine (DFSR or FRS), operations master roles held	Deciding whether the DC can be restored and in what order
Backup	Version identifier, backup type, vault path, committed size, content SHA-256, DC agent version, start and completion times	Addressing the restore and proving the content is unchanged
Hardware	Platform and model, firmware (UEFI or BIOS), Secure Boot, TPM, memory, processors, each disk's size and partition style, volumes and their BitLocker status	Full system recovery checks against the target machine; shaping a replacement machine

Group	What it records	Used for
Network	Each adapter's MAC address, DHCP or static addresses, gateways, DNS servers in order; persistent routes; primary DNS suffix	Putting the original address back after a full system recovery; isolation planning
Roles, features and programs	The Windows roles and features installed, and the installed programs with their versions and publishers	Reinstalling a rebuilt domain controller's roles and features before it is promoted; listing in the recovery report what a rebuild does not bring back: the roles' configuration and the programs

Manifests also record other facts that preflight relies on, such as server roles running beside Active Directory on the domain controller.

The version identifier

Windows Server Backup gives each backup a version identifier, and a restore is addressed by it. The identifier is written in the format of the domain controller's own locale. For example, a backup taken at the same moment reads `09/02/2026-23:00` on a US English domain controller and `02.09.2026-23:00` on a German one.

Cygna Forest Recovery reads the identifier from `wbadmin` after each backup, confirms it by asking `wbadmin` to accept it, and stores and shows it exactly as `wbadmin` wrote it. It is never reformatted, so do not read it as a date in your own locale. The console uses the backup's recorded UTC times for ordering and age.

If the identifier could not be captured, the backup is still committed and kept, but preflight refuses to restore from it. Choose another restore point.

Restore points that cannot be used

A restore point's record is kept when it can no longer be restored, so that the record of what was backed up is not lost. The console says why it cannot be used:

- **Pruned**–Retention has deleted the backup content. The manifest is kept as a record and the restore point detail shows **Backup pruned**, but there is nothing left to restore. Because a pruned backup is not a restore point, the **Restore Points** screen and `cdr backup list` hide pruned backups and count them; you can show them as history.

How long these records are kept is set by the **Manifests** database retention setting, which keeps them indefinitely by default.

- **Older than the tombstone lifetime**—Restoring it would reintroduce deleted objects. Preflight refuses it.
- **Blocked for the chosen method**—For example, a DC state backup cannot be used for a full system recovery, and a backup taken before the machine was last patched cannot be used for a DC state recovery. The restore point picker shows these points, but they cannot be selected, and the reason is shown beside each.

A backup taken while a recovery of its forest was under way can be restored, but it holds the directory part-way through that recovery: operations master roles just seized, domain controllers removed and not yet rebuilt, and initial synchronization turned off. The console tags it **During recovery**, `cdr backup list` marks it **during recovery**, and restore preflight raises an advisory gate that names the plan and the step it had reached. Restore it only if that is the state you mean to return to.

The **Restore Points** screen lists the newest restore points across the estate. To see the full history of one domain controller, open its restore points from the machine's cross-links or filter the screen to that machine.

Verified restore points

A full system restore point can be proven by a test restore on an Isolated Recovery Host, which restores it onto an isolated virtual machine and checks it as a working domain controller. The result is recorded against that one version: verified or failed. A version that was never tested, or whose test ended without a verdict (inconclusive), is unverified. A newer backup of the same domain controller starts unverified.

The result is shown in the **Test restore** column of the Restore Points Screen screen and on each restore point's page, next to each restore point when you plan a single-DC restore, and in `cdr backup list`. For more information, see Test Restores.

Backups taken before a CA re-key

Each backup records which Manager certificate authority its DC agent trusted at the time. If you later re-key the Manager CA, a restore of an older backup brings back a DC agent that still trusts the old CA and refuses to connect to the Manager until it is re-pointed.

Preflight detects this. When a restore would bring back an agent that trusts a different CA, the plan shows it as a finding. It blocks approval if the plan includes network settings for the restored system, because the restored agent would refuse the Manager's signed instructions. Otherwise it is advisory: the directory is restored, and you re-point the agent as soon as the machine is up. Take fresh backups of every domain controller you back up after a re-key. For the re-key procedure, see Re-Keying the Manager CA.

Related topics

- Restore Points Screen
- Retention
- Vault Protection and Integrity

Vault Protection and Integrity

The vault is the folder on the storage machine that holds every committed backup. It is designed so that an attacker who holds every domain credential still cannot read, change, or delete a backup over the network, and so that any change to a stored backup is detected before you rely on it. The storage agent also watches the vault and the staging share for activity that no backup job or retention rule explains, and the Manager raises an alert for each finding.

Who can reach the vault

The vault lives under the storage root on a machine that is not joined to the domain. It has no network share. Its permissions are protected from inheritance and grant access only to:

- The local system account, under which the storage agent runs.
- The local `CygnaDR Administrators` group on the storage machine.
- The local `BUILTIN\Administrators` group, unless you remove it (see Removing local administrators below).

The per-job accounts that write backups have no rights in the vault at any time. They are members of the `CygnaDR Staging Writers` group, which exists only so that the staging share can be granted once. Each account reaches only its own staging folder and is deleted before its backup is moved into the vault. For every local group the storage agent creates, see Accounts and Permissions.

The storage agent checks the staging share's permissions at startup and repairs what backups need, without removing entries it did not create. It reports anything else the share lets write, so that you can review it.

The storage machine also keeps the Manager's configuration backups, in `<StorageRoot>\manager-backups`. The folder has the vault's permissions and is never shared. The backups are encrypted, the storage agent cannot read them, and the Manager cannot delete them: the storage machine keeps at least its own minimum and prunes by its own rules. See Manager Configuration Backups.

No deletion from the network

No API, console action, or command deletes or overwrites a backup. The only way a backup is ever deleted is by the retention rules that run on the storage machine. The deepest of those rules, the retention floor and the minimum number of versions per domain controller, can be set only on the storage machine itself. The console can make retention stricter but cannot weaken it. For details, see Retention.

Read access for a restore

A restore needs to read one backup, so it is given a read lease on that backup only, for the length of the work:

1. The storage agent verifies the backup's hash (see Verification before a restore).
2. It creates a local account for this restore only, in its own group, and a read-only share over the single backup's folder. The vault root is never shared.
3. The machine being restored reads the backup through that share.
4. When the lease ends, the share, its permission, and the account are all removed.

For a DC state recovery, the domain controller copies the backup to its own disk while Windows is still running, and the lease is released before the machine restarts into Directory Services Restore Mode. The backup is exposed for minutes rather than for the whole restore. For a full system recovery, the recovery environment reads the backup over the share for the whole restore.

Every lease is recorded in the audit trail as **Vault read lease issued** and **Vault read lease revoked**, and a refusal as **Vault read refused**.

Scheduled hash verification

The SHA-256 hash recorded when a backup is committed is the evidence that it has not changed. The storage agent re-reads stored backups on a schedule and checks each one against its recorded hash. This catches silent corruption, such as a failing disk array, and deliberate tampering, such as an attacker encrypting the vault.

- Verification is on by default. Each pass reads a limited number of versions, then waits for the next interval, so that verification does not compete with incoming backups.
- Each pass checks the least recently verified versions first. A version that has never been verified goes ahead of every version that has.
- A version that passed recently is skipped until its result is older than the **Re-verify after** setting.
- A version that failed is re-read on every pass, so you can tell a one-off read error from progressive corruption.

- Verification only reads. It never deletes, repairs, or moves a backup. A corrupt backup is kept as evidence.

Each check has one of three outcomes: it passes, it fails because the content no longer matches its hash, or it could not be checked, for example because no hash was recorded or the content could not be read. The **Backup verification** tab of the **Repository** screen shows the results and lists every failure with when it started failing. A version whose latest result is a failure also raises the critical alert **Vault content whose hash does not match**. The pass size, interval, and re-verify window are storage agent settings; see Settings Reference.

Detecting changes to committed backups

After the commit, the only legitimate change to a backup is its deletion by retention. The storage agent therefore treats any other change to a committed backup as a finding: a delete, rename, write, truncation, or change of permissions or attributes, whether it affects one file or thousands. Ransomware and intruders typically go after backups first, and this is how such activity on the storage machine comes to light without waiting for the next verification pass or a restore.

- **How changes are seen**—The storage agent reads the change journal of the volume that holds the vault. The journal records every change on the volume whether or not anything is reading it at the time, so after a restart the storage agent resumes where it stopped and also reports changes made while it was not running. If the volume has no change journal, the storage agent creates one. Changes that the storage agent itself makes, when it commits a backup, applies retention, or sets permissions, are not reported. The record the journal writes when a file is closed repeats what was done to the file while it was open, so it is judged only on what it adds: a backup held open for a long restore does not produce a late finding when it is closed.
- **When the journal lost records**—If the journal was deleted, or records were overwritten before the storage agent read them, the storage agent compares every committed backup with its commit record for the stretch of time it could not read, and reports what the comparison finds.
- **Who made the change**—The storage agent puts an audit entry on each committed backup's folder, so that the Security log records deletes, writes, and permission changes to it. Whether anything is logged depends on the storage machine's audit policy, which Cygna Forest Recovery reads and never changes. With **File System** auditing on, a finding names the account and process; with **Detailed File Share** auditing on as well, a change made over SMB also names the client address. See Audit policy on the storage machine.

Each changed backup raises the critical alert **Committed vault content changed outside retention**, which names the backup, what was done, how many paths, and over how long. The first finding for each backup is also recorded in the audit trail. The alert stays raised until an operator with the `backup.manage` capability records a review with `cdr repository`

`changes review`, with a note saying what the review found. `cdr repository changes` lists the open changes with their paths and who made them. Detection is all that happens: nothing is blocked or reverted, and the changed content is left in place.

A storage machine that cannot watch its vault raises **Vaults not being watched for changes** with the reason, so that silence is never mistaken for safety. The change journal is a feature of NTFS and ReFS; a vault on another file system or on a network share cannot be watched. The storage agent must also run elevated, as the installed service does.

Watching the staging share

The staging share is the only part of the storage machine that domain controllers reach, and each backup job reaches it with a local account that exists for that job only. The storage agent knows which domain controller each account belongs to and how long it should exist, so it can tell when the share is used in a way no job explains:

- A job's account used from an address that is not its domain controller's.
- A job's account used before its credential was issued or after it was removed, or an account of that form with no job on the storage machine.
- Any other account reaching the staging share, another share over the staging folder, or a file under the staging folder through any share, including an administrative share such as `D$`.

The storage agent sees this in two ways: it reads logon and share access events (4624, 5140, and 5145) from its Security log as they happen, and every 30 seconds it polls the sessions, share connections, and files open under the staging folder. A connection shorter than the poll is seen only through the Security log, so it depends on the storage machine's audit policy. Cygna Forest Recovery reads that policy and never changes it; see Audit policy on the storage machine.

Each finding raises the critical alert **Unexpected SMB connections to the staging share**, with the account, the address, and what was wrong, for seven days after it was last seen. The first sighting of each is recorded in the audit trail as **Unexpected use of the staging share**. The storage agent closes no session and removes no account early. A storage machine whose watch is off or partial raises **Storage machines not fully watching the staging share**, which says what is missing. For both alerts, see Alerts Reference.

Verification before a restore

Before the storage agent grants read access to a backup for a restore, it verifies that backup. No unverified content is ever exposed to a restore.

By default, the storage agent can rely on its own recent verification instead of reading the whole backup again, as long as the result is a pass, is inside the re-verify window, and nothing in the backup has been written since. Otherwise it reads the backup in full first. To always force a full read, select **Re-verify the backup in full before restoring it** when you

plan a single-DC restore, or **Re-verify each backup in full before restoring it** in a forest or domain plan. A full read adds minutes per domain controller.

If the backup does not match its hash, the restore is refused with **The backup no longer matches the hash recorded when it was committed**. If there is no recorded hash to check against, it is refused with **The backup could not be checked against a recorded hash**. In both cases nothing has been sent to the machine. Choose another restore point.

Removing local administrators

The **Remove BUILTIN\Administrators from the vault** setting on the storage agents removes the local Administrators group from the vault and staging permissions, leaving the local system account and the `CygnaDR Administrators` group. It is off by default and takes effect when the storage agent restarts.

This setting is not a barrier: a local administrator can still take ownership of the vault and grant themselves access. It provides evidence: reading the backups then requires a deliberate ownership change.

To keep you from locking yourself out:

- The storage agent refuses to apply the setting while the `CygnaDR Administrators` group is empty.
- If the storage agent finds it can no longer write to the vault after applying it, it puts `BUILTIN\Administrators` back.

Grant access by adding accounts, including a break-glass account for the storage machine, to the `CygnaDR Administrators` group rather than by editing permissions by hand.

Encrypting the vault volume

Cygna Forest Recovery does not encrypt backups itself. Turn on BitLocker for the volume that holds the storage root, so that the backups stay unreadable if a disk, an array, or a virtual disk file leaves the machine, for example when it is copied, stolen, or decommissioned.

- **Unlock the volume automatically**—The storage agent does not start if it cannot reach the storage root, and a locked volume cannot be reached. Turn on BitLocker for the operating system volume as well, and turn on auto-unlock for the storage volume, so that it is available after every restart.
- **Keep the recovery key outside the forest you protect**—The storage machine is not joined to a domain, so it cannot back up its BitLocker recovery key to Active Directory. Even if it could, that directory may be the one you are recovering. Store the key offline with the other secrets from setup.

- **Give a virtual storage machine a virtual TPM**–Without one, the operating system volume needs a startup password that someone must enter at every restart.

BitLocker protects the backups at rest only. It does not protect them from someone signed in to the running storage machine, as the next section explains.

What this protection does not cover

These measures keep backups out of reach of the directory and of the network. They do not protect the vault from someone who is a local administrator on the storage machine itself, who could also change the storage agent's own retention configuration. Changes such a person makes to committed backups are detected and raised as alerts, as described above, but detection does not stop them. Treat the storage machine as you treat the most sensitive systems in your estate: keep it off the domain, keep its local administrators few, and keep its sign-in credentials separate from any domain credential.

Retention

Retention decides which backups stay in the vault and which are deleted. It runs on the storage machine, as a sweep on a regular interval and whenever an incoming backup needs room. It is the only way a backup is ever deleted.

Retention has two jobs that pull in opposite directions: keep enough history that a backup from before a compromise is still there when you need it, and keep the vault from filling. The settings below balance them. An attacker who can cause backups can also fill the vault, so filling the vault must never be a way to delete the clean backups an attacker would want gone.

Retention settings

Retention settings are on the **Storage agents** tab of the **Configuration** screen, in the **Vault retention** group. Two of them can be set only on the storage machine itself.

Setting	What it does
Retention floor	An age, in days. Nothing younger than the floor is deleted to make room for an incoming backup. It defaults to the Active Directory tombstone lifetime of 180 days. Set on the storage machine only.
Minimum versions per DC	No rule, for any reason, takes a domain controller below this many versions. It is at least 1, so a machine's only backup is never deleted. Set on the storage machine only.
Maximum version	A ceiling: versions older than this are deleted. `0` keeps them

Setting	What it does
age	indefinitely.
Maximum versions per DC	A ceiling: how many versions to keep for each domain controller, however recent they are. `0` means no limit. Which versions survive is decided by the retention mode.
Retention mode	`newest` or `thinned`. Decides which versions the count ceiling and a full vault delete. See Retention modes.
Free space reserve	Free space kept on the volume beyond what an incoming backup needs, so that the volume never fills completely.
Sweep interval	How often the scheduled retention sweep runs.
Dry run	Selects and reports what retention would delete, without deleting anything.

For defaults, ranges, and configuration keys, see Settings Reference.

Floors and ceilings

The settings fall into two kinds, and they bind different things:

- **The floors protect—Minimum versions per DC** binds every deletion: no ceiling, mode, or full vault takes a machine below it. **Retention floor** binds space pressure only: when the vault is full, nothing younger than the floor is deleted in `newest` mode.
- **The ceilings delete—Maximum version age** and **Maximum versions per DC** ask for deletion. When you set them on the storage machine, they decide how deep the history actually goes, even inside the retention floor. A maximum you set is a maximum.

Note: **Important:** The retention floor does not by itself keep 180 days of history. Whenever a ceiling is set, the ceiling decides the real depth. Check **Maximum versions per DC** and **Maximum version age** against the history you need, which for a ransomware response is usually the attacker's likely dwell time plus a margin, up to the tombstone lifetime.

Settings that only the storage machine can weaken

Most storage agent settings are held by the Manager and pushed to the storage agents, which keep using the last values they received if the Manager is unreachable. Retention is treated differently, because the Manager is on the network and the storage machine is deliberately isolated from it:

- **Local-only settings**—**Retention floor** and **Minimum versions per DC** are read only from the storage agent's own configuration on the storage machine. The console shows what each storage agent reports, marked **Set on the machine**, and the Manager refuses any attempt to set them.
- **Clamping**—For every setting that decides whether a backup is deleted, the storage agent compares the value it receives from the Manager with its own and uses the more protective of the two. A value sent from the console can tighten retention but never weaken it. For example, a **Maximum version age** sent from the console that falls inside the storage machine's floor is raised to the floor. The storage agent reports every clamp, and the **Configuration** screen shows the value that was sent beside the value in use.

To weaken retention below what the storage machine allows, you must change the storage agent's own configuration on that machine.

Changing a setting that can delete backups from the console asks you to review the change and type `CONFIRM` before it is saved. Every retention change is recorded in the audit trail as **Retention changed**.

Retention modes

Newest

In `newest` mode, the default, the count ceiling keeps each domain controller's most recent versions and deletes the oldest first. When the vault is full, retention deletes only versions older than the retention floor. If that does not free enough room, the incoming backup is refused rather than a version inside the floor being deleted.

A refused backup raises the **Vault refusing backups** alert, which you can act on. A deleted clean backup cannot be recovered.

Thinned

In `thinned` mode, retention keeps each domain controller's newest version and its deepest version inside the retention floor, and deletes from between them, choosing whichever deletion leaves the smallest gap for its age. Versions older than the floor are deleted first. What survives is dense near today and sparse toward the floor, so a small number of versions spans the whole floor.

As versions age out at the bottom, thinning keeps the chain without a gap wider than the floor divided by (maximum minus 4), so another version is always close enough to the floor to take over as the deepest.

The trade: under space pressure, `thinned` mode can delete versions inside the retention floor, where `newest` mode refuses the backup instead. It only takes versions from the

middle, never the newest or the deepest, so a full vault costs granularity rather than depth. Neither mode takes a domain controller below **Minimum versions per DC**.

Worked example

Consider a domain controller backed up once a day, with a retention floor of 180 days and **Maximum versions per DC** set to 10.

	Newest	Thinned
Versions kept	The 10 most recent	10, spread from today back toward the floor
History covered	About 10 days	The whole 180 days; the deepest version stays above about 150 days old
Widest gap between versions	1 day	No more than 30 days (180 ÷ (10 − 4))
When the vault is full	Deletes only versions older than 180 days; otherwise refuses the new backup	May delete versions from the middle of the 180 days; keeps the newest and the deepest

If an attack went unnoticed for 60 days, `newest` mode in this example would hold no backup from before it, while `thinned` mode would still hold several. A smaller maximum gives wider gaps and a shallower worst case.

Space pressure

Before an incoming backup starts, the storage agent reserves room for it: the Manager's size estimate from the machine's recent backups, plus a margin, plus the free space reserve. The reservation stops concurrent backups from each passing a free-space check and then together overrunning the volume. If there is not enough room, retention runs under space pressure as described for each mode. The **Repository** screen shows capacity and each retention action, including whether it was a scheduled sweep or space pressure, and any backup that was refused.

Dry run

With **Dry run** on, retention selects and reports what it would delete, but deletes nothing. Use it to see what a new policy would remove before letting it run. The **Repository** screen shows a notice while retention is in dry run, and marks the previewed deletions.

Dry run is the more protective state, so if it is set on the storage machine, it cannot be turned off from the console.

Related topics

- Settings Reference
- Repository
- Sizing Backup Storage

Recovery Images

A full system recovery restores a domain controller's whole server, operating system included, from a full system backup. Windows cannot restore its own system volume while it is running from it, so the restore runs from a separate, small Windows environment loaded into memory: the recovery image.

What a recovery image is

A recovery image is a Windows PE environment built from the Windows Recovery Environment (Windows RE) image inside a Windows Server installation ISO that you supply. The storage agent adds the recovery agent to it, and any drivers you provide. It is not a copy of the domain controller's own operating system, and it holds no backup data. Its only job is to start, reach the vault, and restore the backup onto the machine's disks.

The recovery agent is not installed or enrolled separately. It ships with the storage agent and is copied into every recovery image the storage agent builds.

One image per operating system

A recovery image must come from the same Windows Server release as the domain controller it restores. The storage agent builds one image for each operating system variant your domain controllers run, as recorded in their newest backups. A variant is the combination of:

- The Windows build.
- The edition.
- The install type: Desktop Experience or Server Core.

Each value must match exactly. The update build revision (UBR) is not part of the match, so an ISO older than your patched domain controllers still works. Architecture (x64 or ARM64) must match as well. On the **OS Media** screen, media in a language your domain controllers do not run is reported as **Wrong language only**; it installs and works, but the tool output an operator reads during recovery will be in that language.

How images are built

The storage agent builds images ahead of time, because a build takes minutes and needs the installation media, both of which are scarce during a disaster. It builds them with the DISM tool included in Windows; the Windows Assessment and Deployment Kit is not needed.

- Images are built only when the storage agent's **Recovery image folder** is set. If it is empty, no images are built and no domain controller can use full system recovery. DC state recovery is not affected.
- The installation media come from the storage agent's **Media folder**, which holds your Windows Server ISOs. Keep the recovery image folder separate from the media folder; Cygna Forest Recovery never writes to the media folder.
- The storage agent checks on a schedule whether any required image is missing or out of date, and builds it.
- When you upgrade the storage agent, it builds new images with the recovery agent of the new release. A new image is untested until a restore proves it. A new image is added beside the existing ones, and older ones are retired as described below.
- When more than one image fits a variant, an image already proven by a completed restore is used ahead of a newer one that has not been proven yet.

Proven images and retirement

An image is **Proven** once a full system recovery that staged it reports a completed restore. A test restore counts the same as any other recovery. The Manager records the verdict and the storage agent stores it in the library's catalog. An image that has failed keeps its failed verdict. A new image is **Untested** until a restore proves it.

With an Isolated Recovery Host enrolled and the **Prove new recovery images** setting on, untested images are test-restored automatically, each with a small recent backup of a domain controller the image serves, or as part of a domain controller's scheduled test, so a new image is proven before you need it. Only the newest image of each operating system variant on each storage agent is tested this way, because that is the image a restore would start from. See Test Restores.

Every product build adds a new image per operating system variant, so the library would grow without limit. The Manager therefore decides which images each library keeps, and the storage agent deletes only the images it is told to. For each operating system variant that restore points in retention still need, the Manager keeps:

- The newest image, whatever its verdict.
- The newest proven image.

- If the newest image has failed, the newest image that has not failed.
- Any image a restore that has not ended is using. An image in use is never retired.

  Everything else is retired: superseded images, failed images once a newer one exists, and images of a variant that no restore point in retention needs. Nothing is decided for a library while its last report did not succeed. No setting controls retirement. The **Recovery images** tab of the OS Media screen and `cdr os-media list --images` show what each library keeps, what it retires, and what it has retired.

  For the setup steps, see Setting Up OS Media and Recovery Images.

How an image is staged and started

For a recovery in place or onto a candidate, a recovery image is not booted from removable media or the network. The DC agent stages it on the machine being recovered while that machine's own Windows is still running, and the machine then starts from it once:

1. The DC agent on the target machine fetches the image from the storage machine, checks its hash, and copies it to a folder on the system volume.
2. The DC agent sets the machine to start from the image for the next start only, with the loader that matches its firmware, UEFI or BIOS, and restarts it.
3. The whole image loads into a RAM disk, and Windows PE starts from memory. This is why the target needs enough memory for the image plus about 1 GB.
4. The recovery agent takes its network address, from DHCP unless the plan gives it one. It reaches the storage machine at the storage agent's advertised address—its IP address by default, so no DNS is needed unless the advertised host is a name. It reaches the Manager at the address the machine's DC agent uses, resolved to an IP address before the machine restarts. Its first contact goes out moments after the network is set, so it tries for 15 seconds; if the Manager still cannot be reached, it restores without reporting, since the vault credential is in the staged job.
5. It checks its own view of the machine's firmware and disks against the backup before anything is written, then reads the backup through its read lease and restores every critical volume. It reports each volume's progress to the Manager, using a token that is valid for this one recovery only.
6. After a successful restore, the recovery agent replaces the DC agent on the restored volume with the build of the agent that staged the restore, keeping the backup's copy beside it, so that the restored domain controller does not come back on an older agent from its backup.
7. When the Manager has recorded that the restore succeeded, the recovery environment restarts the machine into the restored system. The restored DC agent reconnects, checks that the domain controller works, and the recovery completes.

Because the start from the image is one-shot, a failure before anything is written is safe. If the image does not start, or the recovery environment refuses (for example, because it cannot reach the vault), the machine returns to its own Windows at the next start and the run reports what happened. If the restore fails after writing, or the Manager does not confirm the result, the recovery environment stays at its prompt for someone to look at.

The DC agent must be able to stage the image, so the machine's own Windows has to start one last time. A domain controller whose Windows no longer starts is recovered onto a candidate machine instead, where the candidate's own DC agent stages the image, or onto a new VM. See Restore Methods.

Booting from a recovery ISO onto a new VM

A full system recovery onto a new VM, and every test restore, needs no Windows or agent on the target. When the run is executed, the storage agent builds one ISO for that run from the installation media's boot files, the recovery image, and the job. The new VM starts from that ISO, and the recovery agent enrolls with the Manager using a single-use token the ISO carries; the vault credential is handed over only after it enrolls. Because the restore cannot start without the Manager, the recovery agent keeps trying its first contact for 90 seconds, which rides out a network that is still coming up. Before anything is written, the recovery agent checks the VM's firmware and disks against the backup.

In a test restore, the recovery agent powers the VM off after the restore instead of restarting it, so the Isolated Recovery Host can isolate the VM before its first start. See Full System Recovery onto a New VM and Test Restores.

Recovery image status

The **Recovery image** column on the **OS Media** screen states, for each operating system your domain controllers run, whether a domain controller on it can use full system recovery. An **Available** status means an image exists and will be used; **image untested** means no restore has proven it yet, which is normal for a new image until a full system recovery or test restore completes with it. An **Unavailable** status names what is missing, such as media to build from or an image folder.

Every unavailable status affects full system recovery only. A DC state recovery of a domain controller whose Windows still starts does not need a recovery image. For each status and what to do about it, see OS Media.

Related topics

- OS Media
- Recovery Requirements

Test Restores

A stored backup that passes its integrity checks is still not proven to restore. A test restore proves it:Cygna Forest Recovery restores the backup onto a new virtual machine on an Isolated Recovery Host, starts the restored system where it cannot reach your network, and checks that it comes up as a working domain controller. The result is recorded against that backup.

Nothing in production changes. The original domain controller keeps running, and the restored copy never reaches its network. When the test ends, the virtual machine and its disks are deleted.

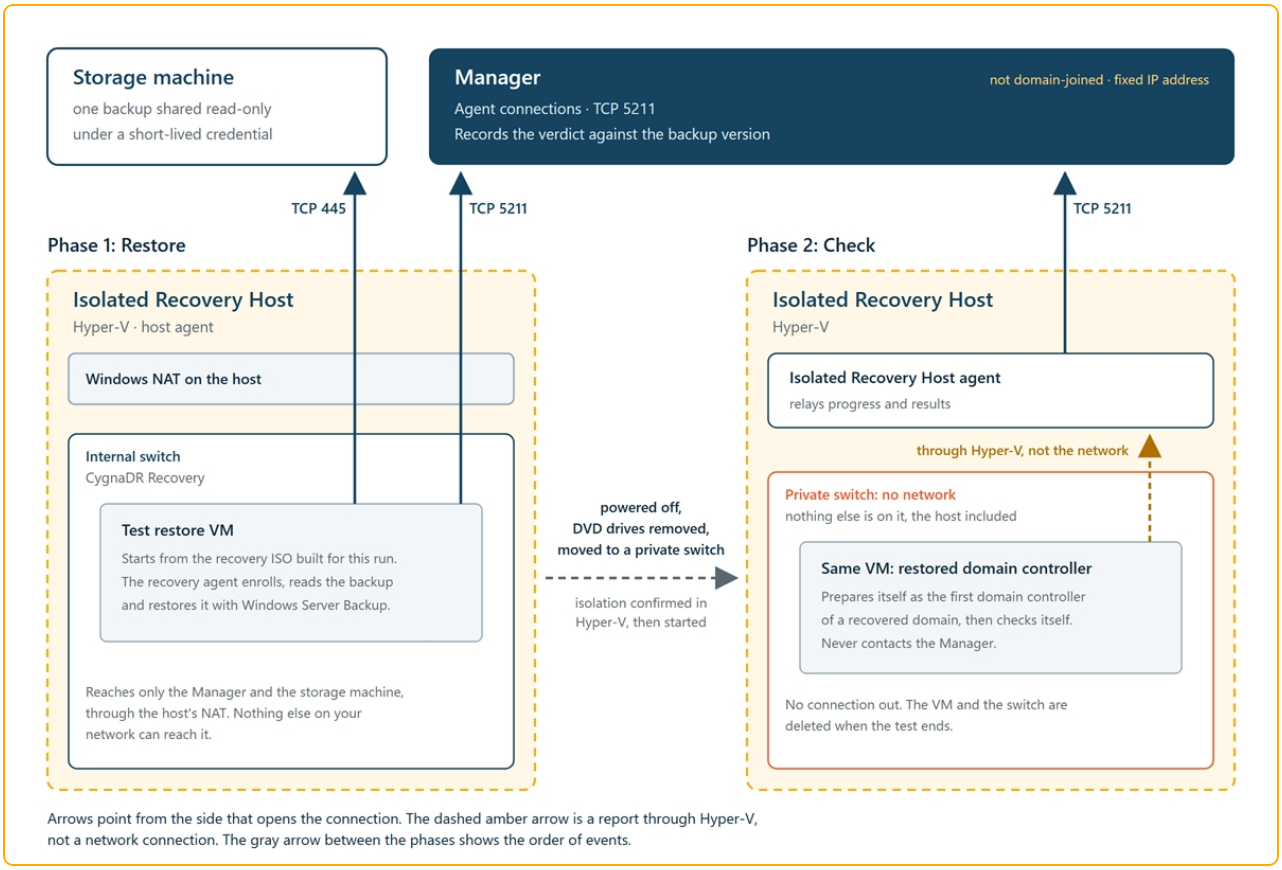
Test restores need an Isolated Recovery Host, which is optional. Without one, everything else works the same, and backups are shown as not verified. To set one up, see Setting Up an Isolated Recovery Host.

How a test restore runs

A test restore is a full system recovery onto a new virtual machine. Up to the moment the restored system first starts, it follows exactly the path a real recovery takes, so it tests that path as well as the backup.

1. **The host builds a virtual machine.** The Isolated Recovery Host agent creates a virtual machine shaped like the backed-up domain controller, from the hardware recorded in the backup, reduced to the processors and free memory the host has. It is connected to the host's recovery network, which reaches the Manager and the storage machine through the host and is unreachable from anywhere else.
2. **The backup is restored.** The virtual machine starts from a recovery ISO built for this run. The recovery agent in it enrolls with the Manager, reads the backup from the vault under a short-lived credential, and restores it with Windows Server Backup, as in any full system recovery.
3. **The virtual machine is cut off.** When the restore finishes, the recovery agent powers the virtual machine off instead of restarting it, so the restored Windows cannot start yet. The host agent moves every network adapter to a private switch created for this run; nothing else is on that switch, not even the host. It also removes the virtual machine's DVD drives, so no media can stay attached. It confirms both from Hyper-V's own records, and only then starts the virtual machine. The host agent enforces this itself, whatever it is asked to do.
4. **The restored system checks itself.** Instructions written to the restored disk, and signed by the Manager, tell the restored domain controller to prepare itself as the first domain controller of a recovered domain and then check itself. It never tries to contact the Manager. It reports its progress and results through Hyper-V, and the host agent relays them.

5. **Everything is deleted.** The virtual machine, its disks, the ISO copy, and the private switch are deleted. Anything the host could not delete is removed the next time the host agent starts.



What is checked

The restored domain controller is prepared the way Microsoft's forest recovery guidance prepares the first restored domain controller of a domain:

- Its SYSVOL is marked authoritative, and it does not wait for replication partners at startup, because it has none on its private switch.
- It is made its own reliable time source: its NTP server is turned on, it announces itself as a reliable time server, and the Windows Time service is set to start automatically. A domain's first restored domain controller has no other time source to follow, and a domain controller that serves no time does not advertise as one.
- Its DNS client is pointed at itself, first at its own address and then at 127.0.0.1, so it can resolve its own names without the DNS servers it used in production, which are not on its private switch. A domain controller without the DNS Server role has its own names mapped locally for the checks instead.

- It keeps the IPv4 address its backup recorded, even one that DHCP gave it, when the backup recorded an address on one adapter. The private switch has no DHCP server, and the domain controller's DNS records name that address.

It then restarts and runs these checks, repeating them every 30 seconds for up to 20 minutes while its services come up:

Check	What it confirms
Prepared as the first DC	The preparation above was applied and the system restarted. The check's detail says what the backup had, for example that its NTP server was off. The check fails when the DNS client could not be pointed at the domain controller itself, when a setting could not be written, or when a Windows Time Group Policy turns the NTP server off or announces no time server. A Group Policy setting overrides the local one, so the detail names the policy setting; a recovered domain's first domain controller must serve time, so that policy has to change before this domain controller can be one.
Directory service running	Active Directory Domain Services started.
SYSVOL ready, **SYSVOL shared**, **NETLOGON shared**	SYSVOL is ready and both shares are published.
DNS zones loaded	The DNS zone that holds the domain's names loaded, on a domain controller that runs DNS. That can be the domain's own zone, a parent domain's zone, or the root zone. A domain controller without the DNS Server role, or one that holds no zone with the domain's names, passes with a note that a forest recovery needs DNS for the domain from its first restored domain controller, and has its own names mapped locally for the checks. A zone it holds that did not load fails the check.
dcdiag (local tests)	The `dcdiag` tests that need no other domain controller pass: Advertising, Services, NetLogons, and SysVolCheck.
Object counts	The restored directory's object counts are compared with the counts last reported for the domain. This check is advisory and does not decide the verdict.

Verdicts

Each test restore ends with one of these results, recorded against the backup version it tested:

- **Restore verified**—The backup restored, the restored system started isolated, was prepared as the first domain controller, and passed every required check.
- **Restore failed**—Something the backup is responsible for went wrong. The verdict names the step and the reason: the restore itself, the restored system not starting, its agent not reporting, or a check that did not pass. If the restored system says nothing within 90 minutes of starting, the host's own evidence decides the reason, for example that Hyper-V never received a heartbeat from it.
- **No verdict**—The test could not do its part, for example because the host could not build or isolate the virtual machine, the test was cancelled, or the host stopped reporting on the virtual machine before a verdict. The backup is not blamed and stays unverified. If a test restore restores and then ends with no verdict, and the virtual machine's console shows that Secure Boot refused to start the system, the host is most likely behind on Windows updates; see Keeping the host current.

The host agent reads the isolated virtual machine every 20 seconds and reports what it reads when it changes, and at least every two minutes. If a read fails, the host keeps trying and reports why it cannot read the virtual machine. If the Manager hears nothing about the virtual machine for six minutes, it says so on the run's timeline and asks the host to watch the virtual machine again, without restarting or moving it. If the host's report is still incomplete when the 90 minutes are up, the run ends with no verdict at step `host`: what the host could not see is not held against the backup. Plan the test restore again.

A verdict is about one backup version only. A newer backup of the same domain controller starts unverified.

Verdicts are shown on the run's page, in the **Test restore** column of the Restore Points Screen screen (sortable and searchable, with a link to each run) and on each restore point's page, next to each restore point in the restore point list of a new single-DC restore (**test restore verified**, **test restore failed**, **test restore running**, **test restore waiting to start**, or **test restore inconclusive**), in `cdr restore show`, and in `cdr backup list`. Two alerts report on them; see Alerts later in this topic.

A test restore that completes its restore also proves the recovery image it started from, exactly as a real full system recovery does. For more information, see Recovery Images.

Scheduled test restores

Test restores go only to a host an operator has approved. A newly installed host waits for approval however ready it is, and a drained host takes no new test restore. With more than one approved host, each test restore goes to whichever host can run one. Once an

approved host is ready, the Manager schedules test restores itself. It looks for work two minutes after it starts and every five minutes after that, and chooses in this order:

1. **Each domain controller's new backups first.** For each domain controller whose newest full system backup has not passed a test restore, it tests that backup before a deadline. The deadline is the **Test each DC's new backups within** setting (30 days by default), counted from the domain controller's first backup taken after the newest one that passed. A backup that passed is never tested again, so a domain controller whose newest backup has passed waits for its next one. A test restore that an operator started counts.
2. **New recovery images in the time left.** For each recovery image that no restore has proven yet, it tests the smallest full system backup from the last 30 days of a domain controller that image serves, starting from that image. An image test starts only when it would finish before a domain controller's test must start. Only the newest image of each operating system variant on each storage agent is tested: that is the image a restore would start from, so proving an older one would prove nothing a recovery relies on. This is controlled by the **Prove new recovery images** setting.

So that new images are proven even when the domain controllers' tests leave no time between them, a domain controller's scheduled test starts from an untested image of its operating system variant when one is waiting, and proves that image too. A domain controller already past its deadline is tested with the proven image instead, to clear its alert sooner. If a test that started from an untested image fails, the domain controller waits for the retry like after any failed test.

Scheduled test restores keep out of the way:

- One test restore runs on a host at a time.
- None starts while a backup is running, or when one is due within the **Keep clear of backups for** setting (120 minutes by default).
- After a scheduled test restore of a domain controller or image does not pass, the schedule leaves it alone for the **After a scheduled test restore fails, wait** setting (24 hours by default) and moves on to the next.
- A scheduled test restore never accepts a confirmation on a person's behalf. If its checks need one, it is discarded and tried again later.

The earliest deadline goes first. Each test starts as late as still lets every domain controller waiting finish before its deadline, and before the backups the schedule predicts from the enabled backup policies. The prediction allows for only **Maximum concurrent backups** running at once, so domain controllers sharing a start time keep test restores away until the last of their backups is expected to finish. The result is about one test per domain controller in each period, run shortly before its deadline. Each test restore reads a full backup from the vault, and a host runs one at a time, so the host's speed and the vault's load decide how many domain controllers a period can cover. With more than a host can reach, the tests start earlier and run back to back. If a deadline still passes, the alert for domain controllers

with no recent verified backup is raised: add a host or lengthen the period if it keeps happening.

Only full system backups can be test-restored. Retired domain controllers are skipped. The Agents screen and `cdr recovery-host list` show what the schedule last decided for each host and why. For the settings, see Settings Reference.

Test restores and the rest of the product

- A test restore never holds a backup, never delays a DC agent update, and never blocks a real recovery. A recovery always takes priority; a test restore waits for anything else in progress on the same domain controller.
- The original domain controller keeps running throughout. The restored copy is never mistaken for it, because it never contacts the Manager.
- The recovery banner, the Dashboard's live operations panel, and `cdr restore list` say that a test restore is one, and whether it was scheduled or who approved it. On the Recovery Plans screen, a test restore is listed as a single-DC restore. See Dashboard.
- Keep the Isolated Recovery Host on current Windows updates, and its agent at the version the Manager offers, so that the host supports everything the Manager asks of it during a test restore. See Maintaining Agents.
- A test restore can be cancelled at any point. The virtual machine is deleted and the backup stays unverified.
- Planning and approving a test restore requires the `plan.approve` capability, and running or cancelling it requires `plan.execute`, as for any restore. Scheduled test restores run as **schedule** in the audit trail.

Alerts

- **Domain controllers whose latest test restore failed** (critical)—The newest test restore of a domain controller's backups ended with **Restore failed**. Open the test restore to see the step and the reason, then test a newer backup.
- **Domain controllers with no recent verified backup** (warning)—A domain controller's newest backup has not passed a test restore, and its deadline under the **Test each DC's new backups within** setting has passed: the hosts did not reach it in time, or its tests are not passing. This alert is raised only while an Isolated Recovery Host is enrolled, and only for domain controllers with full system backups. Setting the period to 0 turns it off.

These alerts appear on the Alerts screen. For details, see Alerts Reference.

Related topics

- Running a Test Restore
- Setting Up an Isolated Recovery Host
- Restore Points
- Restore Methods

Coverage and Readiness

Cygna Forest Recovery continuously judges whether your estate could be recovered with the backups you have now. It gives a verdict for each domain controller, each domain, and each forest, and the dashboard rolls them up. This topic explains how each verdict is reached, so that you can tell what a verdict means and what clears it.

How verdicts are read

Every verdict has one of five states, each shown with its own icon as well as a color:

- **OK**–Verified good.
- **Warning**–A known, bounded problem. Recoverable today, but with a reason it may not be tomorrow. Shown as "at risk" for domains and forests.
- **Critical**–Verified bad. Recovery capability is impaired now.
- **Unknown**–Something could not be found out. An unknown is never reported as good or as gone. When figures are rolled up, unknown ranks above warning, because a risk of unmeasured size can be worse than one that has been measured.
- **Info**–A neutral notice with no action implied.

Domain controllers

Each domain controller has a protection verdict, shown on the **Domain Controllers** screen. It answers one question: is there a restorable backup of this machine, and is it being kept current? The freshness window is twice the machine's backup policy interval.

Verdict	Meaning
Protected	A restorable backup exists within the freshness window, and the last attempt succeeded.
Last backup failed	The newest restorable backup is within the window, but the most recent attempt failed. A backup refused because the machine had no agent, made before its current agent enrolled, does not count

Verdict	Meaning
	as a failed attempt.
Backup stale	The newest restorable backup is older than the window.
No schedule	A restorable backup exists, but no enabled policy covers the machine, so nothing will replace it.
Unprotected	A backup target with no restorable backup: an enabled policy covers the machine, but it has no agent, has never been backed up, or every version has been pruned or never committed. A warning; whether it costs the domain its recovery is the domain's verdict.
Rebuilt on recovery	Not a backup target: no enabled policy covers the machine and nothing of it is in the vault. A recovery rebuilds it by promotion. Info, not a finding.

No domain controller verdict is critical. Critical belongs to a domain or forest that could not be recovered. For how backup targets are decided, see Backup Targets and Rebuilt Domain Controllers.

Site verdicts on the **Sites** screen are a roll-up of these per-machine verdicts over the site's backup targets, and none is critical. Domain and forest verdicts are not a roll-up, because recovering a domain depends on which domain controllers can start the recovery, not on how many are protected.

Domains

A domain is recovered by restoring one writable domain controller from backup and rebuilding the rest. Its verdict on the **Domains** screen therefore asks first whether any writable domain controller has a restorable backup, and then how good that starting point is: whether it is inside the tombstone lifetime, scheduled, a full system backup, and fresh, and whether a second writable domain controller could take its place. Read-only domain controllers never count, because they are never restored. A domain is **Covered** when two or more writable domain controllers hold full system backups, an enabled policy covers at least one of them, and the newest backup is inside the freshness window and the tombstone lifetime. The freshness window for a domain is twice the tightest backup interval among its writable domain controllers.

Verdict	State	Meaning
Covered	OK	Two or more writable domain controllers with full system backups, and the newest backup is fresh and inside the tombstone lifetime.
Single recovery option	Warning	Only one writable domain controller holds a full system backup. Recovery would succeed today, with nothing to fall back to if that backup fails verification or turns out to be post-compromise.
Only one domain controller	Warning	The domain has a single domain controller, so the two-per-domain rule cannot be met by backup configuration. Promote a second domain controller.
Recovery point stale	Warning	The newest restorable backup of a writable domain controller is older than the freshness window.
No full system recovery	Warning	Every restorable backup of a writable domain controller is DC state only. DC state recovery is possible; a lost or encrypted machine cannot be recovered.
No schedule	Warning	A recovery point exists, but no enabled policy covers any writable domain controller, so it only grows older.
Past tombstone lifetime	Critical	The newest restorable backup of a writable domain controller is older than the tombstone lifetime. Restoring it is refused.
FRS SYSVOL – unsupported	Critical	The newest backup reports SYSVOL replicated by FRS. Recovery requires DFSR. Migrate the domain to DFSR, then take fresh backups; backups from before the migration stay unusable.
Unrecoverable	Critical	No writable domain controller holds a restorable backup, so there is nothing to start the domain's recovery from.
Recoverability	Unknown	Restorable backups exist, but whether their

Verdict	State	Meaning
unconfirmed		domain controllers are writable has not been collected yet. A forest enumeration answers it.
No domain controllers found	Unknown	No source has found a domain controller in this domain. This is often leftover metadata from a domain that was not removed cleanly.

A domain with a warning verdict could still be recovered today. The console counts it as recoverable, at risk. A domain with a critical verdict could not.

Forests

A forest recovery restores the forest root domain first and every other domain after it, parent before child. A forest's readiness on the **Forests** screen is therefore a roll-up of its domain verdicts, with the root weighted separately.

Domains you have excluded are left out of the roll-up and its domain and domain controller counts, and the **Forests** screen says how many are not counted. The forest root domain is the exception: it always counts, even when it is excluded, because a forest recovery cannot start without it. Excluding a domain does not change that domain's own verdict on the **Domains** screen.

Readiness	Meaning
Ready	Every domain is **Covered**.
Root domain unrecoverable	The forest root cannot be recovered, so a forest recovery cannot start. Nothing below the root can proceed until this is fixed.
Domains unrecoverable	The root could be recovered, but at least one other domain could not. A recovery would start and stop partway.
Root domain at risk	The root could be recovered today, with a warning against it. A finding on the root decides how the whole recovery begins.
Domains at risk	Other domains carry warnings, and each would still recover today.
Root domain unidentified	The root domain is not known, or cannot yet be judged.
Recoverability	Every domain that can be judged is recoverable, and some

Readiness	Meaning
unconfirmed	are not judged yet. Normal on a new installation until the first topology enumeration arrives.
No domains found	An enumeration reached the forest and returned no domains. Check the DC agent's last topology report.
No agent in this forest	The forest is known only because a trust names it. Nothing in it is protected by this installation.

The dashboard's **Recovery readiness** section states each forest's verdict in a word: **Fully recoverable**, **At risk**, **Not recoverable**, **Not confirmed**, or **Not covered**, with the exact readiness beneath it. Each forest's card also shows the **Root recovery point**—the age of the root domain's newest recovery point—and the **Last successful restore in this forest**.

Tombstone lifetime

The tombstone lifetime is how long Active Directory keeps a record of a deleted object before removing it for good. A backup older than the tombstone lifetime still contains objects that every other domain controller has since forgotten were deleted. Restoring it brings them back as lingering objects. For that reason preflight refuses any restore point older than the tombstone lifetime, and coverage treats such a backup as unusable.

The tombstone lifetime is read from each domain. Until it has been collected, Cygna Forest Recovery assumes the Active Directory default of 180 days, and the console says it is assumed. The **Domains** screen shows how many days of headroom each domain's recovery point has left, and shows **Nearing tombstone lifetime** when less than a quarter of the lifetime remains.

Retired and excluded objects

Two kinds of operator decision change what is counted. Neither hides anything: retired and excluded objects stay listed on their screens, with the reason and who recorded it.

- **Retired (removed from the directory)**—An object the directory no longer defines, such as a domain controller that has been demoted or cleaned up. A domain controller, domain, or trust is recorded as removed by discovery, by the `cdr cleanup` commands, or by an operator with **Mark as removed from the directory…**. Retired objects leave every count and alert. Nothing in Active Directory changes. If the directory names the object again, the record is withdrawn automatically.
- **Excluded from alerts**—An object you have decided is out of scope, such as a forest reached across a trust that this installation is not meant to protect. A domain controller you do not back up needs no exclusion: it is rebuilt on recovery and raises no alert. You record a reason and, optionally, a review date. The object stays listed and judged, but

is left out of every count and alert. After its review date it counts again and shows **Exclusion due for review**. If the object changes, the exclusion is void. An exclusion on a forest, domain, or site applies to everything in it. The one exception is the forest root domain, which always counts toward its forest's readiness, even when excluded.

Excluding and retiring objects requires the `backup.manage` capability. See Excluding and Retiring Objects.

How alerts relate to verdicts

Alerts are raised from the same verdicts the inventory screens show, so the two never disagree. For example, a domain with the **Unrecoverable** verdict raises the **Domains that could not be recovered** alert, and a domain controller with the **Unprotected** verdict raises **Backup targets with nothing to restore from**. A domain controller that is rebuilt on recovery raises no alert. Each inventory screen lists its own alerts as findings in the summary panel at the top, and the **Alerts** screen lists everything currently raised, grouped by severity, with a link to where you act on it. Excluded and retired objects raise no alerts.

Test restore results do not change these verdicts. They raise their own alerts, **Domain controllers whose latest test restore failed** and **Domain controllers with no recent verified backup**, which appear on the **Alerts** screen. See Test Restores.

For every alert, its severity, and what to do, see Alerts Reference.

Related topics

- Dashboard
- Backup Targets and Rebuilt Domain Controllers
- Domains
- Forests

How Recovery Works

Every recovery in Cygna Forest Recovery is planned before anything changes. You describe what to recover and from which restore points, the Manager checks the plan against the estate, a Restore Operator approves it, and only then is it executed, one tracked step at a time. This chapter explains the model, so that you can choose the right recovery and know what it will do before you start one.

Domain and forest recovery is built on Microsoft's Active Directory Forest Recovery Guide: each step of a plan carries out a step of that guide, in its order. For how each step is applied, and where Cygna Forest Recovery differs, see Alignment with Microsoft Guidance.

Plan first

Planning a recovery writes nothing to any machine. It records the plan and evaluates its preflight checks: whether each chosen restore point is usable, whether each target machine can receive it, and whether the plan as a whole is coherent. Each check is either hard, which blocks approval until it is cleared, or advisory, which you read but do not have to clear. Advisory checks tell you what the recovery will cost. A restore can also raise a check that the approver must explicitly accept, for something the Manager could not prove from where it looked.

Approval records that someone has reviewed the plan. In a domain or forest plan, approval first re-checks each candidate machine that a domain controller is restored onto: its firmware and disks, as they are now, against the backup. If one of those checks blocks, approval is refused. Execution starts the plan. Some checks run again at execution, against the live estate: the backup's hash is verified against the vault before any read access is granted, the version is confirmed present where the restore will read it, and the target's boot configuration is rehearsed before it is set for real.

Once a recovery is running, each step reports its progress in the console. Steps that cannot be undone wait for an operator to acknowledge them, and steps only a person can check wait for an operator to attest to them. A domain or forest plan can instead be built to run unattended, in which case the operator who executes it accepts those pauses in advance, in their own name. A step that fails halts the plan until it is retried or the plan is aborted. A plan can email its halts, waits, and final recovery report to addresses you give it. Every decision, gate result, and state change is recorded in the audit trail.

Separation of roles

Recovery permissions are held only by the Restore Operator role:

- `plan.approve`—Draft and approve restores and recovery plans, and discard plans that were never executed.
- `plan.execute`—Run a restore or recovery plan, acknowledge its points of no return, attest to operator steps, retry a halted step, set a plan's notification addresses and email its report, and cancel, abandon, or abort.

No other role implies either capability. In particular, the System Admin role, which configures the product and manages operator accounts, cannot run a restore. Nobody can change their own roles, and the break-glass account cannot be changed by another administrator, so moving from account administration to running a restore always takes a second person. For the role definitions, see Managing Operators and Roles.

Scopes

A recovery has one of three scopes. Each is a different procedure, following the Microsoft Active Directory forest recovery guidance.

Scope	What it does
Single-DC recovery	Restores one domain controller in a domain that is otherwise healthy. The directory converges from the healthy domain controllers, except for the subtrees and objects a DC state recovery names, which go back to the restore point on every domain controller.
Domain recovery	Restores one domain inside a forest that is still running, and reconciles the rest of the forest with it.
Forest recovery	Brings every domain back from backup, forest root first, on isolated targets, then reconnects the recovered forest to production.

A single-DC recovery is a restore run. A domain or forest recovery is a recovery plan, which contains a restore run for each domain controller it restores. For details, see Recovery Scopes.

Methods

Each domain controller in a recovery is brought back by one of these methods:

- **DC state recovery**—Windows still starts. Active Directory goes back to the restore point through Directory Services Restore Mode. In a single-DC recovery, the subtrees and objects you name are marked authoritative, so they replicate out to the rest of the domain.

- **Full system recovery**—The whole server, operating system included, is restored from a full system backup, in place on the same machine, onto a candidate machine, or onto a new virtual machine booted from a recovery ISO built for the run.
- **Rebuild**—The domain controller is not restored. Its metadata is removed and a fresh domain controller is promoted in its place, with the Windows roles and features its last backup recorded reinstalled first. A domain or forest plan restores exactly one domain controller per domain and rebuilds the others.

In a domain or forest plan you can also exclude a domain controller. An excluded domain controller is neither restored nor rebuilt: its metadata is removed from the directory along with the domain's other domain controllers, and it is not brought back. Keep the machine off the network. For details of each method, see Restore Methods.

The same full system recovery path also runs test restores, which restore a backup onto an isolated virtual machine on an Isolated Recovery Host to prove it, and change nothing in production. See Test Restores.

It works when the directory does not

The Manager, its operator accounts, and its certificate authority do not depend on Active Directory or DNS, and agents reach the Manager by IP address. You can sign in, review restore points, and plan and run a recovery while every domain controller is offline. During a full system recovery, the recovery environment reaches the storage machine at its advertised address, by default an IP address, and needs DNS only if that address is a name; see Recovery Requirements.

The agent that stages a full system recovery writes its machine's time zone, and whether the machine keeps its hardware clock in local time or UTC, into the recovery job. The recovery environment reads the hardware clock through them, so the job's expiry, the Manager's certificate, and the times it reports are judged on the correct time wherever the machine is. The recovery environment never sets the clock, so the restored system starts on the clock as the machine kept it. Its reading of the clock is recorded in the run's transcript.

In this chapter

- Recovery Scopes
- Alignment with Microsoft Guidance
- Restore Methods
- Choosing a Recovery Approach
- Recovery Lifecycle
- Recovery Requirements

- Isolating Recovery Targets
- What Recovery Does Not Cover

Recovery Scopes

A recovery has one of three scopes: a single domain controller, one domain in a running forest, or the whole forest. The scope decides which procedure runs, what it is allowed to change, and what it refuses to do. You choose the scope on the **Recovery Plans** screen, under **Active Directory recovery**.

Single domain controller

Use a single-DC recovery when one domain controller is damaged and the rest of its domain is healthy. The domain controller keeps its identity, and its directory goes back to the restore point. After the restore, Active Directory and SYSVOL converge from the domain's healthy domain controllers, so everything they hold that this domain controller missed comes back through replication. Anything written only to this domain controller after the restore point and not yet replicated away is lost.

A DC state recovery of a single domain controller also undoes changes that have already replicated: the subtrees and objects you name are marked authoritative after the restore, and replicate out to every domain controller of the domain as they were at the restore point. A full system recovery marks nothing.

The restore runs in three stages:

1. **Preflight**—Confirms that the restore point is usable and that this domain controller may be restored at all.
2. **Restore**—Restores the domain controller by DC state recovery or full system recovery. The stage passes when Active Directory starts and both the directory and SYSVOL begin converging from healthy peers. For a DC state recovery, the named subtrees and objects must also have been marked authoritative before the restart, their back-links imported after it, and their raised versions read back from replication metadata.
3. **Validation**—Checks the secure channel and replication against a healthy domain controller in the same domain. The restored domain controller's machine account password is reset against a healthy peer when needed.

A single-DC recovery never seizes roles, removes other domain controllers' metadata, resets trust passwords, or raises the RID pool. Those steps belong to domain and forest recovery. It resets `krbtgt` only after a DC state recovery whose subtree holds the `krbtgt` account, such as `CN=Users`, and only when you choose the reset. It then resets it twice: first once the rollback validates, and again after the first reset has reached every writable domain controller.

A single-DC restore is refused when:

- The domain controller is a read-only domain controller. Rebuild it instead.
- Its operations master roles were seized by another domain controller after the backup was taken. Rebuild it; never restore it.
- The restore point is older than the tombstone lifetime, has been pruned, or has no version identifier.
- The domain replicates SYSVOL with FRS.
- Another restore of the same machine is already in progress.
- A DC state recovery names nothing to roll back, or names a subtree or object it does not accept. See Restoring a DC with DC State Recovery.

One domain in a running forest

Use a domain recovery when one domain is damaged or compromised and the other domains of the forest are healthy. The rest of the forest stays live, and the plan writes to it: it resets the domain's trusts on both sides and reconciles every global catalog in the forest with the restored domain. The plan shows these effects outside the domain before you approve it.

In the plan, you choose for each domain controller in the domain whether it is restored, rebuilt, or excluded. An excluded domain controller is neither restored nor rebuilt: its metadata is removed from the directory along with the domain's other domain controllers, and it is not brought back, so keep the machine off the network. By default, one writable domain controller is restored and the rest are rebuilt. The recovery then runs in this order:

1. Plan and preflight, including the global catalog strategy.
2. Isolate the domain's restore targets from production, with firewalled paths only to the Manager and the storage machine. You attest to this. See Isolating Recovery Targets.
3. Restore the chosen domain controller and make its SYSVOL authoritative, then verify that the restored data is from before the damage. You attest to this.
4. Reset and secure the domain: apply the breach resets if the security breach flag is set, seize the domain-wide operations master roles, remove the metadata of the domain's other domain controllers, repair DNS, raise the RID pool, reset the domain controller's machine account password twice, reset `krbtgt` twice while the domain is still isolated, and reset each of the domain's trusts on both sides.
5. Reconnect the domain to the production network and the rest of the forest. This is a point of no return that you acknowledge.
6. Check health: inbound replication from the live forest converges and `dcdiag` passes.
7. Confirm that the default administrator of every domain in the forest is enabled and its password known, because the next step can briefly disrupt logons that need a global

catalog.

8. Reconcile every global catalog outside the domain, one site at a time. By default, lingering objects are removed with each global catalog staying online, in advisory mode first. For small forests you can choose instead to unhost the domain's partition from every global catalog and rehost it.
9. If the security breach flag is set, re-create the domain's group managed service accounts under a new KDS root key, before any domain controller is rebuilt.
10. Add the global catalog back on the restored domain controller if it was one, take a fresh backup of it, rebuild the domain's other domain controllers, clean up and set the new DSRM password you supplied, and produce and email the recovery report of manual follow-ups.

A domain recovery is refused when:

- The domain is the forest root. Domain recovery never seizes the forest-wide roles, which live in the root, so recovering the root is a forest recovery.
- The forest has only one domain. Recovering it is a forest recovery.
- A global catalog outside the domain has no DC agent. The reconciliation runs through the agent on every global catalog.
- Another recovery in the same forest, whether of the forest or of any domain in it, is already active.

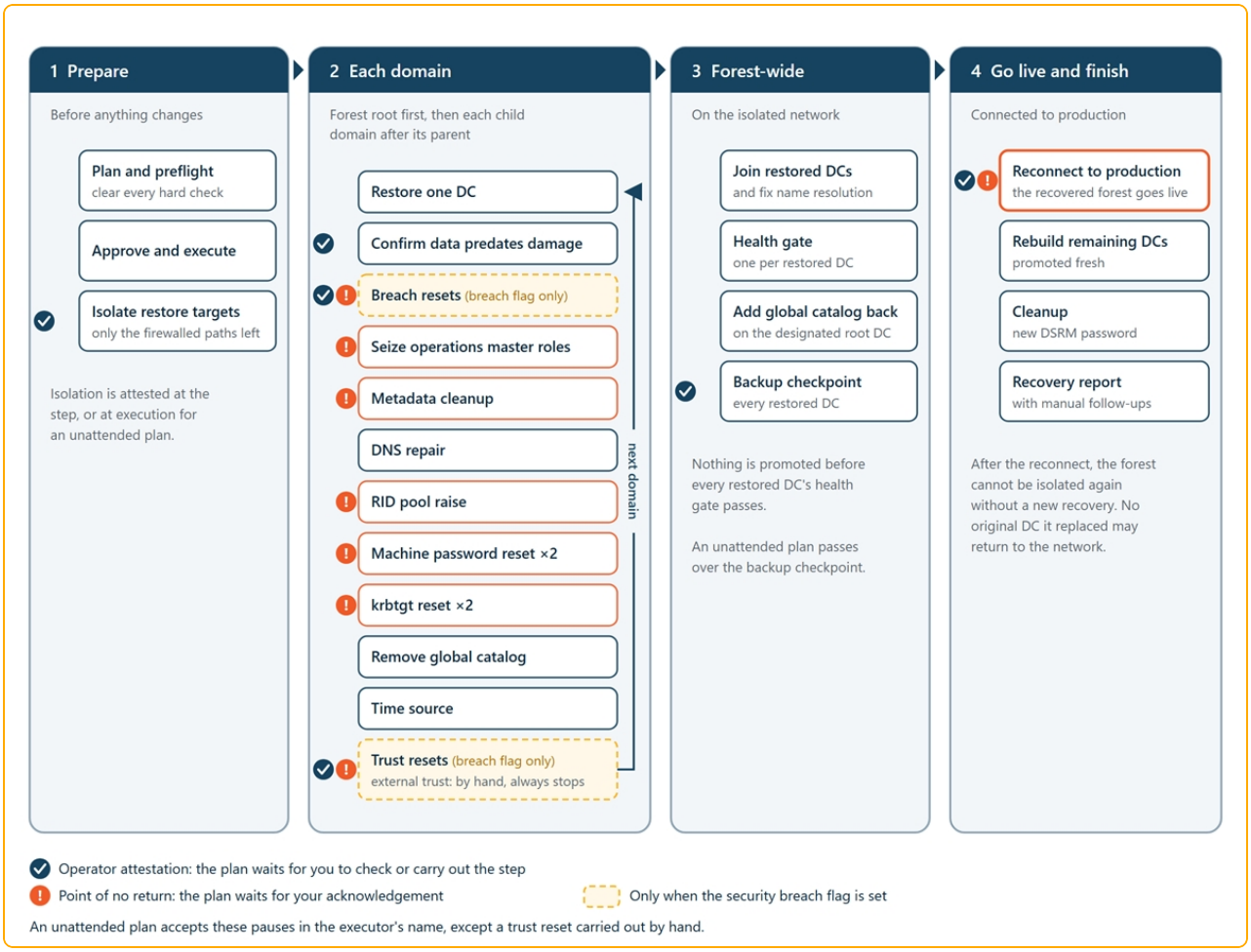
Whole forest

Use a forest recovery when the forest as a whole cannot be trusted or does not work, for example after a ransomware attack or a compromise of the forest root. Every domain is brought back from backup on isolated targets, the forest root first and each child domain after its parent. Only when the recovered forest has passed its health checks and been backed up is it reconnected to production.

In the plan, you choose for each domain controller whether it is restored, rebuilt, or excluded. An excluded domain controller is neither restored nor rebuilt: its metadata is removed from the directory along with the domain's other domain controllers, and it is not brought back, so keep the machine off the network. By default, in each domain, the connected writable PDC emulator is restored and every other domain controller is rebuilt, except domain controllers excluded from protection, which start as excluded. You also give at least one external time source reachable from the isolated network, and choose whether to set the security breach flag.

Forest recovery phases

A forest recovery runs in these phases:



- **Prepare**—Plan, clear every hard check, and approve. Isolate every restore target from production on a recovery network that still reaches the Manager and the storage machine.
- **Restore each domain**—Forest root first, then each child after its parent: restore one domain controller and verify that its data is from before the damage.
- **Fence each domain off from its old state**—Seize the operations master roles, remove the other domain controllers' metadata, repair DNS, raise the RID pool, and reset the machine account and `krbtgt` passwords. If the security breach flag is set, also apply the breach resets and reset every trust; without it, no trust is reset, and each side keeps the trust password its own restore point holds.
- **Bring the forest together**—Join the restored domain controllers on the isolated network, pass the forest health check, add the global catalog back, and take a fresh backup of every restored domain controller.
- **Go live**—Reconnect to production. This is a point of no return.

- **Finish**—Rebuild the remaining domain controllers, clean up and set the new DSRM password you supplied, and produce and email the recovery report of manual follow-ups.

Some steps are carried out or checked by you rather than by the agents; the plan waits for you to attest to them, unless it was built to run unattended. A trust reset carried out by hand, such as this forest's side of an external trust, always waits for a person. For every step and what it needs from you, see Recovering a Forest.

A forest recovery is refused when:

- The forest root domain is not known.
- No external time source is given.
- The designated global catalog is not a restored domain controller of the forest root.
- Another recovery of the same forest is already active.

Refusals shared by domain and forest recovery

These checks apply to every domain in a domain or forest plan:

- Every domain in scope must restore one writable domain controller with a usable restore point.
- A read-only domain controller can only be rebuilt or excluded.
- Every restore needs a restore point, and something to carry it out: the domain controller's own agent, or a candidate machine bound to it.
- A DC state recovery can only restore in place. Use full system recovery for a candidate machine. In a plan, a DC state recovery is non-authoritative and names nothing to roll back.
- Exactly one domain controller per domain is restored from backup, with or without the security breach flag. The others are rebuilt, and replicate the whole directory from the restored one.
- A machine bound to a restored or rebuilt domain controller must be a candidate: an enrolled DC agent on a machine that is not itself a domain controller. Each machine can be bound to only one domain controller.
- Two domain controllers cannot be planned onto the same isolated address.
- The RID pool increase must be positive.

Advisory findings include restore points from different backup windows (more than 6 hours apart), which Microsoft warns can introduce lingering objects; rebuilds with no machine bound yet; trusts whose far side must be reset by hand; and, in a forest plan without the security breach flag, trusts whose two sides are restored from points 30 days or more apart. For a domain controller restored onto a candidate machine, the plan also checks the

candidate's firmware and disks against the restore point, when the plan is built and again at approval; a blocking result refuses approval. For every check, see Preflight Check Reference.

Related topics

- Choosing a Recovery Approach
- Recovery Lifecycle
- Recovering a Forest
- Recovering a Domain

Alignment with Microsoft Guidance

Domain and forest recovery in Cygna Forest Recovery is built on Microsoft's Active Directory Forest Recovery Guide. Each step of a recovery plan carries out a step of that guide, in the guide's order. This topic was checked against the guide as published on Microsoft Learn on October 13, 2026; its overview and initial recovery pages were last revised on February 16, 2026.

Microsoft guidance and how it is applied

Microsoft guidance	How it is applied
Recover in isolation from the production network.	Every restore target is isolated before anything is restored, and you attest to it. The targets keep firewalled paths to the Manager and the storage machine only.
Restore one writable domain controller per domain, forest root first and each parent before its child: non-authoritative AD DS, authoritative SYSVOL. Verify the data before going on.	A plan restores exactly one writable domain controller per domain, in that order: by full system recovery followed by an authoritative SYSVOL synchronization, or by DC state recovery, a system state restore with authoritative SYSVOL onto the same Windows installation. You then check the restored data, or reject it and re-plan with an older restore point.
Seize the operations master roles: all five in the root, the domain-wide roles in each child.	Seized on the restored domain controller, as a point of no return.

Microsoft guidance	How it is applied
On a restored role holder, set `Repl Perform Initial Synchronizations` to 0, and back to 1 once the forest is recovered.	Set to 0 with the role seizure and back to 1 during cleanup.
Clean up the metadata of every other writable domain controller.	Removed for every domain controller in the domain that is not restored, including those without a DC agent.
Make DNS work on the restored domain controller and remove stale name server and service records.	The root points at itself, each child at the root, and stale records are removed.
Raise the available RID pool by 100,000, then invalidate the current RID pool.	Raised by 100,000 by default; you can change the amount in the plan. The current pool is always invalidated.
Reset the domain controller's computer account password twice.	Reset twice, and each reset is verified.
Reset the `krbtgt` password twice.	Reset twice; read-only domain controllers' `krbtgt` accounts are left alone. See the differences below.
After a security breach, reset trust passwords, one side at a time, with the same password on both sides.	In a forest recovery, trusts are reset only when the security breach flag is set. Each side is reset by the agent on its own domain's restored domain controller, one side at a time, to one password per trust that the Manager generates. For an external trust, only this forest's side is reset, and you carry out and attest that step.
In a forest with more than one domain, remove the global catalog from the restored domain controller; later add one in the forest root.	Removed from every domain's restored domain controller after the restore, and added back on a restored root domain controller once the forest health check passes. The step waits until that domain controller has completed its first full synchronization and serves as a global catalog.
Set the root PDC emulator to an external time source.	The plan requires at least one external time source.

Microsoft guidance	How it is applied
Join the restored domain controllers on a common isolated network, check replication health, and take a new backup of each.	A health gate on each restored domain controller, then a backup checkpoint, both before the forest is reconnected. Each gate pulls replication from the domain controller's partners, checks that its inbound replication converges and that its domain lists exactly its restored domain controller, and runs `dcdiag`. Nothing is promoted before every gate passes.
Redeploy the remaining domain controllers by installing AD DS.	Each is rebuilt by promotion, with the roles and features its last backup recorded reinstalled first.
Clean up: revert DNS, delete records of unrecovered domain controllers, move operations master roles, re-create missing objects, and restore external trusts.	The recovery report lists these as follow-ups. See After a Recovery.

What Cygna Forest Recovery adds

Microsoft's guide is a template you adapt and run by hand. Cygna Forest Recovery adds:

- **Preflight checks**–Hard checks block approval until cleared, for example a second restored domain controller in one domain or a restore point past the tombstone lifetime. See Preflight Check Reference.
- **Isolation attestation**–A named operator attests that the targets are isolated before anything is restored.
- **Points of no return**–Steps that cannot be undone wait for a typed acknowledgment.
- **Audit trail**–Every decision, check result, and state change is recorded.
- **Breach resets**–The security breach flag adds the resets Microsoft describes for a compromise, before any domain controller is rebuilt.
- **Unattended execution**–A plan can accept its pauses in the executing operator's name. A trust reset carried out by hand, such as this forest's side of an external trust, still stops for a person.
- **Recovery report**–What happened to each domain controller, and the follow-ups left to you.

Where Cygna Forest Recovery differs, and why

- **krbtgt resets back to back**–Microsoft's reset procedure says to wait 10 hours, the default maximum ticket lifetime, between the two resets, so that tickets issued under the first new key can expire. By default, Cygna Forest Recovery runs the second reset as soon as the first is acknowledged. Both resets run while the forest is still isolated, and reconnecting waits for the second, so no client holds a ticket issued between them. If your policy requires the 10-hour wait, contact Cygna Labs support before you plan the recovery, so that the plan is created with that gap.
- **RID pool always invalidated**–Microsoft notes that a system state restore invalidates the pool by itself. A full system recovery does not, so the pool is invalidated explicitly every time. Invalidating twice does no harm.
- **Trust resets in every domain recovery**–Microsoft describes trust resets for a security breach. A forest recovery follows this: it resets trusts only when the security breach flag is set. Without the flag, each side of a trust keeps the trust password its own restore point holds, and the plan warns when the two sides' restore points are 30 days or more apart, because Windows rotates the trust password every 30 days and the two may no longer match. A domain recovery, which brings one domain back into a forest that kept running, resets every trust of the domain whether or not the flag is set.
- **Isolation keeps two paths open**–For a physical domain controller, Microsoft's guide disconnects the network cable. The agents run every step, so the targets keep firewalled paths to the Manager and the storage machine. See Isolating Recovery Targets.
- **Redeployment by promotion only**–Microsoft also describes virtualized domain controller cloning and install from media. Rebuilds promote each domain controller and replicate the directory over the network.

DC state recovery and authoritative restore

A DC state recovery of a single domain controller follows the authoritative restore procedure in Microsoft's Restore user accounts and groups in AD (checked as published on Microsoft Learn on October 21, 2026; last revised February 12, 2026):

- **System state restore in DSRM, then marking**–The domain controller is restarted into Directory Services Restore Mode, system state is restored, and each subtree or object is marked with `ntdsutil authoritative restore` before the normal restart.
- **Back-links with ldifde**–After the restart, the back-link files `ntdsutil` writes are imported with `ldifde` on the restored domain controller, so the marked objects' group memberships come back.
- **Subtrees and objects, never the whole database**–The recovery marks only what you name. It refuses a whole partition, the schema, the domain controllers, and `CN=System`,

so trust and domain controller passwords are not rolled back. Microsoft advises against restoring the whole database authoritatively for the same reason.

Where it differs:

- **Inbound replication stays on**—Microsoft's procedure turns off inbound replication on the recovery domain controller while the restored objects replicate out. A DC state recovery does not; the marked objects carry raised version numbers, and the agent reads them back from replication metadata before the run counts as restored.
- **Marks are verified**—The run records the evidence of each mark and the hash of each back-link file, imports only those files, and reports the recovery as restored only when the marks are read back.
- **krbtgt after rolling back CN=Users**—When a DC state recovery puts `krbtgt` back to the backup and you choose the reset, `krbtgt` is reset twice once the recovery validates. The second reset waits until the first has reached every writable domain controller, not for the 10-hour ticket lifetime, so tickets issued under the backup's keys stop working and users and services sign in again.



Note: Microsoft revises its guidance from time to time. The mapping on this page is Cygna Labs Corp.'s own reading of it; it does not imply review or endorsement by Microsoft.

Restore Methods

Each domain controller in a recovery is brought back by one of five methods. The method depends on whether the domain controller's Windows still starts, whether its hardware is still there, and what kind of backup you have.

DC state recovery

Use DC state recovery when the domain controller's Windows still starts and only its directory needs to go back. It is a supported restore through Directory Services Restore Mode, not a virtual machine snapshot revert, so it does not cause USN rollback.

When you plan a DC state recovery of one domain controller, you name the subtrees and objects to roll back, and they are restored authoritatively: their changes since the backup are undone on every domain controller of the domain. When a domain or forest recovery plan restores a domain controller by DC state recovery, nothing is marked; the restore is non-authoritative.

1. The DC agent copies the chosen backup from the vault to a local volume on the domain controller and verifies it there. The vault read lease is released before the machine restarts, so the restore does not depend on networking in Directory Services Restore Mode.

2. The DC agent sets the machine to start in Directory Services Restore Mode for the next start only, and restarts it. The machine's normal boot entry is never changed, so a power loss or failure at this point ends in a normal start.
3. In Directory Services Restore Mode, the DC agent runs a `wbadmin` system state recovery.
4. For a single-DC recovery, still in Directory Services Restore Mode, the DC agent marks each named subtree and object authoritative with `ntdsutil`.
5. The machine restarts normally, and the DC agent reconnects and validates the result. For a single-DC recovery, it first imports the back-links `ntdsutil` wrote, so that group memberships come back, and reads the marked objects' replication metadata to confirm the marks.

DC state recovery works from either backup type. It needs:

- A connected DC agent on the machine, because the agent drives the restore.
- The backup's Windows build and revision to match the machine as it runs now. DC state cannot be put back across a Windows Update, so after patching, older backups cannot be restored onto the machine this way until a new backup is taken.
- Free space on a local volume of about 1.15 times the backup's size, because the agent copies the backup to the machine first and restores from that copy.

DC state recovery always restores onto the same machine. It cannot restore onto another one.

Full system recovery in place

Use full system recovery in place when the domain controller's operating system itself must go back, for example because it is damaged or encrypted, but Windows still starts one last time. The whole server, operating system and directory, is restored from a full system backup onto the same machine's existing disks.

1. The DC agent fetches the recovery image for the backup's operating system and stages it on the system volume.
2. The machine restarts once into the recovery image, from a RAM disk.
3. The recovery agent restores every critical volume from the vault onto the existing disks.
4. When the Manager has recorded success, the machine restarts into the restored system, and the restored DC agent reconnects.

An in-place recovery restores onto the existing disk layout and cannot repartition. If disks were added, resized, replaced, or converted between partition styles since the backup, choose a backup taken after the change. For how the recovery image works, see Recovery Images.

Full system recovery onto a candidate

Use full system recovery onto a candidate when the domain controller's Windows will not start, or its hardware is gone. A candidate is a machine that is not a domain controller, with the DC agent installed and enrolled, that receives the restored server. The candidate's own agent stages the recovery image, and the candidate comes up as the original domain controller: same name, same security identifier, same configuration.

Because of that, the original domain controller must be powered off or isolated before you execute. Two machines with one domain controller's identity damage a domain rather than recover it. Preflight asks the candidate to check every address the original used, and refuses approval while the original still answers.

A candidate needs at least as many disks as the backup, each the same size or larger, and the same firmware type (UEFI or BIOS) as the original. For virtual machines, the `cdr dc rebuild` command can create a Hyper-V virtual machine, or write a VMware ESXi script, shaped like the backup, ready for you to install Windows Server and the DC agent onto. To skip installing anything, recover onto a new VM instead, described below.

The restored system starts with the network settings in the backup. On different hardware, Windows puts the old static address on an adapter that no longer exists, so the restored domain controller would start on DHCP. To prevent that, the Manager by default plans the backup's own static address for the restored system, and the restored DC agent applies it to the new adapter at first start. This needs a backup that recorded exactly one static address. For a single-DC full system recovery you can also choose the restored system's address yourself.

After any full system recovery, the restored domain controller's DC agent reconnects to the Manager and checks that the domain controller works: AD DS running, SYSVOL ready, replication converging, and dcdiag clean. The run completes or fails on its answer.

Full system recovery onto a new VM

Use full system recovery onto a new VM in the same situations as a candidate, when you would rather not prepare a machine first. The target is an empty virtual machine: nothing is installed on it, not even Windows.

1. When you execute the recovery, the storage agent builds a recovery ISO for this run only, from the installation media's boot files and the recovery image. The ISO carries only the run's single-use enrollment token, the Manager's address, and the Manager's CA fingerprint.
2. On Hyper-V, `cdr dc rebuild --restore-run` builds the VM shaped from the run's restore point, attaches the ISO, and starts it. On another hypervisor, you build the VM and attach the ISO yourself.

3. The VM boots the ISO, and the recovery agent enrolls with the Manager, which then hands it the vault credential. Before anything is written, the recovery agent checks the VM's firmware and disks against the backup, and stops with nothing written if they do not fit.
4. The recovery agent restores every critical volume from the vault, and the VM restarts into the restored domain controller, as after a recovery onto a candidate.

A recovery onto a new VM needs:

- A full system backup.
- The original domain controller powered off. While its agent is connected to the Manager, preflight blocks approval and execution is refused.
- The installation ISO for the backup's operating system in the OS media library.
- A recovery image for the backup's operating system.
- Hyper-V for the automated VM build.

The restored system's address is planned as for a candidate. Recovery onto a new VM is a single-DC recovery; domain and forest recovery plans do not offer it. A test restore uses the same path on an Isolated Recovery Host, and then isolates and checks the restored copy instead of putting it into production; see Test Restores. For the procedure, see Full System Recovery onto a New VM.

Rebuild

A rebuild does not restore the domain controller at all. Its metadata is removed from the directory, and a fresh domain controller is promoted in its place, which then replicates the directory from the domain's other domain controllers. A rebuild needs no backup of that machine, so it is the path when its backups are stale, past the tombstone lifetime, or suspect.

A rebuild is required for:

- Read-only domain controllers, which are never restored.
- A domain controller whose operations master roles were seized by another domain controller after its backup.

In a domain or forest plan, rebuild is a disposition you set per domain controller, and the plan carries it out after the domain is back. It promotes either a candidate machine you bind to the domain controller in the plan, which is renamed to match, or the domain controller's own machine, reinstalled and connecting with the DC agent under its own name. Before promoting it, the plan reinstalls the Windows roles and features the domain controller's last backup recorded, and names any that would not install. The roles come back without their configuration or data, and installed programs are not reinstalled; insecure legacy features such as SMB 1.0 are left out. The plan names all of these for each rebuilt domain controller, and the recovery report lists them as follow-ups. To rebuild a single domain controller

outside a plan, remove its metadata with `cdr cleanup dc` and promote a replacement with your usual tools.

Authoritative and non-authoritative restores

A restore is non-authoritative for Active Directory unless you name what to roll back: the restored domain controller accepts changes from its replication partners rather than overwriting them.

- In a DC state recovery of a single domain controller, the subtrees and objects you name are marked authoritative, so they replicate out to every domain controller of the domain as they were at the backup. Everything else, including SYSVOL, converges from the domain's healthy domain controllers. Group Policy objects roll back in the directory, but their SYSVOL files do not. See Restoring a DC with DC State Recovery.
- In a single-DC full system recovery, both Active Directory and SYSVOL converge from the domain's healthy domain controllers.
- In a domain or forest recovery, the first restored domain controller in each domain has its SYSVOL made authoritative, because there are no healthy partners to converge from. This happens on exactly one domain controller per domain.
- In a test restore, the restored copy is started on a private network as the first domain controller of its domain, so its SYSVOL is made authoritative. This affects only the isolated copy, which is deleted afterward.

A DC state recovery never marks a whole partition, the schema, a `krbtgt` account on its own, the domain controllers, or `CN=System` and the objects directly in it. To recover individual objects and attributes without restoring a domain controller, use the Recovery for Active Directory module of Cygna Auditor; see Related Cygna Products.

Comparing the methods

Method	Machine state	Backup type	Target
DC state recovery	Windows starts; same build and revision as the backup	Full system or DC state	Same machine
Full system recovery in place	Windows starts one last time; disk layout unchanged	Full system	Same machine
Full system recovery onto a candidate	Original will not start or is gone; must be off or isolated	Full system	Candidate machine with the DC agent

Method	Machine state	Backup type	Target
Full system recovery onto a new VM	Original will not start or is gone; must be powered off	Full system	Empty VM booted from the run's recovery ISO
Rebuild	Any	None needed	A machine promoted fresh

For everything each method needs at recovery time, see Recovery Requirements.

Choosing a Recovery Approach

The right recovery depends on how much of the directory you can still trust and what state the machines are in. Start from the widest damage: if you cannot trust the forest, recovering one domain controller does not help, however healthy the backup.

Decide the scope first

- **The rest of the domain is healthy and trustworthy**—Recover the single domain controller.
- **One domain is damaged or compromised, and the other domains are healthy**—Recover the domain. This is not available for the forest root domain or for a forest with only one domain; use forest recovery for those.
- **The forest as a whole cannot be trusted**—Recover the forest. This includes any compromise of the forest root, a compromise whose reach you do not know, and ransomware that has reached domain controllers.

Situations

Situation	Approach
Changes to accounts, groups, or other objects have replicated across a healthy domain and must be undone, for example a mass password reset	Single-DC recovery by **DC state recovery** of one domain controller, naming the subtrees and objects to roll back. The backup must match the machine's current Windows build and revision. If you roll back `CN=Users`, choose the `krbtgt` reset.
One domain controller's operating system or	Single-DC recovery by **full system recovery in place**, from a full system backup taken since any change to the

Situation	Approach
directory is damaged, but Windows still starts, and the rest of the domain is healthy	machine's disks. Alternatively, rebuild it.
One domain controller will not start	Single-DC recovery by **full system recovery onto a candidate**: a new physical or virtual machine with the DC agent enrolled. Power off or isolate the original first. Alternatively, rebuild it.
One domain controller's hardware is lost	**Full system recovery onto a candidate** with matching firmware and disks at least as large as the backup's. `cdr dc rebuild` can create a suitable virtual machine. Alternatively, rebuild it.
The domain controller is read-only, or its roles were seized elsewhere after its backup	**Rebuild**. Restoring is refused.
Every backup of the domain controller is stale, past the tombstone lifetime, or suspect, and the domain is healthy	**Rebuild**. A rebuild needs no backup.
One domain (not the forest root) is corrupted or compromised; the other domains are healthy	**Domain recovery**. If it is a security incident, set the security breach flag and choose restore points from before the compromise.
The forest root domain is corrupted or compromised, or the forest has only one domain	**Forest recovery**. Domain recovery is refused for these.
The whole forest is compromised, or ransomware has reached domain controllers	**Forest recovery** with the security breach flag set, from restore points taken before the compromise, aligned to one backup window across domains.

Restore or rebuild

Within a recovery, each domain controller is either restored or rebuilt. Restoring is faster where the directory is large and replicating it again over the network would be slow. Rebuilding avoids every constraint a restore has: no backup is needed, the hardware does not have to match, and the Windows build can be newer. When the rest of the domain is healthy, rebuilding a single domain controller is often the simpler choice.

In domain and forest recovery, a plan restores exactly one domain controller per domain and rebuilds the rest; a plan that restores more cannot be approved. A rebuilt domain controller brings the whole directory back by replication from the restored one, so users, groups, memberships, and permissions lose nothing. What a rebuild does not bring back is what was on the machine itself: its Windows roles and features are reinstalled from its last backup's record, but their configuration and its installed programs are not. If a domain controller runs a role whose configuration you cannot rebuild by other means, choose it as the domain's restored domain controller.

The security breach flag

Set the **Security breach** option in a domain or forest plan when you are recovering from an attack rather than from an accident. The flag must be set before the plan is approved, because it changes what the plan contains:

- Breach resets are added on every restored domain controller, before any other domain controller is promoted.
- In a forest recovery, every trust is reset, including trusts that leave the forest. Without the flag, a forest recovery resets no trust: each side keeps the trust password its own restore point holds. A domain recovery resets the domain's trusts either way.
- In a domain recovery, the domain's group managed service accounts are re-created under a new KDS root key.

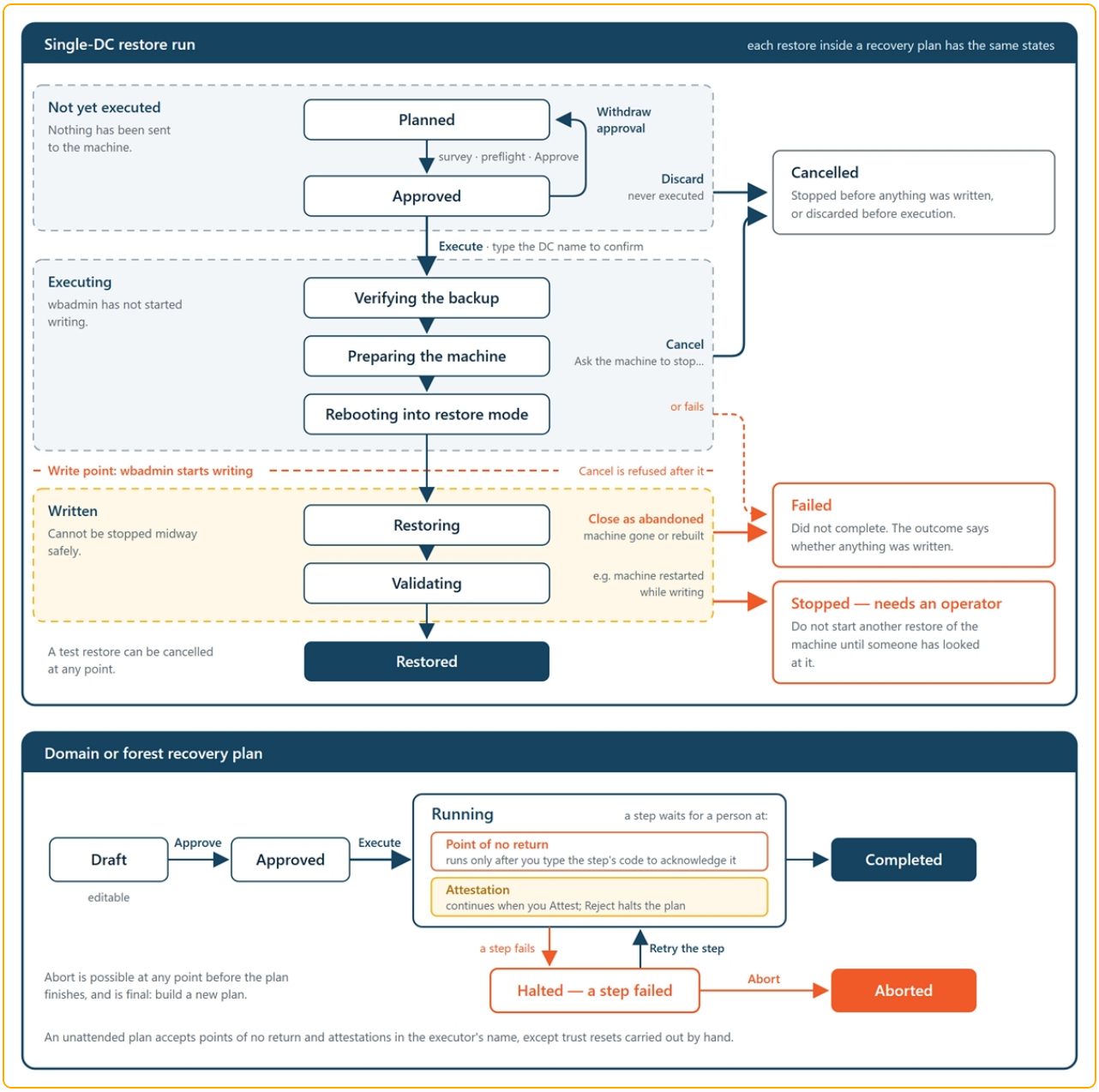
The breach flag does not choose your restore points. Pick restore points from before the compromise, allowing for the time an attacker may have been present before being detected, and verify the restored data when the plan asks you to. If the restored data is not clean, reject that step and plan again with an older restore point. To keep older restore points available, see Retention.

Related topics

- Recovery Scopes
- Restore Methods
- Recovery Procedures

Recovery Lifecycle

Every recovery moves through the same stages: plan, preflight, approve, execute, and monitor. The console shows these as a stepper at the top of each recovery's page. This topic explains each stage, the states a recovery can be in, and the ways to stop one, so that you know before you start what you can still undo at each point.



Plan

You create a restore run (single-DC recovery) or a recovery plan (domain or forest recovery) on the **Recovery Plans** screen. Planning writes nothing to any machine. It

records your choices and evaluates preflight. A forest or domain plan starts as a **Draft**, which you can edit until it is approved. A forest can have only one active recovery at a time, whether of the forest or of a domain in it.

Preflight

The Manager checks the plan against the estate and lists every finding. Hard findings block approval until they are cleared. Advisory findings do not block approval; they say what the restore will cost or what to watch for. A restore run can also raise findings that need your acceptance: something that failed from where the Manager looked but may not fail from where the recovery environment runs, such as a storage machine name the target cannot resolve now. The approver accepts each one by its code, or cancels the run. Preflight is evaluated when the plan is built, and again at approval and execution. In a domain or forest plan, approval also re-checks the firmware and disks of each candidate machine against the backup restored onto it, and is refused if one of those checks blocks. See Preflight Check Reference.

Survey

For a full system recovery, the Manager asks the target machine's DC agent to describe its hardware as it is now: firmware, memory, disks, free space, network adapters, whether it can reach the storage machine at the address the recovery environment will use, and, for a candidate, whether the original domain controller still answers. The survey is taken when the run is planned, approved, and executed, and you can ask again from the run's page with **Ask the machine again**. Preflight checks the backup against the survey.

Approve

A Restore Operator approves the plan with `plan.approve`. Approval records that the plan has been reviewed, under the approver's name. The Manager refuses to approve a plan with an outstanding hard finding, or a restore run with a finding nobody has accepted. Accepted findings are recorded on the run under the approver's name. Until execution, a single-DC approval can be withdrawn with **Withdraw approval**.

Execute

A Restore Operator executes the plan with `plan.execute`. Execution asks you to type a confirmation: the domain controller's short name in capitals for a single-DC restore, or the forest or domain name for a recovery plan. The checks that can run only against the live estate run now, including the hash verification of each backup before any read access is granted.

A recovery plan built to run unattended is confirmed at execution: whoever executes it types the forest or domain name and attests, then and there, that the restore targets are isolated. From then on the plan does not stop for a person at points of no return,

attestations, the backup checkpoint, or restore preflight findings; each is accepted in the name of the operator who executes the plan and recorded. It still halts on a failed step, waits for a missing credential or DSRM password, waits out a krbtgt gap the plan set, and stops at every trust reset carried out by hand, such as this forest's side of an external trust. Whoever holds `plan.execute` can take a running plan back, after which every pause waits for a person again. See Recovering a Forest.

Monitor

While a recovery runs, its page shows each step or phase as it happens. A domain controller being restored is expected to go quiet while it restarts; the console says how long to expect, and the agent reports the whole timeline when it returns. A banner across the console's screens shows recoveries in progress and any recovery waiting for you; the Dashboard shows them in its own **Recovery in progress** panel instead. Times on a run's timeline are shown on the Manager's clock, even when a machine's own clock is off.

A recovery plan can also email its own notification addresses when it halts, is aborted, or waits for a person, and sends them the recovery report when it completes. The report is built from the plan's own records whenever it is asked for, so it is also available on the plan's page and from `cdr recovery report` at any time after the plan starts.

Restore run states

A single-DC restore, and each restore inside a recovery plan, has one of these states:

State	Meaning
Planned	Recorded, with preflight evaluated. Nothing sent to the machine.
Approved	Reviewed and ready to execute.
Verifying the backup	The backup's hash is being verified and its read lease granted.
Preparing the machine	The backup is being copied to the machine (DC state recovery), or the recovery image is being staged (full system recovery).
Rebooting into restore mode	The machine is restarting into Directory Services Restore Mode or into the recovery image.
Restoring	`wbadmin` is restoring DC state or the volumes.
Validating	The restored domain controller is being checked: AD DS running and SYSVOL ready, then replication converging and

State	Meaning
	dcdiag clean. After a full system recovery, the restored domain controller's own agent makes these checks once it reconnects, waiting up to 15 minutes for replication.
Restored	Completed and validated.
Failed	The restore did not complete. The outcome says whether anything was written.
Cancelled	Stopped before anything was written, or discarded before execution.
Stopped — needs an operator	The restore stopped in a state that cannot safely be resumed or stopped automatically, for example because the machine restarted while the restore was writing. Do not start another restore of the machine until someone has looked at it. The ways forward are a fresh restore from a different restore point, or a rebuild.

Recovery plan states

A domain or forest plan moves through **Draft**, **Approved**, and **Running**, and ends as **Completed** or **Aborted**. If a step fails, the plan becomes **Halted — a step failed**, and nothing more is issued until you retry the step or abort the plan. A retry runs the step again from its checkpoint, without repeating what it already did, and keeps its point-of-no-return acknowledgment. Within a running plan, each step is **Pending**, **Running**, **Done**, or **Failed**, or waiting:

- **Waiting for acknowledgement**—A point of no return. See below.
- **Waiting for you**—Something only a person can do or check. Attest to it, or reject it, which halts the plan. A restore whose preflight raised a finding only a person can accept is decided on its restore run instead.
- **Waiting for a credential**—The step cannot run without a domain's administrator credential or the new DSRM password, and the Manager holds none. It runs by itself once one is supplied.
- **Waiting (krbtgt)**—Only in a plan created with a gap between the two `krbtgt` resets. By default there is none: both resets run while the directory is isolated, and going live waits for the second. The second reset runs by itself when the gap ends, or earlier if you override the wait with a reason.
- **Waiting for its machine to be built**—A rebuild is waiting for the machine it promotes to

connect with the DC agent for the first time.

- **Restarting**–A rebuilt machine is restarting between steps. Nothing is needed from you.

How steps can be stopped

Each step is classified by what stopping at that point means. A single-DC restore's **In progress** card shows the label for where the run is now. For a recovery plan, the abort dialog says which points of no return are behind the plan and which changes stay to finish or undo by hand.

Label	Meaning
Safe to stop – nothing was changed	Stopping here leaves the machine as it was.
Stopping needs cleanup – the machine was changed	The step changes the machine. If you stop after it, the machine must be restored again or rebuilt.
Point of no return – runs only after an operator acknowledges it	The step cannot be undone, for example seizing roles, removing metadata, or resetting `krbtgt`. The plan waits until you acknowledge it by typing the step's code as the console shows it. The acknowledgment is recorded in the audit trail.
Cannot be stopped or resumed – needs an operator	Applies to a single-DC restore while `wbadmin` is writing. A DC state recovery cannot safely be stopped midway.

Attestations

Some steps are checks only a person can make. The plan waits, shows a prompt, and continues when you select **Attest**. The attestations are:

- **Isolation**–Nothing of production can reach any restore target, it has no route out, and its only links are firewalled paths to the Manager's agent port and the storage machine's SMB, enforced outside the target. See Isolating Recovery Targets.
- **Pre-damage check**–You signed in to the restored domain controller and the corruption or encryption is absent.
- **Logon continuity**–In a domain recovery, the default administrator of every domain in the forest is enabled and its password is known.

- **Reconnect**—The recovered forest or domain goes live on the production network, and none of the original machines the recovery replaced can rejoin it. This is also a point of no return.

Steps that the agents do not carry out themselves also wait for you in the same way: the step shows what to do, you carry it out, and you attest. If a check fails, select **Reject** and give a reason. Rejecting halts the plan.

In an unattended plan, every attestation is accepted without asking, and a step you would carry out by hand, such as the backup checkpoint, is passed over and listed in the recovery report. Trust resets carried out by hand are the exception: they always wait for a person. Trust resets the agents carry out are accepted like any other point of no return.

Stopping a recovery

Action	Applies to	What it does
Discard (**Discard this plan…**)	A single-DC restore that was planned or approved but never executed	Closes the plan as cancelled. Nothing was sent to the machine, and it can be planned again straight away. Needs `plan.approve`. Not available for a restore that belongs to a recovery plan; abort the plan instead.
Cancel (**Ask the machine to stop…**)	A single-DC restore in progress, before `wbadmin` starts writing	The machine abandons the restore. Once writing has started, the Manager refuses, because stopping midway cannot be done safely. It also refuses a DC state recovery whose machine has restarted into DSRM and is away, because the restore may be running there. Needs `plan.execute`.
Abandon (**Close as abandoned…**)	A single-DC restore past the write point whose machine is gone or will be rebuilt	Closes the run as failed with the reason you give. Nothing is sent to the machine, and whatever was written stays there. Do not use that machine as a domain controller until it is rebuilt or its transcript has been read. Needs `plan.execute`.
Abort (**Abort…**)	A domain or forest plan, at any point before it finishes	Stops the plan where it is, with the reason you give. Abort is final: an aborted plan cannot be resumed, and you build a new one. Steps that already ran are not undone; the dialog says what aborting at this point leaves. Each restore the plan started is asked to stop; one already writing carries on. Needs `plan.execute`.

Credentials during a plan

A domain or forest plan acts in each domain as an account you supply on the plan's page while it is approved or running: a Domain Admin for each domain, and an Enterprise Admin and Schema Admin for the forest root. A domain recovery also needs one for each partner domain whose live domain controller resets its side of a trust. The plan's page shows which domains the Manager holds a credential for; it never reads one back out. The Manager holds each credential in memory for that plan only, never writes it to disk or shows it again, and drops it when the plan ends. It holds the new DSRM password you choose for the recovered domain controllers the same way. If the Manager restarts, it forgets them, and a step that needs one waits in **Waiting for a credential** until it is supplied again.

Audit

Every stage is recorded in the audit trail under the operator's name: planning, approval and its withdrawal, accepted preflight findings, execution, each gate result, acknowledgments, attestations and rejections, acceptances made by an unattended plan, credential supplies (never the secret), retries, wait overrides, cancellations, discards, abandonments, aborts, notification and report emails, and each vault read lease. See Auditing Operator Activity.

Related topics

- Recovery Plans
- Recovery Procedures

Recovery Requirements

This topic lists what each restore method and recovery scope needs at recovery time. Preflight checks almost all of it and tells you what is missing, but most of these needs are cheaper to meet before an incident than during one. For the requirements of the product itself, see System Requirements.

Requirements for every restore

- **A usable restore point**—Still in the vault, younger than the domain's tombstone lifetime, with a captured version identifier, taken by the DC agent (so it has a manifest), and with SYSVOL replicated by DFSR. Domains that still use FRS for SYSVOL must migrate to DFSR and take fresh backups first.
- **A writable domain controller**—Read-only domain controllers are rebuilt, never restored. A domain controller whose operations master roles were seized elsewhere after its backup must also be rebuilt.

- **No backup running on the machine**–A restore restarts the machine, which would abandon the backup. Let it finish or cancel it.
- **A connected agent on the machine that carries out the restore**–The domain controller's own DC agent, or the candidate's.
- **A verifiable backup**–The backup must still match the hash recorded when it was committed.
- **The storage machine and Manager reachable**–From the machine being restored: SMB (TCP 445) to the storage machine, and TCP 5211 to the Manager. See Ports and Protocols.
- **The Restore Operator role**–Approval needs `plan.approve` and execution needs `plan.execute`. Both belong to the Restore Operator role and to no other role.

DC state recovery

- Windows on the domain controller still starts, and its DC agent is connected.
- The restore point was taken on the same Windows build and update revision the machine runs now. A backup taken before the machine was last patched cannot be restored onto it this way. A backup from a different Windows release never can.
- Free space on the domain controller for a local copy of the backup. Before the domain controller restarts into Directory Services Restore Mode, the DC agent copies the backup from storage to the domain controller's own disk, and the restore then runs from that copy. The copy needs about 1.15 times the backup's size, on a fixed volume other than the system volume. The DC agent setting **Restore staging volume** chooses the volume; by default it uses the fixed, non-system volume with the most free space.
- Either backup type.
- A DC agent as current as the Manager and running as its Windows service, so that it can hold back the restart wbadmin starts and mark the named subtrees and objects authoritative.
- The distinguished names of the subtrees and objects to roll back. To roll back `CN=Users` with the `krbtgt` reset, a domain administrator's user name and password at approval.

Full system recovery

Full system recovery, in place or onto a candidate, needs everything below. The candidate column notes where the two differ.

Requirement	Details	Onto a candidate
Backup type	A full system backup. DC state backups cannot do a full system recovery.	Same.
Recovery image	A recovery image for the backup's operating system that has not failed. This needs the storage agent's **Recovery image folder** set and a matching Windows Server ISO in its media folder. See Recovery Images.	Same.
Target machine	The domain controller's own Windows starts one last time, with its DC agent connected, so the agent can stage the recovery image.	A machine that is not a domain controller, with the DC agent installed and enrolled. The original domain controller must be powered off or isolated.
Disks	The disk layout is unchanged since the backup. An in-place recovery cannot repartition, so disks that were added, resized, replaced, or converted need a newer backup.	At least as many disks as the backup, each the same size or larger.
Firmware	Unchanged since the backup.	The same firmware type as the backup: UEFI with GPT, or BIOS with MBR. Both are supported. On Hyper-V, a generation 2 virtual machine is UEFI and generation 1 is BIOS.
Memory	Enough to load the whole recovery image into a RAM disk: the image size plus about 1 GB.	Same. Less memory than the original works, but the domain controller may run slowly.
Staging space	Free space on the system volume for the	Same.

Requirement	Details	Onto a candidate
	recovery image: about 1.1 times the image's size.	
Name resolution	None, by default. The recovery environment reaches the vault at the storage agent's advertised address, which is the storage machine's IP address unless the storage machine has more than one address or its advertised host is set to a name. It reaches the Manager at the address the machine's DC agent uses, resolved to an IPv4 address before the restore starts. When the vault is reached by name, the recovery environment can resolve it through DNS only; a hosts file, LLMNR, or NetBIOS is not enough. The restored domain controller's DC agent dials the Manager address it was installed with.	Same.
Network address	The recovery environment takes its address from DHCP unless the plan gives it the machine's own static address or another one. With no DHCP server on the network and no planned address, it refuses before writing and restarts back into Windows. The recovery environment needs a driver for the machine's network adapter. Add any driver it lacks to the storage agent's **Driver folder**, so that it is built into the recovery image.	Same.
Clock	Within five minutes of the Manager's. A restored domain controller trusts its own clock until it synchronizes, and Kerberos refuses beyond five minutes. On a virtual machine, correct the host's time.	Same.
BitLocker	If the backed-up volumes were BitLocker-protected, have their recovery keys at hand.	Same. A TPM protector belongs to the original machine's TPM, so the restored

Requirement	Details	Onto a candidate
		volumes are likely to ask for the keys.
Storage controller	Unchanged.	The restored Windows needs a driver for the candidate's boot controller, or it stops with `INACCESSIBLE_BOOT_DEVICE`. If the candidate's boot controller differs from the original's, install its driver on the original domain controller before the backup. Built-in drivers cover common virtual controllers.

Windows binds a static address to the adapter it was set on, so the original's address does not follow it onto a different adapter by itself. Unless you choose otherwise, the Manager puts the backup's own static address back on the restored system: onto a candidate, and in place when the machine's latest description says which adapter to use. This needs a backup that recorded exactly one static address. Otherwise, set the address by hand afterward.

The restored domain controller comes back running the DC agent its backup held. The recovery environment replaces it with the agent build that staged the restore, when the recovery image can do so. A plan step is not sent to an agent older than the Manager's agent package; update such an agent with `cdr agent update` or from the **Agents** screen, and retry the step. See Maintaining Agents.

Rebuild

- No backup of the domain controller is needed to rebuild it.
- In a domain or forest plan, a machine with Windows Server and the DC agent enrolled: either a candidate bound to the domain controller in the plan, which the plan renames, or the domain controller's own machine reinstalled under its name. The rebuild waits until it connects. Each machine can be bound to only one domain controller.
- The domain controller the rebuild takes its DNS from must have a known address: an isolated address in the plan, or the static address recorded in its chosen backup.

- To have its Windows roles and features reinstalled before promotion, the domain controller needs a backup taken by a current DC agent, which records them. Their configuration and data, and installed programs, are not carried; the plan names them for each rebuilt domain controller.

A domain or forest plan restores exactly one domain controller per domain and rebuilds the others. A plan that restores two in one domain cannot be approved.

Credentials at recovery time

Cygna Forest Recovery stores no directory credential. Have these available, stored offline, before you need them:

Credential	When it is needed
Your Cygna Forest Recovery operator account with the Restore Operator role	Every recovery. Operator accounts do not depend on Active Directory. Keep the break-glass credential offline as a fallback.
An Enterprise Admin and Schema Admin account of the forest root	Forest recovery, supplied to the plan for the forest root.
A Domain Admin account of each other domain in scope	Domain and forest recovery, supplied to the plan per domain.
The password of each domain's built-in Administrator account (RID 500), as it was at the restore point	Signing in to a restored domain controller to check the restored data, and, in a domain recovery, confirming that every domain's default administrator can sign in while the global catalogs are reconciled. The account is often not named Administrator; the **Domains** screen shows its name.
A new Directory Services Restore Mode (DSRM) administrator password	Domain and forest recovery. You choose it, and the plan sets it on every domain controller it restores or rebuilds. It must meet the domain's password policy. Store it as your password policy requires:Cygna Forest Recovery never shows it again.

A plan holds the credentials and the DSRM password you supply in memory only, for that plan, and drops them when the plan ends or the Manager restarts. A step that cannot run without one waits until it is supplied. A single-DC restore does not ask for a directory credential.

Network and isolation

Domain and forest recovery restore domain controllers on isolated targets before they rejoin production. Plan the isolation in advance:

- No domain controller, member, or client of the directory being recovered, and no other production machine, can reach the restore targets, and they have no route out.
- Their only links are two firewalled paths: to the Manager's agent port and to the storage machine's SMB, enforced outside the targets. Isolated does not mean unplugged.
- For a forest recovery, the root PDC emulator needs to reach at least one external time source (NTP), and the restored domain controllers of different domains must reach each other.
- Restored domain controllers keep their original addresses unless the plan gives them others, so every original must be powered off or isolated.

For the paths, layouts, and a checklist, see Isolating Recovery Targets.

Tombstone lifetime

Every restore point must be younger than its domain's tombstone lifetime, 180 days by default. Keep backups current well inside it: the **Domains** screen shows the headroom left for each domain. Keep the retention floor at or above the tombstone lifetime so that older clean backups survive a full vault. See Coverage and Readiness and Retention.

Related topics

- Preflight Check Reference
- Accounts and Permissions
- Isolating Recovery Targets

Isolating Recovery Targets

A forest or domain recovery restores domain controllers from backups that are not yet trusted, and fences the recovered directory off from its old state before it rejoins production. Until then, the restore targets must be isolated: nothing of production may reach them, and they must have no route out. The plan's first operator step (FR-02 in a forest recovery, DR-02 in a domain recovery) asks you to attest that this is so.

Isolated does not mean unplugged. The restore itself needs the network: the target's agent reports to the Manager and runs every step after the restore, and the restore reads the backup from the vault over SMB for its whole length. Isolation therefore means firewalling. The Manager and the storage machine can share a network with the targets, as long as the only links between them are the paths described here.

What you attest

When you attest isolation, you confirm that, for every restore target in the plan:

- No domain controller, member, or client of the directory being recovered, and no other production machine, can reach it.
- It has no route out.
- Its only links are two firewalled paths: to the Manager on the agent port, and to the storage machine over SMB.
- Those paths are enforced outside the target—at the switch, the virtual switch, a router, or the host's port ACL—not by the target's own firewall.

The target's own firewall does not count because the target is restored from a backup that is not trusted until you have checked the restored data. Its firewall rules come back with that backup.

In an unattended plan, the person who executes the plan attests isolation in the execute dialog, as the plan starts, instead of when the plan reaches the step.

Required paths

From	To	Port	Used by
Each restore target	Manager	TCP 5211 (the agent port, by default)	The recovery agent in the recovery environment while it restores, and the restored domain controller's DC agent from its first start, which carries out every later step.
Each restore target	Storage machine	TCP 445 (SMB)	The restore reading the backup from the vault, and the backup checkpoint writing a fresh backup.

For every port the product uses, see Ports and Protocols.

Paths you may also need

Depending on the plan, allow these as well, each limited to the machines named:

- **Time source**—in a forest recovery, the restored root PDC emulator is pointed at the external time sources you name in the plan, over NTP (UDP 123). Allow it to reach those servers and nothing else. Without this path the time step cannot confirm the external source; the check shows as not confirmed and becomes a follow-up in the recovery report.

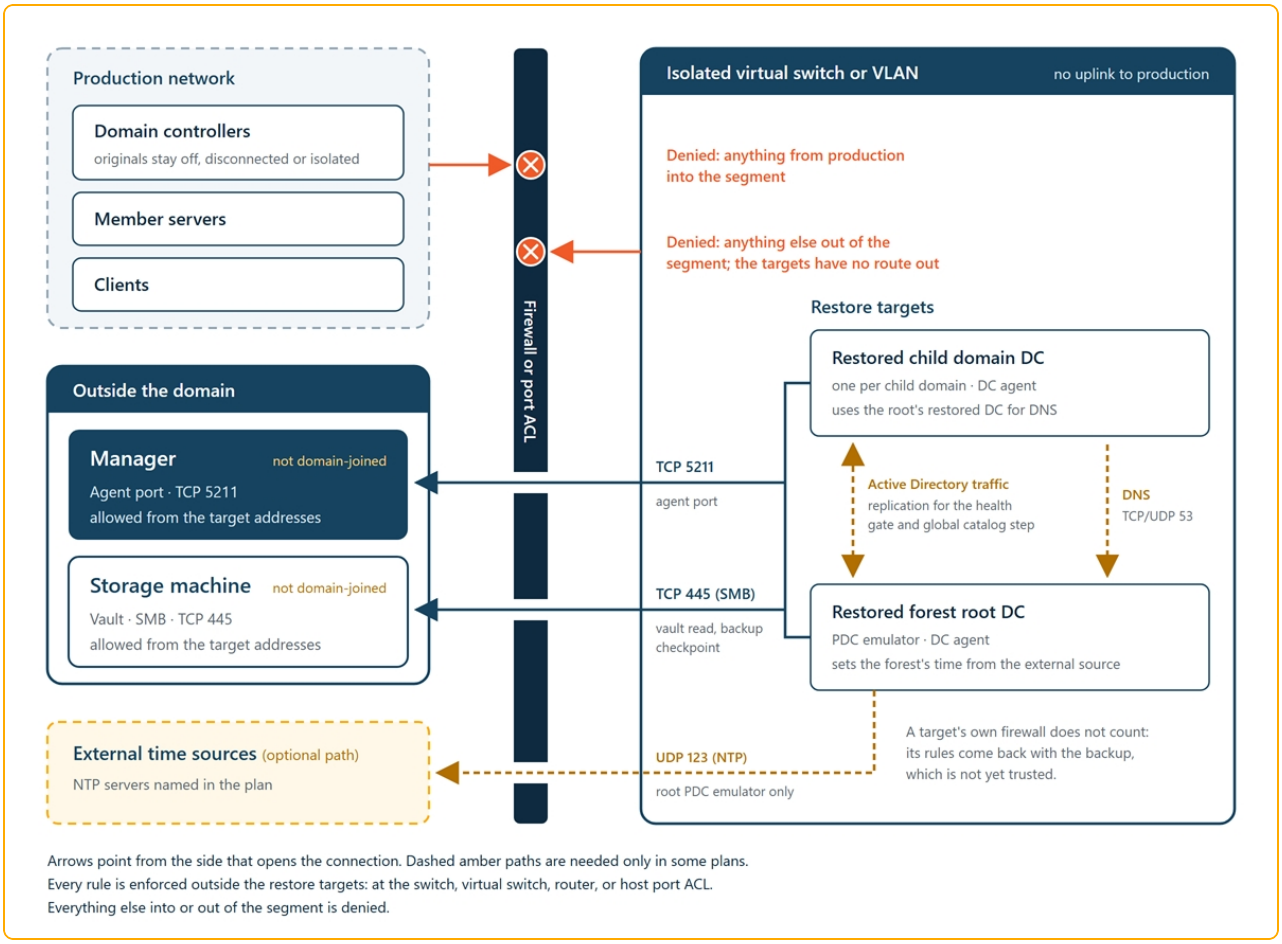
- **Between restored domain controllers**—in a forest with more than one domain, the restored domain controllers of each domain are joined to one isolated network, must reach each other, and must resolve each other's names. Allow Active Directory traffic among them, and only among them. From the DNS repair on, a child domain's restored domain controller needs DNS (TCP and UDP 53) to the forest root's restored domain controller, which the repair makes its preferred DNS server. The health gate and the global catalog step then pull replication between the restored domain controllers.
- **Rebuilt domain controllers**—a rebuilt domain controller is promoted from a restored one after the reconnect. Its machine needs the same agent port to the Manager, and must reach the restored domain controller of its domain, which holds the RID master role: the DSRM password step waits up to 15 minutes for a newly promoted domain controller's first RID pool from it.

Reference layouts

Any layout works if it enforces the paths above outside the targets. Three common ones:

- **Isolated Hyper-V virtual switch with port ACLs**—connect the restore targets to a private or internal virtual switch with no uplink to production. Give the Manager and the storage machine, or a firewall in front of them, an adapter on that switch, and use Hyper-V port ACLs on each target's adapter to allow only the agent port to the Manager and SMB to the storage machine. Virtual domain controllers can keep their original addresses on such a switch.
- **VLAN with router ACLs**—place the restore targets on a dedicated VLAN. On the router or layer 3 switch that joins it to the rest of the network, deny everything from the VLAN except the agent port to the Manager's address and SMB to the storage machine's address, and deny everything into the VLAN from production.
- **Physical targets**—cable physical domain controllers to dedicated switch ports or a dedicated switch, and apply the same rules with switch port ACLs or the upstream firewall.

Do not give the Manager a second network adapter with a general route between production and the targets. If the Manager must sit on both networks, firewall that adapter down to the agent port from the target addresses.



Rules on the Manager and storage machine

Limit the inbound rules on both machines to what the recovery needs:

- On the Manager, allow the agent port only from the addresses the restore targets use during the recovery, in addition to your normal agents.
- On the storage machine, allow SMB only from the same addresses, in addition to the domain controllers it serves normally.

The restore reads the vault with a short-lived, read-only credential issued for that restore. The vault path stays reachable from the restored domain controller's first start until the backup checkpoint, so keep the SMB rule limited to the restore targets' addresses.

Addressing

Restored domain controllers keep their original IP addresses unless the plan gives them other ones. If an original domain controller is still running anywhere the isolated segment can see, the two collide. Power off, disconnect, or isolate every original before the restore, and keep it that way for good: once the recovered directory goes live, an original that returns still holds the directory as it was before the recovery.

Name resolution

The recovery does not depend on production DNS:

- The vault is addressed by the storage agent's advertised address, which by default is the storage machine's IP address, so the recovery environment needs no DNS to reach it. On a storage machine with more than one network address, set the advertised host to the IP address the restore targets reach; otherwise the agent falls back to its machine name, and the restore needs DNS to resolve it.
- The Manager's address is resolved to an IPv4 address before the recovery environment starts, so the recovery environment needs no DNS to reach the Manager either.
- The restored domain controller's DC agent dials the Manager address it was installed with. If that is a name, the isolated segment must resolve it. Install DC agents with the Manager's IP address to remove this dependency. See Deploying DC Agents.

Restored domain controllers resolve each other through their own DNS once the plan repairs it. In a child domain, the repair makes the forest root's restored domain controller the preferred DNS server.

Before you attest

To check isolation before you attest it:

1. Confirm that every original domain controller of the directory being recovered is powered off, disconnected, or isolated, and will stay so.
2. Confirm that each restore target is connected only to the isolated segment, and that its adapter or switch port carries the rules described here.
3. From a machine on the isolated segment, confirm that you cannot reach a production domain controller, member server, or client, nor any address outside the allowed paths.
4. Confirm that the Manager's agent port and the storage machine's SMB answer from the isolated segment. On the plan's page, each target's agent shows as connected.
5. In a forest recovery, confirm that the restored root PDC emulator can reach the named time sources, if you allowed that path.
6. Attest the step on the plan's page, or with `cdr recovery attest`.

Related topics

- Recovery Requirements
- Recovering a Forest

- Recovering a Domain
- Ports and Protocols

What Recovery Does Not Cover

Cygna Forest Recovery recovers Active Directory Domain Services: the directory, SYSVOL, DNS as it runs on domain controllers, and the domain controllers themselves. Some services that commonly run on or depend on domain controllers are outside that scope. A recovery restores the machine they ran on, but does not make them work again. Plan for each of these before an incident, with whoever owns them.

What is not recovered

Item	Why it needs attention	What to do
Active Directory Certificate Services on a domain controller	A certification authority running on a domain controller is not recovered as a service. After a forest or domain recovery, its state may not match the directory, and after a rebuild the role is reinstalled empty, without its keys or database.	Recover the certification authority with its own procedure, from its own backup of the CA database and key. Better, move it off domain controllers before you need to recover.
DHCP Server on a domain controller	Scopes, leases, and authorization are not recovered as a service.	Restore the DHCP configuration from an export or backup made with the DHCP tools, and re-authorize the server in the directory if needed.
Directory synchronization services	Services that synchronize the directory with another identity system depend on accounts and passwords that a recovery resets, including the `krbtgt` and machine account resets and, with the breach flag, administrative and	After the recovery, re-establish each synchronization service's credentials and service accounts, then confirm synchronization with the vendor's procedure.

Item	Why it needs attention	What to do
	service accounts.	
External and outgoing trusts	Trusts to domains outside the forest are not restored from backup. Trusts inside the forest come back with the forest: in a forest recovery, each side keeps the trust password its own restore point holds, and the recovery resets them only when the security breach flag is set. A domain recovery always resets the domain's trusts.	Re-create each external and outgoing trust after the recovery, with the partner's administrator. A trust password reset has to be mirrored on a live domain controller of the partner domain; where no DC agent sits there, it is a manual step for the partner's administrator. After a forest recovery without the security breach flag, check that each trust inside the forest authenticates if its two sides were restored from points 30 days or more apart.

Anything else that reaches a domain controller by a specific address, such as DNS forwarders on other servers, firewall rules, DHCP options, and applications, needs updating if a recovered domain controller comes back on a different address.

How the console warns you

The console points these out before a recovery:

- **In preflight**—When a backup records other server roles on the domain controller, a restore of it shows the advisory finding **Colocated roles are not recovered by this product**, naming the roles. A domain or forest plan lists each external trust whose far side is a manual step, and each trust reset with no agent on the live side.
- **On the plan's page**—For each rebuilt domain controller, the **Plan as built** card says which roles and features are reinstalled, which roles come back without their configuration, and how many installed programs are not reinstalled.
- **On the Forests screen**—The **Trusts** card has an **At recovery** column. A trust inside the forest shows **Restored with the forest**. Every other trust shows that it must be re-created, and whether its password reset can be driven through a DC agent in the partner domain or is a manual reset.

Agree the manual steps with the owners of partner domains and colocated services before a recovery needs them.

The recovery report

The last step of every domain and forest plan produces the recovery report and emails it to the plan's notification addresses. Its manual follow-ups list the work the recovery leaves open, including:

- External and outgoing trusts to re-create.
- Colocated roles to recover: on a restored domain controller, services to check; on a rebuilt one, roles that were reinstalled without their configuration and data, programs that were not reinstalled, and insecure features that were left out.
- Directory synchronization services to resynchronize after the krbtgt resets.
- Final DNS client configuration and operations master placement, which the recovery leaves as it needed them rather than as your design has them.

For everything the report contains, see After a Recovery.

After any recovery

Take a fresh backup of every recovered and rebuilt domain controller after the recovery. Older restore points describe the directory as it was before the recovery and are not suitable to restore the recovered estate. Forest and domain plans take a backup checkpoint of each restored domain controller before redeployment; rebuilt domain controllers are backed up by their backup policies. For the full checklist, see After a Recovery.

Using the Console

The console is the web interface for Cygna Forest Recovery. You use it to see what is protected, schedule backups, review restore points, plan and run recoveries, and administer the Manager. The Manager serves the console itself, so it does not depend on Active Directory, DNS, or any other server: you can sign in and work while every domain controller is offline.

This chapter describes each console screen: what it is for, who can see it, what its columns, filters, and statuses mean, and which actions it offers. The procedures that use these screens are in the task and recovery chapters; each screen topic links to them.

Opening the console

Open a browser on any machine that can reach the Manager and go to `https://<manager>:5210/`, where `<manager>` is the Manager's IP address or a name your workstation can resolve. Because the Manager is not joined to the domain, its IP address is usually the most reliable choice.

If the HTTP redirect is enabled on the Manager (it listens on port 80 by default), `http://<manager>/` redirects you to the HTTPS address. The console refuses plain HTTP for everything else.

The Manager uses a certificate issued by its own certificate authority. Your browser warns about the certificate until you trust that authority on your workstation. For the steps, see Trusting the Manager Certificate. For the ports involved, see Ports and Protocols.

Signing in and out

You sign in with a Cygna Forest Recovery operator account: a username and password stored in the Manager's own database. Operator accounts are not Windows or domain accounts, which is why sign-in keeps working when the directory is down.

The first time anyone opens the console after installation, the sign-in page shows **Set up this manager** instead, which creates the break-glass account and sets the recovery passphrase. See Signing In for the First Time.

To sign in:

1. Open the console address.

   The sign-in page shows **Sign in to Cygna Disaster Recovery Suite**. On a wide enough window, the panel beside it shows the product name, **Cygna Disaster Recovery Suite**, with the module it carries, **Cygna Forest Recovery for Active Directory**.

2. Enter your **Username** and **Password**.
3. Select **Sign in**.

   The console opens the Dashboard, or the screen you were on if your previous session ended.

If you have turned on multi-factor authentication, the sign-in page asks for a code after the password. Enter the 6-digit code your authenticator app shows, or one of your recovery codes, in **Code**, and select **Verify and sign in**. See Multi-factor authentication.

If an administrator created your account or reset your password, the console shows **Choose a new password** after you sign in with the temporary password. Enter a new password in **New password** and **Repeat it**, then select **Set password and sign in**. The password must be at least 12 characters, must not contain your username, and must differ from the temporary one. If your account uses multi-factor authentication, the code is asked for next.

The sign-in page reports these problems:

Message	What it means
The username or password is not correct, or the account cannot sign in now. An account is locked for 15 minutes after 5 failed sign-ins; an administrator can unlock it.	The sign-in page gives this one answer for a wrong username or password, a locked account, a disabled account, and a removed one, so it tells nobody which accounts exist. Check both and try again. If it goes on, ask an administrator: the audit trail records the real reason for each refused sign-in. The break-glass account is never locked by wrong passwords.
Too many failed sign-ins from this address. Try again later.	10 sign-ins from your address failed within 15 minutes, for any accounts. Sign-ins from it are refused until the oldest of those failures is 15 minutes old. The break-glass account is refused too, except when it signs in on the Manager machine itself.
That code is not correct.	The authenticator code or recovery code was wrong or already used. Wait for the next code and try again. If a fresh code is refused too, check that the clocks on your phone and on the Manager are both right.
Too many wrong codes: authenticator codes are refused for now.	Authenticator codes are locked until the time shown. A recovery code still works, or an administrator can unlock the account. If you did

Message	What it means
	not enter the wrong codes, someone else has your password: change it.
This manager cannot read your authenticator enrolment: its database was moved from another machine.	The Manager was rebuilt from a database snapshot. Sign in with a recovery code and set MFA up again, or ask an administrator to reset your MFA.
The manager did not respond. Check that it is running and reachable.	The browser could not reach the Manager. Check that the `CygnaDRManager` service is running and that your workstation can reach port 5210.

To sign out, open the user menu at the right of the top bar and select **Sign out**. If the screen you are on holds choices that are not saved yet, the console asks you to confirm before it leaves the page.

Session timeouts

A console session ends after 30 minutes without activity, and in any case 12 hours after you signed in. The user menu shows when your current session ends (**Session ends**). A session also ends when an administrator disables your account or resets your password or multi-factor authentication, and your other sessions end when you change your own password or turn on multi-factor authentication.

When a session ends, the next request returns you to the sign-in page with the message **Your session ended. Sign in again to carry on where you were.** After you sign in, the console returns you to the screen you were on.

Changing your password

To change your password:

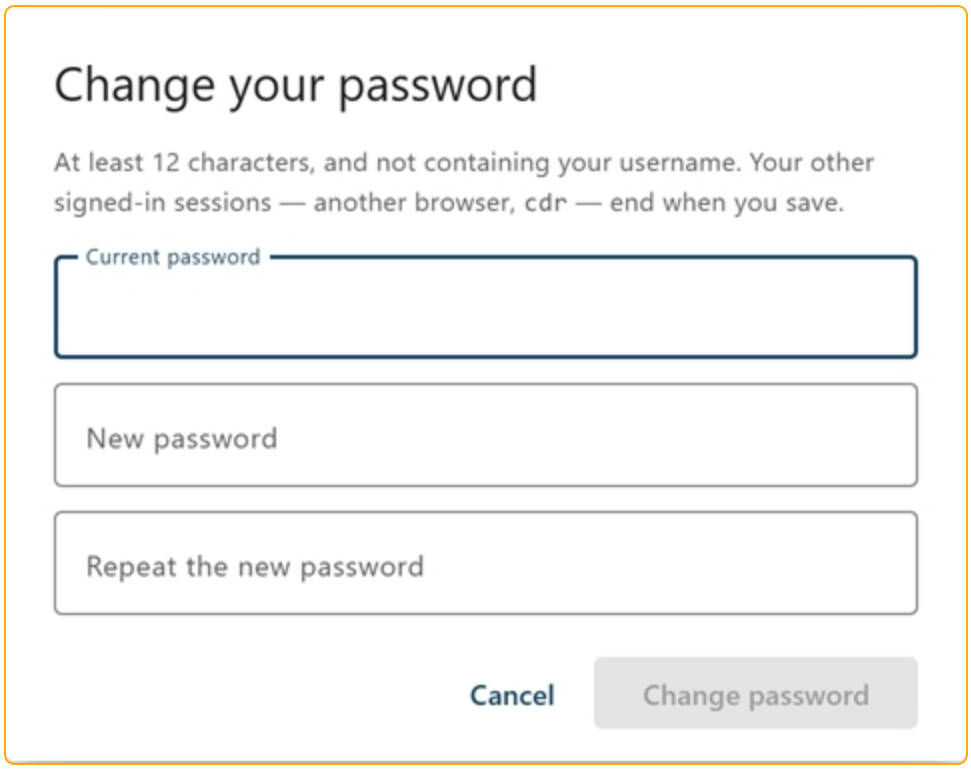
1. Open the user menu and select **Change password**.

   The **Change your password** dialog opens.
2. Enter your **Current password**.
3. Enter the new password in **New password** and again in **Repeat the new password**.

   The password must be at least 12 characters and must not contain your username.
4. Select **Change password**.

   The console confirms with **Password changed. Your other sessions were signed out.** Sessions in other browsers and in the `cdr` command-line tool end.

Note: **Important:** No administrator can reset the break-glass account's password. This dialog is the only way to change it. When you change it, update the offline record of the credential as well.



Multi-factor authentication

Multi-factor authentication (MFA) makes sign-in ask for a 6-digit code from an authenticator app on your phone after your password, in the console and in `cdr`. It is optional, and each operator turns it on for their own account. Open the user menu and select **Multi-factor authentication**; the menu item shows **On** or **Off**. In the dialog you can:

- **Set up**—Turn MFA on: enter your password, scan the QR code with your authenticator app, enter the code it shows, and store the ten recovery codes the dialog then shows once.
- **New recovery codes**—Replace all your recovery codes, used or not. The dialog shows how many unused codes you have.
- **Turn off**—Turn MFA off. To move MFA to a new phone, turn it off and set it up again.

Replacing the codes and turning MFA off each need your password and a code from the app, or a recovery code. For the full procedures, see Multi-Factor Authentication.

Changing the appearance

The console offers six color themes. Your choice is saved to your operator profile on the Manager, so it follows you to any browser or computer you sign in from, and it does not affect other operators.

- **Light**–Navy headings, light surfaces, and amber selection. This is the default.
- **Dark**–Charcoal surfaces with brighter text and status colors.
- **High contrast dark**–Black surfaces, white text, yellow links, and strong borders.
- **High contrast light**–White surfaces, black text, blue links, and strong borders.
- **Tactical**–Blue-black panels, glowing cyan readouts, and a faint grid.
- **Starship**–A space-computer look: black screens with bright colored panels.

To change the appearance:

1. Open the user menu and select **Appearance**.

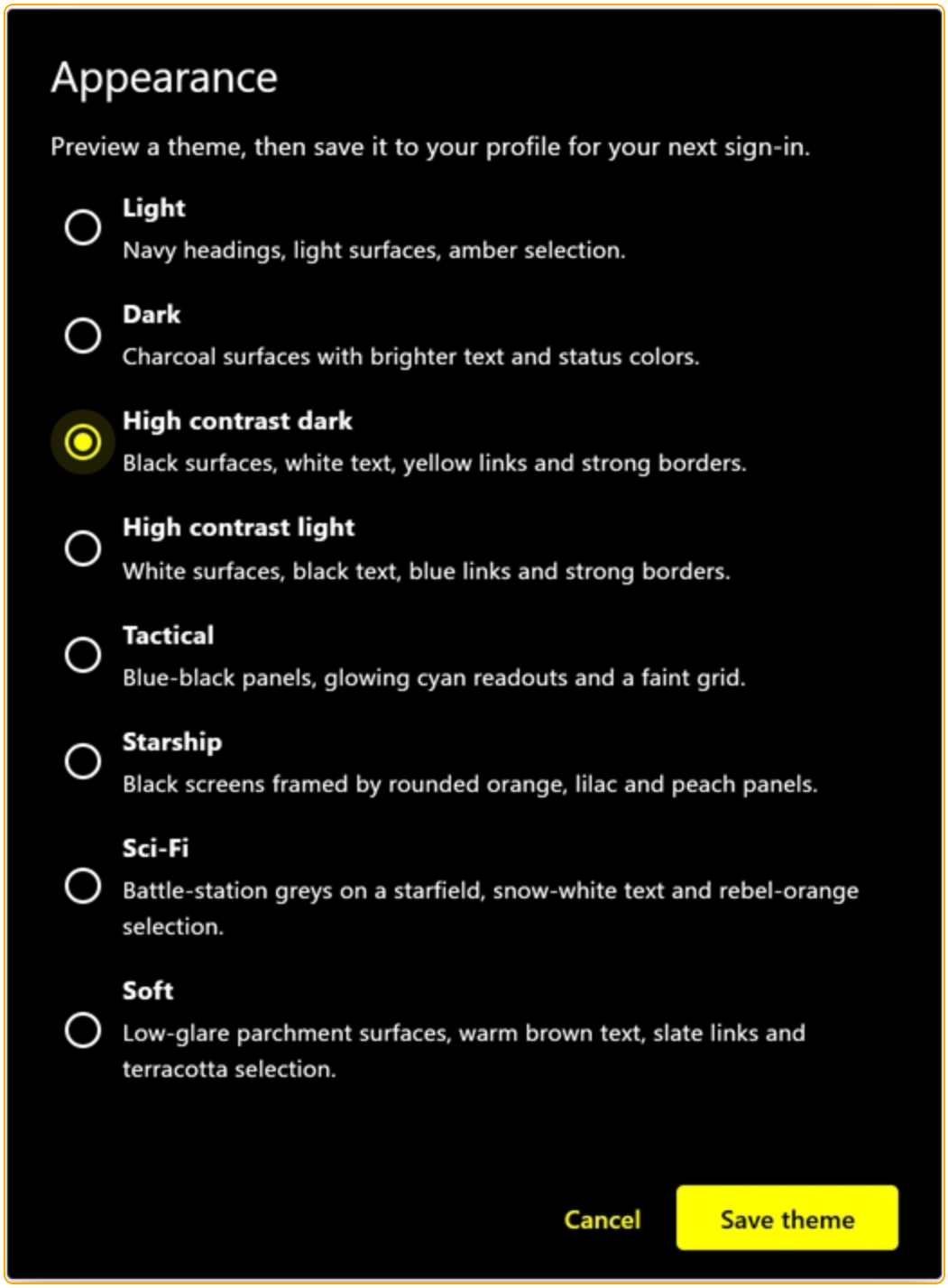
   The **Appearance** dialog opens.

2. Select a theme.

   The console shows the theme at once, so you can preview it.

3. Select **Save theme**.

   The console confirms with **Theme saved to your profile.** To keep your saved theme instead, select **Cancel** or press Esc.

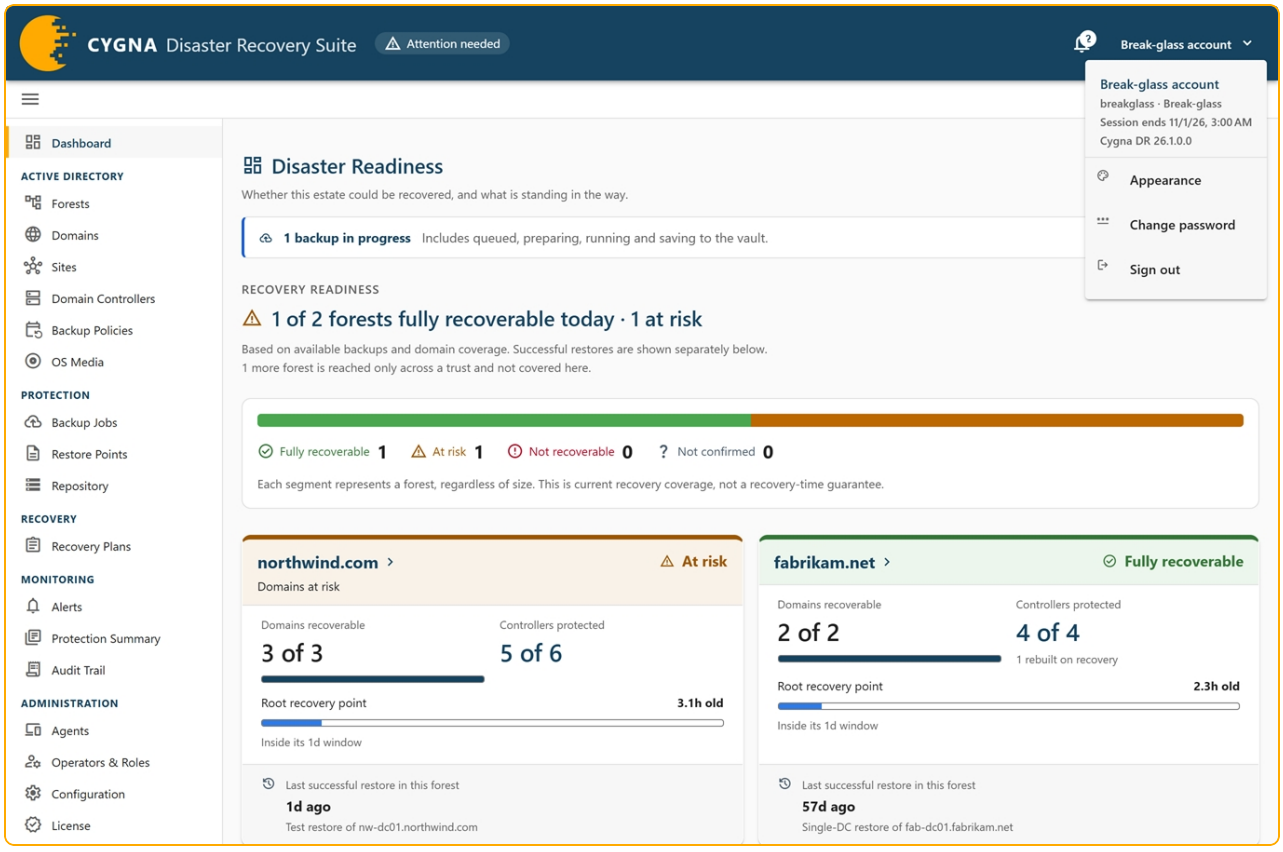


The theme is applied when you sign in. When you sign out, the console returns to the Light theme. In every theme, a status is conveyed by a label and an icon, never by color alone.

Console layout

Every screen shares the same frame:

- **Top bar**–The product mark on the left returns you to the Dashboard. On the right are the health chip, the alerts bell, and the user menu.
- **Navigation**–The left pane lists the screens you can open, in groups. Use the **Toggle navigation** button to collapse it to icons; the browser remembers your choice.
- **Breadcrumb**–Shows the group and screen, for example **Active Directory** › **Domains**. When a screen is scoped to a forest, domain, site, or machine, or a detail sheet is open, the breadcrumb adds that object. Each part is a link.
- **Recovery banner**–Appears below the breadcrumb while a recovery is running. See The recovery banner.
- **License banner**–Appears below the recovery banner when the license is ending soon, has ended, or is stopping backups and recovery operations. See The license banner.
- **Screen**–The screen's header shows its title, a one-line description, when the data was last updated, and a **Refresh now** button.



Top bar

The **health chip** summarizes the domain verdicts and the protection of the domain controllers. Select it to open the screen behind the verdict: the Domains screen when a domain set the color, otherwise the Domain Controllers screen, filtered to the machines behind the verdict when there is a problem.

Health chip	Meaning
System healthy	Every backup target is protected: it has a current, restorable backup and its last backup succeeded. No domain has a finding against it.
Attention needed	At least one domain is at risk, or at least one domain controller has a stale backup, a failed last backup, or no schedule, or is a backup target with nothing to restore from.
Action required	At least one domain could not be recovered. This is shown as soon as the domain verdicts are known, even before the domain controller list has been read.
Coverage incomplete	Fewer than 90 percent of the domain controllers the chip counts have a verdict, so the chip cannot speak for the rest. The tooltip shows how many were assessed, and how many domain controllers recovery rebuilds, which the chip does not count.
Nothing discovered	No domain controller that the chip counts is known yet, and no domain has a finding against it.
Not yet assessed	The console has not received the first answer from the Manager.

Domain controllers that are excluded from alerts, recorded as removed from the directory, or rebuilt on recovery do not count toward the health chip, and neither do excluded domains. See Backup Targets and Rebuilt Domain Controllers.

The **user menu** shows your display name, your username and roles (or **Break-glass**), when your session ends, and the product name and Manager version. It also contains **Appearance**, **Change password**, **Multi-factor authentication** (with **On** or **Off**), and **Sign out**.

Navigation groups

The navigation pane shows only the screens your roles allow you to open. A group with no screen you can open is not shown.

Group	Screen	What it shows
(none)	Dashboard	Whether each forest could be recovered today, and what to fix first. The page title is **Disaster Readiness**.
Active Directory	Forests	Forest readiness, and the trusts each forest holds.
Active Directory	Domains	Recovery coverage for each domain.
Active Directory	Sites	Site coverage, subnets, and site links.
Active Directory	Domain Controllers	Every known domain controller, its protection, agent, and reachability. Backups and agent installs start here.
Active Directory	Backup Policies	The backup schedule of each domain controller.
Active Directory	OS Media	Windows Server installation media, whether each operating system in use can use full system recovery, and the recovery images each storage agent keeps and retires.
Protection	Backup Jobs	Every backup attempted and its outcome.
Protection	Restore Points Screen	What each backup captured about its domain controller.
Protection	Repository	Vault capacity, retention, and integrity.
Recovery	Recovery Plans	Restore runs, including test restores, and forest and domain recovery plans.
Monitoring	Alerts	Active alerts and their history.
Monitoring	Protection Summary	Backup outcomes over a period and coverage now, with a CSV export.
Monitoring	Audit Trail	Every operator and system action.
Administration	Agents	Enrolled DC agents, storage agents, and Isolated Recovery Host agents, and their versions.

Group	Screen	What it shows
Administration	Operators and Roles Screen	Operator accounts and role assignments.
Administration	Configuration	Manager and agent settings.
Administration	**License**	The installed license, where it stands, and the enabled user accounts beside its count. See Licensing.

Roles and what you can see

Each operator holds one or more roles. Every role can view the monitoring and inventory screens; the roles differ in what they can change. Duties are deliberately separated: administering the Manager does not carry backup or recovery rights, and recovery rights do not carry backup policy control. The break-glass account holds every capability.

Role	Screens and actions
Viewer / Auditor	Every screen except **Operators & Roles** and **Configuration**. Changes nothing.
Backup Operator	As Viewer / Auditor, plus backup policies and exclusions, and starting and stopping backups.
Restore Operator	As Viewer / Auditor, plus drafting, approving, and running restores, test restores, and recovery plans.
System Admin	As Viewer / Auditor, plus **Operators & Roles**, **Configuration**, agent installs and updates, and setting the recovery passphrase. No backup or restore rights.

Within a screen, most actions you are not allowed to take are hidden. A few are shown, and the Manager refuses them if you lack the role. For example, a Viewer / Auditor sees the Domain Controllers screen without the row selection checkboxes or the actions menu. The Manager checks every request as well, so a change you are not allowed to make is refused even if you reach it another way.

If you open a screen your roles do not allow, for example from a bookmark, the console shows **Not permitted**. The page names the capability the screen needs, the roles that hold it, and the roles you hold. Select **Go back** or **Go to the dashboard**. To have a role added, ask a System Admin. For the full role definitions, see Managing Operators and Roles.

Notifications

The bell in the top bar shows a badge with the number of alert types currently firing at critical or warning severity. Select the bell to open the notifications panel:

- **Recoveries in progress**–Each running restore or recovery plan, with its status. A plan that is waiting for an operator shows **Waiting for you**, and a plan stopped by a failed step shows **Halted – a step failed**. Select an item to open it.
- **Active alerts**–Up to six alerts, worst first, each with its severity and count, and a link to the screen where you can act on it. **No alert is firing at critical or warning.** means there is nothing to act on.
- **Open Alerts**–Opens the Alerts screen.

The bell's tooltip also says when a recovery is waiting for you.

The recovery banner

While any restore run or recovery plan is in progress, a banner appears below the breadcrumb on every screen except that recovery's own page and the Dashboard. The Dashboard shows running recoveries in its own **Recovery in progress** panel. The banner reads **Recovery in progress**, or **A recovery is waiting for you** when a step needs an operator to acknowledge, attest, or act. The banner lists up to three recoveries, such as **Restore of** a machine or **Recovery of** a forest, with their status; select one to open it, or select **and** *n* **more** to open the Recovery Plans screen. The banner refreshes every 15 seconds and disappears when nothing is in progress.

When the only runs in progress are test restores, the banner says so on a neutral band: **Test restore in progress – nothing in production changes**. Each test restore is listed with who started it, for example **Scheduled** or **Approved by** an operator. A test restore never changes production; see Test Restores.

For what each status means, see Recovery Plans.

The license banner

When the license needs your attention, a banner appears below the recovery banner on every screen except the **License** screen. It starts with **License ending soon** in the last 30 days before the end date, **License ended** during the grace period, or **Backups and recovery operations are stopped** when the license has run out or no usable key is installed. The license's own sentence follows, with its dates, and **View license** opens the **License** screen. The banner uses the critical color during the grace period and when operations are stopped. It refreshes every 5 minutes, and disappears once the license is current and ends more than 30 days from now. See Licensing.

Working with screens and tables

Most screens follow the same conventions.

Refreshing

Screens refresh themselves while open, and the header shows when the data was last updated, for example **Updated 20s ago**. Select **Refresh now** to reload at once. If a refresh fails, the screen keeps the data it has, shows **Refresh failed** with how old that data is, and offers **Try again**. If a screen cannot load at all, it shows the reason and a **Retry** button.

Load error	What to check
The manager did not respond.	The Manager service is not running, or your workstation cannot reach it.
The manager could not be reached.	Something between the browser and the Manager could not connect. Check the service and the address you use.
The manager refused this request.	Your roles do not include the capability the screen or action needs, or your session has ended.
The manager does not support this request.	The console and Manager versions may not match. Reload the browser page.

Summary cards

Inventory screens open with a summary card, such as **Recovery readiness** on the Forests screen or **Controller protection** on the Domain Controllers screen. Its headline states the screen's most important finding, and a line below it says what the verdict is based on and how many objects are in scope or excluded. Below that, the card lists the problems on that screen, worst first, each marked by severity—for example **Action required** or **Blocked** for a critical problem, and **Review** for a warning. Every problem is listed; there is nothing to expand. Each problem names the affected objects and says what clears it.

Status chips

Verdicts and statuses appear as chips that combine an icon with a color: a check mark for good, a warning sign for something that needs attention, an error sign for a critical problem, an information sign for a neutral fact, and a question mark when the state is not known. A state that is not known is never shown as good. Hover over a chip to see the reasoning behind it.

Search, filters, and scope

- **Search**—The box above a table narrows the rows to those that contain the text you enter. The count beside it shows how many rows match, for example **12 of 240 domain controllers**. Select **Clear search** to remove the text.
- **Filter chips**—The chips above a table count the rows in each state. Select a chip to show only those rows, and select it again to show every row. A filter changes only the table: the counts and alerts above it still describe everything on the screen. A notice marked **Filtered** says what is shown, with a **Clear filter** button.
- **Scope**—When you drill down from another screen, for example from a forest to its domains, the screen is scoped. A notice marked **Scoped** names the forest, domain, or site, and every count and alert on the screen then describes that scope alone. Select **Clear filter** to see everything again.

Sorting, paging, and detail sheets

- Select a column heading to sort by that column, and select it again to reverse the order.
- Long tables are paged. Choose the number of rows per page at the bottom of the table.
- Select an object's name to open its detail sheet at the side of the screen. Select **Copy name** to copy the name, and close the sheet with the close button or Esc.
- The browser address keeps the search, sort, filters, page, and open detail sheet. They survive Back and a browser refresh, and you can bookmark the address or send it to a colleague to show the same view.

Actions and confirmations

Row actions are on the actions menu (⋮) at the end of each row. On screens that support bulk actions, select rows with the checkboxes and a bulk bar appears with **Select all** *n* **matching**, the available actions, and **Clear selection**.

Actions that change something open a confirmation dialog that states the consequences. Some ask for a reason, which is recorded in the audit trail. Actions that cannot be undone ask you to type a phrase, such as the machine name, before the confirm button works. If the Manager refuses an action, the dialog shows **Refused** and the Manager's reason.

Pages where you build a plan keep your choices only in the browser until the plan is built. If you leave such a page, the console asks you to confirm with **Leave and discard**.

Longer explanations are folded under a **Details** link; select it to expand them.

Exclusions

On the Forests, Domains, Sites, and Domain Controllers screens, a Backup Operator can exclude an object from alerts: for example, a test domain controller you do not intend to protect, or an orphaned domain partition. An excluded object is still listed with its real verdict, but it is left out of the counts, alerts, and readiness figures. Excluding a forest, domain, or site also excludes the objects in it.

To create or change an exclusion, select **Exclude from alarms…** or **Edit exclusion…** on the row's actions menu. The **Exclusion** dialog asks:

- **Why is <object> out of scope?**–Required. The reason is recorded against your name and shown wherever the object appears.
- **Review by (optional)**–A date after which the object counts again and is flagged as due for review. Leave it empty to exclude the object until someone clears the exclusion.

Select **Exclude from alarms** or **Update exclusion** to save, or **Clear exclusion** to remove the object's own exclusion. If the object is excluded through its domain, site, or forest, the dialog says so; clear the exclusion there instead.

Exclusion chip	Meaning
Excluded / Excluded from alarms	The exclusion is active. Hover to see the reason, who set it, and the review date.
Excluded via its domain / its site / its forest	The object is excluded because a containing object is.
Exclusion due for review	The review date has passed. The object counts again until someone renews or clears the exclusion.
Exclusion void – the object changed	The object changed after the exclusion was recorded, so the exclusion no longer applies.

For the procedure and guidance on when to exclude, see Excluding and Retiring Objects.

In this chapter

- Dashboard

Active Directory

- Forests
- Domains
- Sites
- Domain Controllers

- Backup Policies
- OS Media

Protection

- Backup Jobs
- Restore Points Screen
- Repository

Recovery

- Recovery Plans

Monitoring

- Alerts
- Protection Summary
- Audit Trail

Administration

- Agents
- Operators and Roles Screen
- Configuration

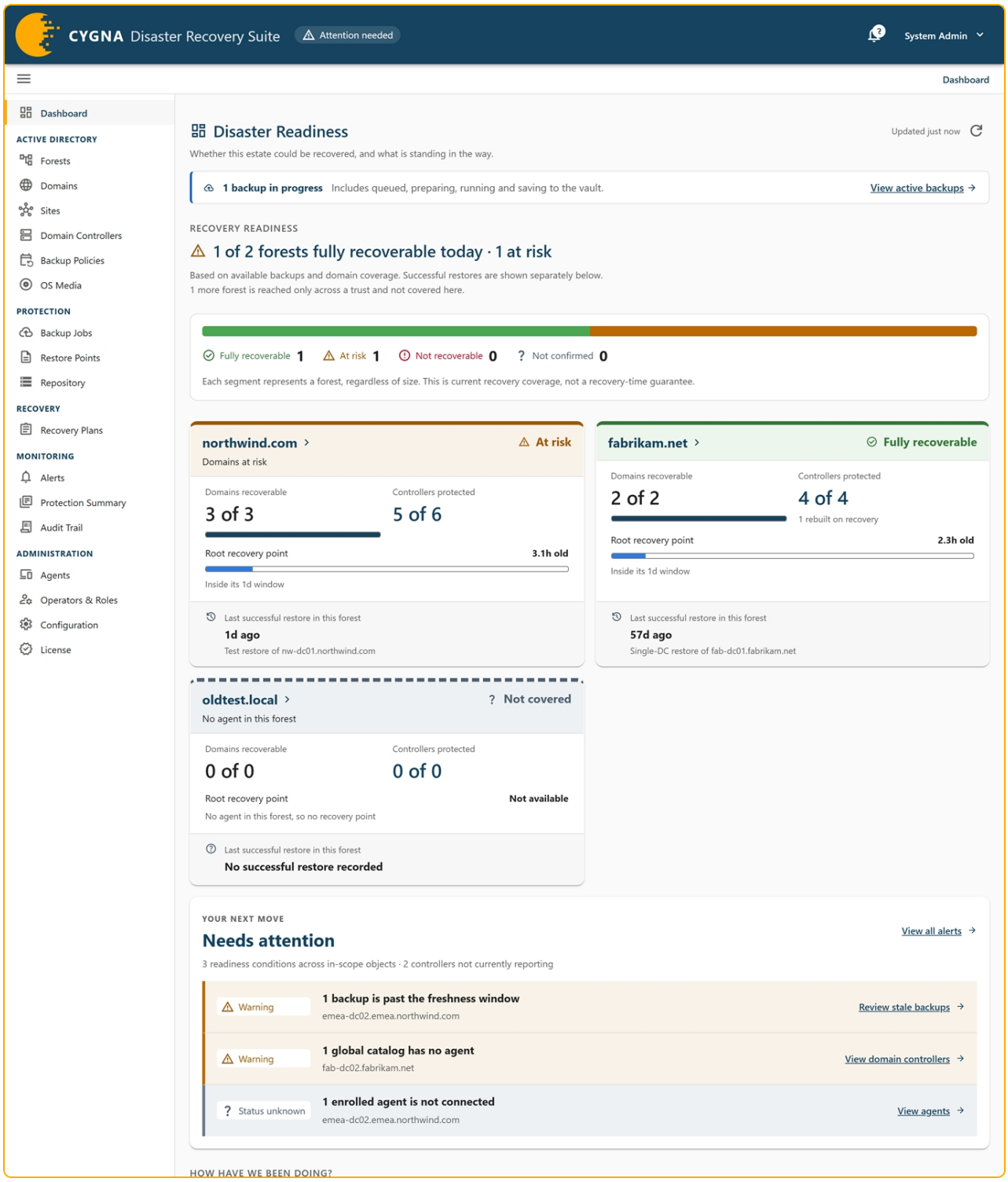
Dashboard

The Dashboard answers two questions: could this estate be recovered today, and what is standing in the way. It is the first screen after you sign in, and its page title is **Disaster Readiness**. To open it from anywhere, select **Dashboard** in the navigation pane or the product mark in the top bar.

Every role can see the Dashboard. It is read-only: each figure links to the screen where you can act on it.

All counts leave out objects an operator has excluded from alerts and domain controllers recorded as removed from the directory. The inventory figures refresh every 30 seconds; the backup and recovery history refreshes every five minutes. If a refresh fails, the Dashboard keeps showing the last figures it received and says how old they are.

Until a DC agent has enrolled, the Dashboard shows **Nothing has been discovered yet.** Install the DC agent on one domain controller; it reads the forest's configuration and the Dashboard starts reporting. See Deploying DC Agents.



Backups in progress

For an operator who can view backups, a line at the top of the Dashboard counts the backups in progress, including those queued, preparing, running, and saving to the vault, for example **3 backups in progress**. With none, it reads **No backups in progress**. **View**

active backups opens the Backup Jobs screen filtered to them. The line refreshes every ten seconds. If it cannot be refreshed, it reads **Backup activity could not be refreshed**, with the last count it received.

Recovery in progress

When a restore or recovery is running, and you can view recovery plans, a **Recovery in progress** panel at the top of the Dashboard lists each one with its status, its current step, how many of its steps are complete, and when it started. Select **View recovery** to open its page, or **Review required action** when it is waiting for a person. **All recoveries** opens the Recovery Plans screen. If the Dashboard cannot check for active recoveries, it shows **Recovery activity could not be refreshed** with a **Check recoveries** link. The recovery banner that appears on other screens is not shown on the Dashboard.

A test restore is marked as one, so nobody signing in mid-test takes it for a real recovery. It is shown compactly: the domain controller, its status, whether it is **Scheduled**, **Approved by** an operator, or **Manual**, and **View test restore**. Only a test restore that is waiting for a person gets the full entry. When every running item is a test restore, the panel is smaller, headed **Test restore in progress** or **Test restores in progress** with how many are active, and lists each by its domain controller's name. At most two routine test restores are listed; beyond that, a line says how many more are in progress, with **View all test restores**. On other screens, the recovery banner then reads **Test restore in progress — nothing in production changes** on a neutral band instead of the information color a recovery gets, and names whether each was scheduled or who approved it. For test restores, see Test Restores.

Recovery readiness

This section opens with a one-line verdict for the estate, for example **All 2 forests could be fully recovered today**, or a count of forests that are fully recoverable today and of those that are not. Forests known only through a trust, with no agent in them, are not counted; the line below the verdict says how many there are.

A bar below the verdict has one segment for each forest, regardless of its size, with counts for **Fully recoverable**, **At risk**, **Not recoverable**, and **Not confirmed**. It shows current recovery coverage, not a recovery-time guarantee.

Below the bar is one card for each forest:

- **Verdict**—**Fully recoverable**, **At risk**, **Not recoverable**, **Not confirmed**, or **Not covered**. When the forest's readiness on the Forests screen says more, for example **Domains at risk**, it appears beneath the forest name.
- **Domains recoverable**—How many of the forest's domains could be recovered.
- **Controllers protected**—How many backup targets are protected, out of those in scope. Domain controllers rebuilt on recovery are not backup targets, so they are left out of

the figure and named below it. A line below says what the others are, when it is not a missing backup, for example `1 need a backup policy · 1 last backup failed (still recoverable) · 2 backup too old · 4 rebuilt on recovery · 1 excluded`. Domain controllers that need a backup policy are ones a domain with too few backup targets needs backed up; they stay in the figure. See Backup Targets and Rebuilt Domain Controllers. Select the figure to open the Domain Controllers screen scoped to the forest.

- **Root recovery point**–The age of the forest root domain's newest recovery point against its freshness window. A forest recovery starts from the root domain, so this is the age of the point the whole forest would return to.
- **Last successful restore in this forest**–When a restore or recovery in this forest last completed, and what it was, for example **Test restore of dc01.corp.example.com** or **Forest recovery**. A test restore that passed counts as a completed restore here. When something newer than the forest's last forest recovery is named, a second line gives that forest recovery's age, for example **Last forest recovery 3 days ago**. **No successful restore recorded** means no recovery has proven these backups yet.

Select the forest name to open the Domains screen scoped to that forest. If forests are excluded, a line says how many, with a link to them on the Forests screen.

For how forest and domain verdicts are decided, see Coverage and Readiness.

Needs attention

The **Needs attention** section, under **Your next move**, lists what stands in the way of recovery, worst first. Each item names the affected objects and links to the screen that holds the details. Select **View all alerts** to open the Alerts screen. When nothing is wrong, the section shows **No active readiness alarms**.

The section lists the alerts in the table below. The test restore alerts, **Domain controllers whose latest test restore failed** and **Domain controllers with no recent verified backup**, are not listed here; they appear on the Alerts screen and in the notifications panel. Neither are the security and recovery plan alerts, such as **Unexpected SMB connections to the staging share**, **Committed vault content changed outside retention**, or **krbtgt second reset due**; the bell's badge in the top bar counts them with the rest.

Alert	Severity	What to do
Forests could not be recovered	Critical	The forest root domain cannot be recovered, so recovery cannot start, or a child domain cannot, so it would stop partway. The Forests screen says which.
Domains could not be recovered	Critical	The domain has nothing restorable, its backups are past the tombstone lifetime,

Alert	Severity	What to do
		or it replicates SYSVOL by FRS. The Domains screen names the reason.
Backup targets have nothing to restore from	Warning	A policy covers each and nothing of it is in the vault: the first backup has not completed, every one failed, or every one was pruned. Check that the agent is enrolled, then run a backup. Whether this costs a domain its recovery is the domain alert above. **Review unprotected controllers** opens the Domain Controllers screen, filtered to the unprotected machines.
Domain controllers whose last backup failed	Warning	Check the failure and whether an older usable restore point exists. The Backup Jobs screen holds the failure and its transcript. The alert counts every domain controller whose newest backup attempt is treated as failed, including interrupted and uncommitted attempts. A backup refused because the domain controller had no agent, made before its current agent enrolled, is not counted.
Backups past the freshness window	Warning	Check that the agent is connected, then run a backup. **Review stale backups** opens the Domain Controllers screen, filtered to the stale machines.
Domain controllers with no backup schedule	Warning	A restore point exists and nothing keeps it current. Give each an enabled policy on the Backup Policies screen.
Forests have a stale topology	Warning	No agent has reported a recent enumeration of the forest, so the figures describe the estate as it was. Confirm a DC agent is running and reporting in each forest.
Global catalogs have no agent	Warning	Recovering any other domain in the forest reconciles every global catalog through its

Alert	Severity	What to do
		agent, and that recovery does not start until each has one. Install the DC agent, or remove the machine's global catalog role. Other domain controllers without an agent are not listed here.
Enrolled agents are not connected	Status unknown	Their figures are the last thing each reported. Confirm the agent service is running on each machine.

The freshness window is derived from your backup policies: it is twice the shortest enabled backup interval. It is not a separate setting.

Domain controllers that you do not back up, and that are rebuilt on recovery, raise no item here. Critical items belong to forests and domains that could not be recovered. For how the console decides which domain controllers need a backup, see Backup Targets and Rebuilt Domain Controllers.

How have we been doing?

Two cards show history.

Backup success covers every backup that finished in the last 30 days, compared with the 30 days before:

- The success rate, as a percentage. It shows good at 100 percent, a warning from 90 percent, and critical below 90 percent. The rate shows 100 percent only when every backup succeeded.
- The change in points against the previous 30 days.
- A bar chart of succeeded, failed, and cancelled backups per day. Days with a failed backup are marked ▼ .

Select **Open the protection summary** on the card's menu to open the Protection Summary screen.

Recent recoveries covers restores and domain and forest recoveries that actually ran in the last 90 days and have finished; one still running is shown in the Recovery in progress panel instead. It shows when the last successful restore completed and the four most recent recoveries with their status; each links to its page. A domain controller restore that a recovery plan carried out is part of that plan and is not listed on its own. Test restores are not among the four: the newest finished one is shown on its own line, **Latest finished test restore**. Select **View recoveries** on the card's menu to open Recovery Plans.

A note at the bottom of the Dashboard says that the krbtgt second-reset countdown is tracked on each recovery's page, on the Recovery Plans screen. A second reset still due and any recovery protection override are also raised on the Alerts screen. The history cards cover backup outcomes only.

Related topics

- Backup Targets and Rebuilt Domain Controllers
- Coverage and Readiness
- Forests
- Domains
- Alerts
- Using the Console

Backup Targets and Rebuilt Domain Controllers

Cygna Forest Recovery judges your backups against what a recovery needs, not against every domain controller. This topic explains how it decides which domain controllers need a backup, and how that shows on the Dashboard, the inventory screens, the command line, and the Protection Summary.

What a recovery needs

A domain recovery restores one writable domain controller from backup and rebuilds the other domain controllers of the domain by promotion. A forest recovery does the same in every domain. So you back up at least two writable domain controllers per domain, not every domain controller. The second backup is the one you fall back on if the first is unusable.

For which domain controllers to choose, see Choosing Which Domain Controllers to Back Up.

Backup targets

A domain controller with an enabled backup policy is a backup target. Every protection finding about a missing backup is about a backup target. A domain controller with no enabled policy and nothing in the vault is not a backup target: a recovery rebuilds it, so its verdict is **Rebuilt on recovery**. That is information, not a finding, and it raises no alert.

A disabled policy counts as no policy. A domain controller whose only policy is disabled is rebuilt on recovery when nothing of it is in the vault. When it still has a restorable backup, it reads **No schedule** instead, a warning, because that restore point only grows older.

Protection	Severity	When
Protected	Good	A restorable backup exists within twice the policy interval, and the last attempt succeeded.
Last backup failed	Warning	A fresh restorable backup exists, but the most recent attempt failed.
Backup stale	Warning	The newest restorable backup is older than twice the policy interval.
No schedule	Warning	A restorable backup exists, and no enabled policy will replace it.
Unprotected	Warning	A backup target with nothing restorable: the first backup has not completed, every backup failed, or every version was pruned. Raises **Backup targets with nothing to restore from**.
Rebuilt on recovery	Information	No enabled policy and nothing in the vault. A recovery rebuilds it by promotion. Raises no alert.

No domain controller verdict is critical. Whether losing a backup target costs its domain its recovery is a question for the domain, below.

Whether the domain can be recovered

The domain's coverage verdict on the **Domains** screen decides whether a domain can be recovered, whatever its rebuilt domain controllers read. A domain with two protected writable domain controllers and 198 that are rebuilt on recovery is **Covered**. If its backup targets lapse so that no writable domain controller holds a restorable backup, the domain turns **Unrecoverable**, which is critical and raises **Domains that could not be recovered**. Forests roll up their domains in the same way.

For every domain and forest verdict, see Coverage and Readiness.

Domain controllers that need an agent without a backup

A recovery of one domain also reconciles every global catalog outside that domain, through each global catalog's DC agent. The plan does not start while one of them has no agent. So a global catalog needs the DC agent, even if you do not back it up, when its forest holds another domain this installation would recover: a domain that still has a domain controller in the directory and is not excluded.

The Manager flags those domain controllers on each row of the domain controller inventory, and the console and the alert read the same flag:

- The alert **Global catalogs with no agent** names only these global catalogs.
- On the **Domain Controllers** screen, **No agent** is a warning on these global catalogs and on backup targets. On any other domain controller it is information, because a recovery rebuilds it and needs nothing of it. Hover over **No agent** to see which case applies.

A domain controller whose global catalog flag has not been read yet is not treated as a global catalog.

Sites

A site's verdict is a roll-up of its backup targets. Domain controllers rebuilt on recovery are counted and shown, but they are not findings, and no site verdict is critical: a location comes back by promotion, and the domain verdict answers whether each domain could be recovered.

- **Rebuilt on recovery**—The site has domain controllers and none of them is a backup target. Information.
- **Nothing to restore**—The site has backup targets, and no domain controller in it holds a restorable backup. Warning.
- **Partly protected**—Some backup targets in the site have nothing to restore from. Warning.
- **Protected**—Every backup target in the site is protected.

For every site verdict, see Sites.

The health chip

The health chip in the top bar rolls up the domain verdicts as well as the domain controllers' verdicts. A domain that could not be recovered turns it to **Action required**, and selecting the chip then opens the **Domains** screen. Domain controllers rebuilt on recovery do not count toward the chip. See Using the Console.

How the counts read

Protected domain controllers are counted against the backup targets. Domain controllers rebuilt on recovery are named beside them, not counted as a shortfall, with one exception. A recovery needs restorable backups of two writable domain controllers in each domain, or of every writable one in a domain that has fewer. A domain with fewer backup targets than that needs the difference, and that many of its rebuilt domain controllers stay in the figure as **need a backup policy**. A forest of two domains with three writable domain controllers each and no backup policy yet reads `0 of 4 · 4 need a backup policy · 2 rebuilt on recovery`, not `0 of 0`.

Which domain controllers take the policy is your choice, so no domain controller's own verdict changes. Read-only domain controllers neither count toward the need nor fill it. A backup target that you excluded still fills its domain's need, and an excluded domain needs nothing.

The counts read like this:

- **Dashboard**—Each forest card's **Controllers protected** reads, for example, `2 of 2` with `198 rebuilt on recovery` below it. See Dashboard.
- **Inventory screens**—The **Controllers** column on the **Forests**, **Domains**, and **Sites** screens reads, for example, `2 protected · 198 rebuilt on recovery`. The **Domain Controllers** screen has a **rebuilt on recovery** filter chip, and a rebuilt domain controller's **Restorable backup** and **Schedule** cells are plain text rather than warnings. See Domain Controllers.
- **Command line**—`cdr domain list`, `cdr forest list`, and `cdr site list` show **PROTECTED** as, for example, `2/2 (198 rebuilt on recovery)`. A site with no backup target reads `no backup targets`. A domain with too few backup targets reads, for example, `0/2 (2 need a backup policy, 1 rebuilt on recovery)`. `cdr dc list --protection rebuild` lists the domain controllers that are rebuilt on recovery. See Inventory Commands.
- **Protection Summary**—The domain table shows the numbers that need a backup policy and are rebuilt on recovery under **Protected**, the forest table names them beside **DCs protected**, and the CSV export has a column for each. See Protection Summary.

Exclusions

Exclusions are for objects that are out of scope, such as a forest reached across a trust that this installation is not meant to protect. You do not need to exclude a domain controller that recovery rebuilds: it already raises no alert, and it stays in the counts as rebuilt on recovery. An excluded domain controller leaves the counts altogether. See Excluding and Retiring Objects.

When a domain controller stops being a backup target, or a recovery no longer needs an agent on it, alert email drops it from the alert without reporting it as cleared. See Alerts Reference.

Related topics

- Dashboard
- Domain Controllers
- Sites
- Domains
- Coverage and Readiness

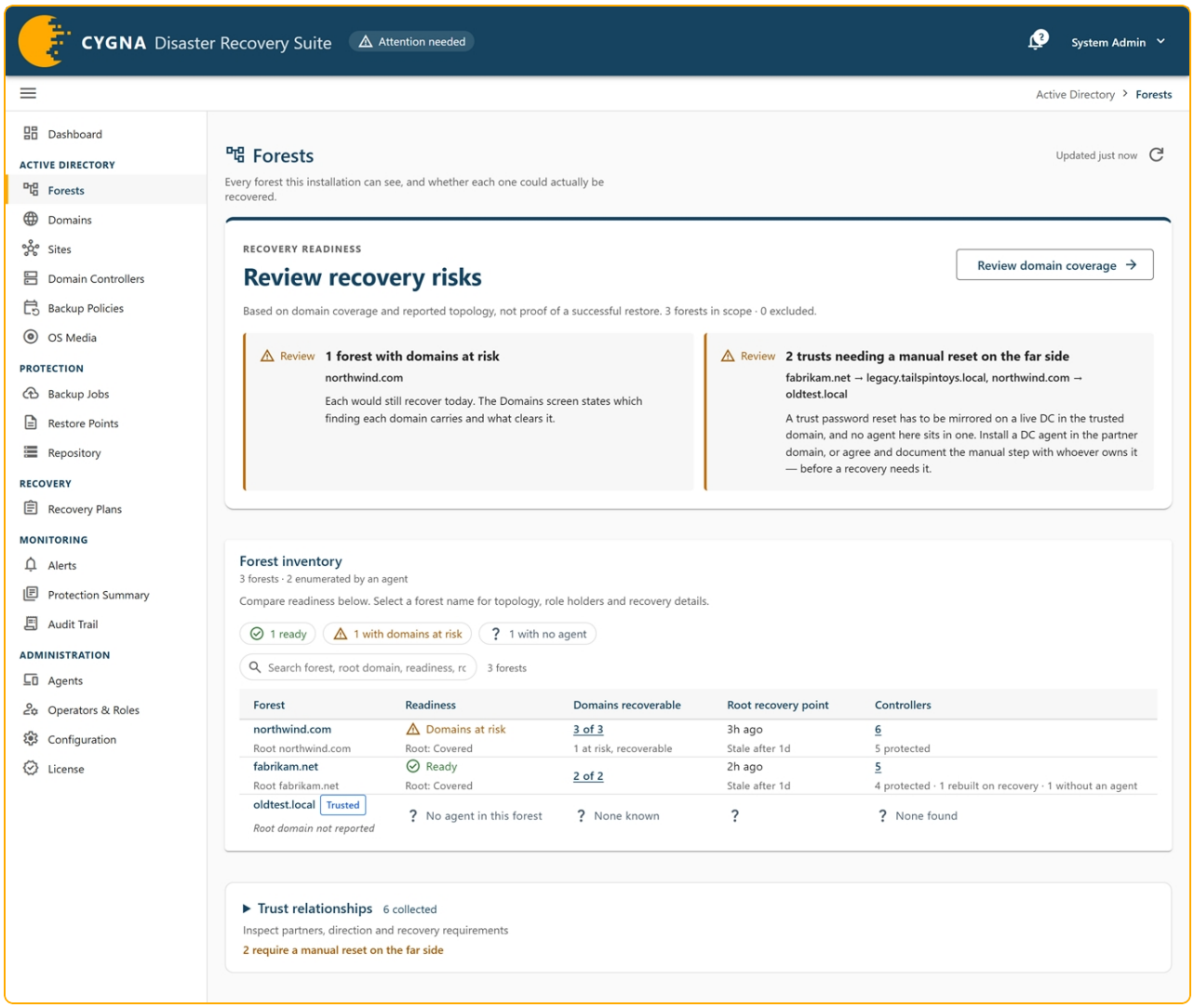
- Choosing Which Domain Controllers to Back Up
- Alerts Reference

Forests

The Forests screen lists every forest this installation can see and says whether each one could be recovered. A forest recovery restores the forest root domain first and every other domain after it, so a forest's readiness depends on its root domain above all. The screen also lists the trusts each forest holds and what each one needs at recovery time.

To open it, select **Active Directory** > **Forests**. Every role can see the screen. Only a Backup Operator (or the break-glass account) sees the row actions menu, which holds the exclusion action.

Forests appear when a DC agent reads its forest's configuration. That read also reports forests on the far side of a trust, which this installation can see but not protect. If no forest is known yet, the screen shows **No forest is known yet.** See Deploying DC Agents.



Notices

Notice	Meaning
No forest enumeration yet	No DC agent has reported a forest enumeration. The forests listed are the ones machines claim membership of; nothing on the screen was read from Active Directory yet.
Unattributed	Some domains or domain controllers could not be placed in any forest, so they count toward none of the readiness figures. They are listed on the Domains and Domain Controllers screens. Retired domain controllers are not counted.

Findings

Findings appear in the **Recovery readiness** panel at the top of the screen, under a headline that sums up the forests' state, and cover forests that are not excluded. Each finding is marked **Blocked** (critical) or **Review** (warning). Select **Review domain coverage** to open the Domains screen.

Finding	Severity	What to do
Forests cannot begin recovery	Critical	The forest root domain has no usable recovery option. Give a writable domain controller in each root domain a full system backup policy. The root's row on the Domains screen says what is missing.
Forests would stop partway through recovery	Critical	The root domain would restore but a child domain would not. Open the Domains screen for the forest and fix every domain reported as unrecoverable.
Forest roots with a finding against them	Warning	Every other domain recovers after the root, so a finding here matters most. The root's row on the Domains screen names it.
Forests with domains at risk	Warning	Each domain would still recover, with a finding. The Domains screen states the finding and what clears it.

Finding	Severity	What to do
Forests whose topology is stale	Warning	No agent has enumerated these forests within twice the expected interval. Check on the Agents screen that a DC agent is connected and its last topology report succeeded.
Trusts needing a manual reset on the far side	Warning	A trust password reset must be mirrored on a domain controller in the trusted domain, and no agent sits there. A domain recovery always resets the domain's trusts; a forest recovery resets trusts only when it is planned as a security breach response. Install a DC agent in the partner domain, or agree the manual step with the domain's owner before a recovery needs it.
Forests had a forest-wide FSMO role move after the root's recovery point	Warning	Confirm each move was a transfer, not a seizure. A domain controller whose roles were seized elsewhere after its backup must be rebuilt, not restored.

Forest inventory

The **Forest inventory** card has filter chips for each verdict, a search box, and one row per forest.

Readiness	Severity	Meaning
Ready	Good	Every domain in the forest is covered: recoverable, root domain first, with no finding against any of them.
Root domain unrecoverable	Critical	The forest root cannot be recovered, so a forest recovery cannot start. Every other domain depends on the root's schema, Enterprise Admins, and DNS records. Chip: **cannot begin recovery**.
Domains unrecoverable	Critical	The root is recoverable, but at least one other domain is not. A recovery would restore the root and stop when it reached that domain.

Readiness	Severity	Meaning
		Chip: **would stop partway**.
Root domain at risk	Warning	The root could be recovered today, with a finding against it.
Domains at risk	Warning	One or more other domains carry a finding but would still recover.
Root domain unidentified	Unknown	The installation cannot yet judge the root domain, usually because no enumeration has said which of its domain controllers are writable.
Recoverability unconfirmed	Unknown	Every domain that can be judged is recoverable, and some are not judged yet. This is normal on a new installation until a topology enumeration arrives.
No domains found	Unknown	An enumeration reached the forest and returned no domains. Check the DC agent's last topology report for failures.
No agent in this forest	Unknown	The forest is known only because a trust names it. Nothing in it is protected, and its state cannot be judged from here. Chip: **with no agent**.

Two further chips filter the table: **stale topology** (forests not enumerated within twice their expected interval) and **excluded** (forests an operator excluded from alerts; their readiness is still shown).

Columns

Column	Description
Forest	The forest name, its root domain, and tags. **Trusted** marks a forest known only from a trust. Select the name to open the detail sheet.
Readiness	The readiness verdict, with the root domain's coverage verdict beneath. Hover for the reasoning.
Domains recoverable	Recoverable domains out of all known domains, with counts of unrecoverable, at-risk, and unjudged domains. Opens the

Column	Description
	Domains screen scoped to the forest.
Root recovery point	The age of the newest restorable backup of a writable domain controller in the root domain, and when it becomes stale. **None** means a forest recovery has nothing to start from. **Role moved since** warns that a forest-wide role changed holder after that backup.
Controllers	The number of domain controllers in the forest from every source, with counts of protected, need a backup policy, rebuilt on recovery, unprotected, without an agent, excluded, and removed machines. Domain controllers rebuilt on recovery are not backup targets and are not a shortfall in protection, except those a domain needs backed up because it has too few backup targets, which are counted as **need a backup policy**. Opens the Domain Controllers screen scoped to the forest.

Row actions

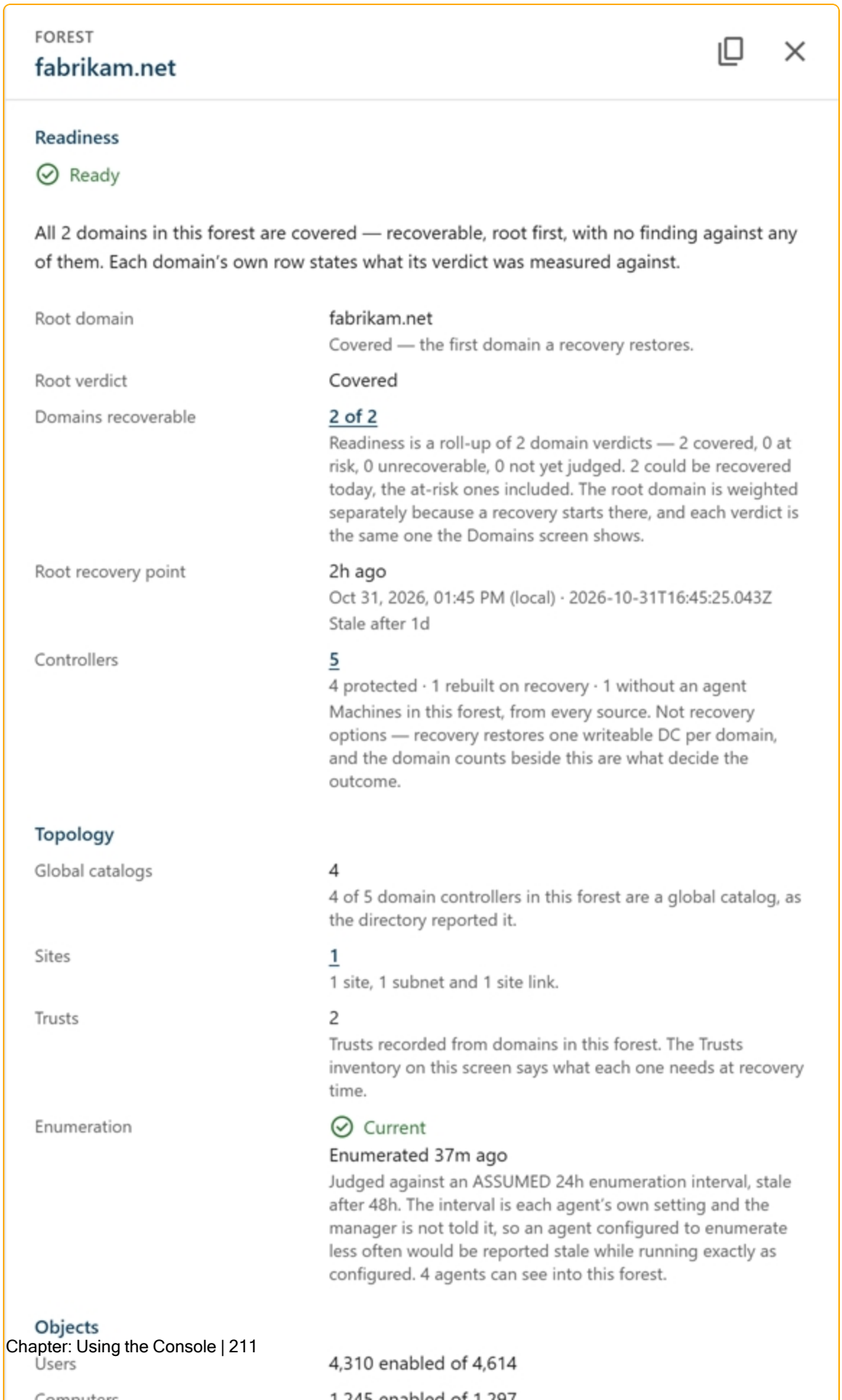
The actions menu (⋮), shown to Backup Operators, has **Exclude from alarms…** or **Edit exclusion…**, and **Domains**, which opens the Domains screen scoped to the forest. Excluding a forest excludes everything in it. See Excluding and Retiring Objects.

Forest detail sheet

Select a forest name to open its detail sheet.

Section	Contents
Readiness	The verdict and its reasoning, the exclusion if any, the root domain and its verdict, domains recoverable, the root recovery point, and controllers.
Topology	Global catalogs, sites (with a link to the Sites screen scoped to the forest), trusts, and **Enumeration**: when the forest was last enumerated and whether that is **Current**, **Enumeration due**, **Stale**, or **Never enumerated**.
Objects	Users, computers, and groups, summed from each domain's latest count. Domains with no agent are not counted.
Forest-wide operations	The schema master and domain naming master holders, and the last recorded move.

Section	Contents
masters	
Functional level	The forest functional level and schema version.
Identity	Where the forest is known from, and the schema and configuration naming contexts.


FOREST
fabrikam.net
Readiness
Ready
All 2 domains in this forest are covered — recoverable, root first, with no finding against any of them. Each domain's own row states what its verdict was measured against.
Root domain
fabrikam.net
Covered — the first domain a recovery restores.
Root verdict
Covered
Domains recoverable
2 of 2
Readiness is a roll-up of 2 domain verdicts — 2 covered, 0 at risk, 0 unrecoverable, 0 not yet judged. 2 could be recovered today, the at-risk ones included. The root domain is weighted separately because a recovery starts there, and each verdict is the same one the Domains screen shows.
Root recovery point
2h ago
Oct 31, 2026, 01:45 PM (local) · 2026-10-31T16:45:25.043Z
Stale after 1d
Controllers
5
4 protected · 1 rebuilt on recovery · 1 without an agent
Machines in this forest, from every source. Not recovery options — recovery restores one writeable DC per domain, and the domain counts beside this are what decide the outcome.
Topology
Global catalogs
4
4 of 5 domain controllers in this forest are a global catalog, as the directory reported it.
Sites
1
1 site, 1 subnet and 1 site link.
Trusts
2
Trusts recorded from domains in this forest. The Trusts inventory on this screen says what each one needs at recovery time.
Enumeration
Current
Enumerated 37m ago
Judged against an ASSUMED 24h enumeration interval, stale after 48h. The interval is each agent's own setting and the manager is not told it, so an agent configured to enumerate less often would be reported stale while running exactly as configured. 4 agents can see into this forest.
Objects
Users
4,310 enabled of 4,614

Trusts

Expand **Trust relationships** at the bottom of the screen to show the **Trusts** card. It lists every trust recorded by the domains in the known forests, as read from each domain's trusted domain objects. Trusts that discovery, an operator, or `cdr cleanup` recorded as removed are not listed; the card's subtitle counts them.

Column	Description
Trusting domain	The domain whose trusted domain object records the trust.
Partner	The other side of the trust, with its NetBIOS name.
Direction	**Inbound**, **Outbound**, **Two-way**, or **Disabled**.
Type	**Active Directory**, **Windows NT (downlevel)**, **MIT Kerberos realm**, or **DCE**.
Attributes	The trust attributes that are set, such as **Forest trust**, **SID filtering**, or **Selective authentication**. **None set** means an ordinary external trust. Attribute bits the console has no name for are counted as unrecognized rather than hidden.
At recovery	What the trust needs after a forest recovery. See the table below.
Seen in directory	When an enumeration last reported the trust.

At recovery	Meaning
Restored with the forest	A parent-child or tree-root trust inside the forest. It comes back with the forest recovery: each side keeps the trust password its restore point holds, and the recovery resets the trust only when it is planned as a security breach response.
Re-create · reset drivable	A trust that is not restored from backup and must be re-created after the recovery. An agent in the partner domain can run the mirrored password reset when a recovery resets the trust.
Re-create · manual reset	Must be re-created after the recovery, and the mirrored password reset on the partner side has to be run by hand, because no agent sits in the partner domain. Agree this with the partner domain's owner in advance.
Manual reset · far side excluded	As above, and the partner side is excluded from alerts.

For what recovery leaves to you, see What Recovery Does Not Cover and After a Recovery.

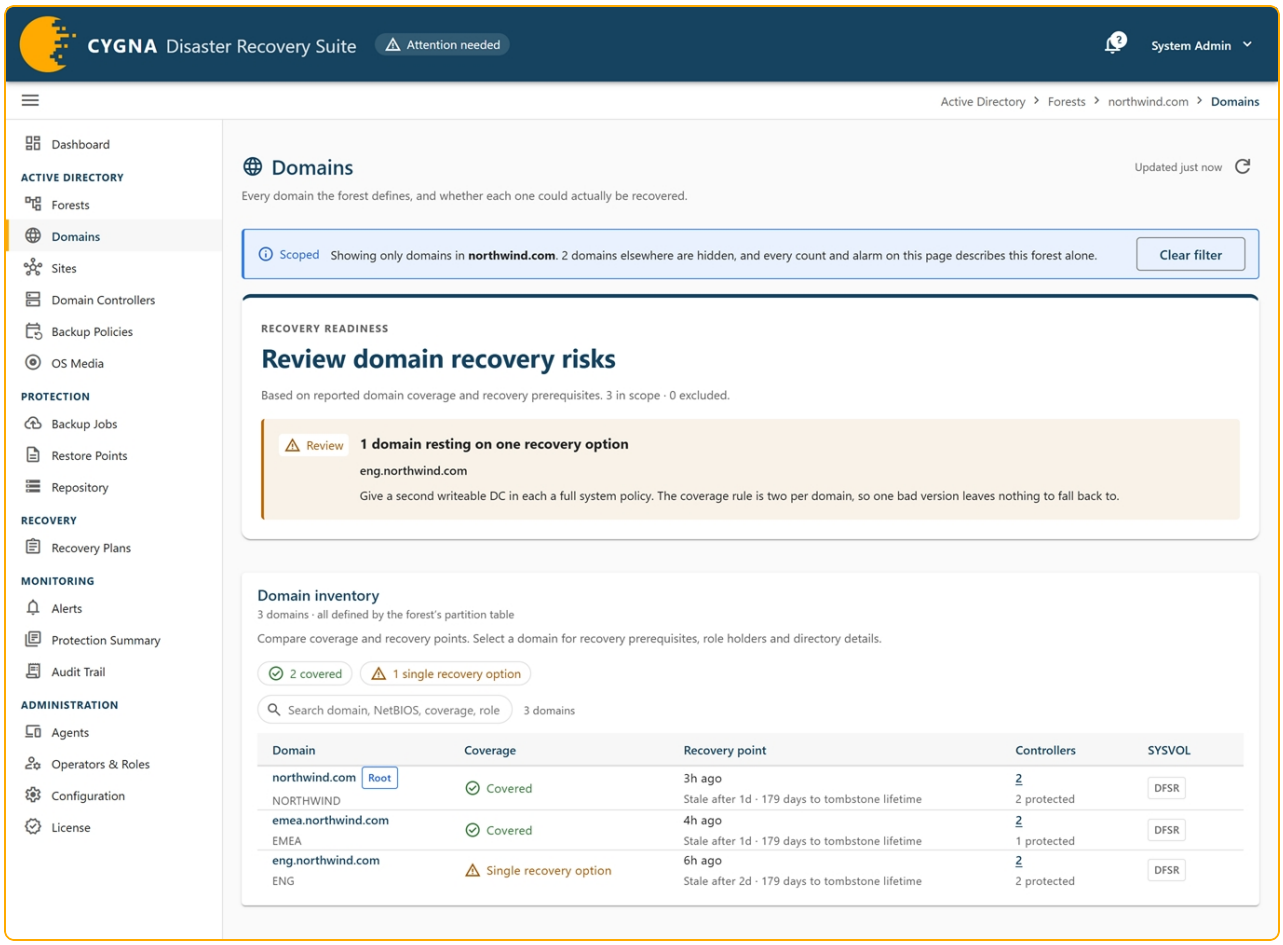
Related topics

- Coverage and Readiness
- Recovering a Forest
- Domains
- Sites
- Domain Controllers
- Excluding and Retiring Objects

Domains

The Domains screen lists every domain the forest defines and says whether each one could be recovered. A domain is recovered by restoring a writable domain controller from a full system backup taken inside the tombstone lifetime. The coverage verdict therefore asks whether such a backup exists, how fresh it is, and whether there is a second one to fall back on. The other domain controllers in the domain are rebuilt by promotion; see Backup Targets and Rebuilt Domain Controllers.

To open it, select **Active Directory** > **Domains**, or select a forest's domain count on the Forests screen. The health chip in the top bar also opens it when a domain's verdict set the chip's color. Every role can see the screen. Only a Backup Operator (or the break-glass account) sees the row actions menu, which holds the exclusion action.



Scope and notices

When you open the screen from a forest, it is scoped: a **Scoped** notice names the forest, and every count and finding describes that forest alone. Select **Clear filter** to see every domain.

Notice	Meaning
Not verified against the forest	No DC agent has reported a forest enumeration yet. The domains listed are the ones machines claim membership of; domains with nothing protected in them are missing, and directory facts such as the built-in administrator and operations masters are not collected yet.
n unplaced	Domain controllers that could not be placed in any domain. They count toward no coverage figure and are listed on the Domain Controllers screen. Excluded domain controllers and those removed from the directory are not counted here; the notice says how many more are excluded.

Notice	Meaning
n domains removed from the directory	Shown at the bottom of the screen. Domains recorded as removed from the directory. They are left out of every figure; expand the section to see their names. If the directory names one again, it returns on its own.

Findings

Findings appear in the **Recovery readiness** panel at the top of the screen, under a headline that sums up the domains' state, and cover domains that are not excluded. Each finding is marked **Blocked** (critical) or **Review** (warning).

Finding	Severity	What to do
Domains cannot be recovered	Critical	Each needs a full system backup of a writable domain controller. Install the DC agent on one and give it a full system policy. Read-only domain controllers cannot start a recovery.
Domains past the tombstone lifetime	Critical	Restoring from these backups is blocked. Take a fresh backup of a writable domain controller in each domain, then find out why backups stopped.
Domains replicate SYSVOL by FRS	Critical	Restore preflight refuses FRS. Migrate SYSVOL replication to DFSR, then take a fresh backup. Backups taken before the migration remain unusable.
Domains with no full system recovery option	Warning	Change the backup policy on a writable domain controller in each domain to **Full system**. A DC state backup cannot restore a machine whose operating system is gone.
Domains with nothing scheduled	Warning	Add or enable a backup policy on a writable domain controller, or the existing recovery point ages past the tombstone lifetime.
Domains whose recovery point is stale	Warning	Check the Backup Jobs screen for failures, then run a backup.

Finding	Severity	What to do
Domains resting on one recovery option	Warning	Give a second writable domain controller a full system policy. The coverage rule is two per domain.
Domains with only one domain controller	Warning	No backup setting can clear this. Promote a second domain controller in the domain.
Domains with no recorded built-in administrator	Warning	Recovery asks for the password of the domain's built-in administrator account (RID 500), whose name is often not Administrator. Confirm the account exists and is enabled.
Domains had an FSMO role move after the recovery point	Warning	Confirm each move was a transfer, not a seizure. A domain controller whose roles were seized elsewhere after its backup must be rebuilt, not restored.
Domains whose partition could not be read	Warning	The forest lists the domain but its naming context did not answer. Confirm the domain still exists; orphaned metadata from an unclean removal looks the same and disrupts a recovery plan.

Domain inventory

The **Domain inventory** card has filter chips for each verdict, a search box, and one row per domain.

Coverage	Severity	Meaning
Covered	Good	Two or more writable domain controllers hold full system backups, the newest inside the freshness window and well inside the tombstone lifetime.
Single recovery option	Warning	Only one writable domain controller holds a full system backup. Recovery would succeed today, with nothing to fall back to if that backup fails verification or turns out to be compromised.

Coverage	Severity	Meaning
Only one domain controller	Warning	The domain has a single domain controller, so the two-per-domain rule cannot be met. Promote a second domain controller.
Recovery point stale	Warning	The newest restorable backup of a writable domain controller is older than the freshness window (twice the shortest backup interval among the domain's candidates).
No full system recovery	Warning	Every restorable backup of a writable domain controller is DC state only. DC state recovery works; full system recovery of a lost or encrypted machine does not. Chip: **DC state only**.
No schedule	Warning	A recovery point exists, but no enabled policy covers a writable domain controller, so nothing replaces it.
Past tombstone lifetime	Critical	The newest restorable backup of a writable domain controller is older than the tombstone lifetime. Restoring it would reintroduce long-deleted objects, so it is blocked.
FRS SYSVOL — unsupported	Critical	The newest backup reports SYSVOL replicated by FRS. No backup in the domain can be used until it is migrated to DFSR.
Unrecoverable	Critical	No domain controller holds a restorable backup, or every one that does is read-only. A read-only domain controller must be rebuilt, not restored, so it cannot start a recovery.
Recoverability unconfirmed	Unknown	Domain controllers hold restorable backups, but whether they are read-only has not been collected yet. A forest enumeration answers it.
No domain controllers found	Unknown	No source has found a domain controller in this domain: its metadata may have outlived its domain controllers, or no enumeration has reached them yet.

Two further chips filter the table: **FRS SYSVOL** and **excluded**.

Columns

Column	Description
Domain	The domain name and NetBIOS name. **Root** marks the forest root domain, which is recovered first. An icon marks a domain the forest does not name, or one whose partition could not be read. Select the name to open the detail sheet.
Coverage	The coverage verdict and any exclusion. Hover for the reasoning.
Recovery point	The age of the newest restorable backup of a writable domain controller, when it becomes stale, and how many days remain before the tombstone lifetime. **Nearing tombstone lifetime** appears when less than a quarter of the lifetime remains. **None** means no writable domain controller has a restorable backup.
Controllers	The number of domain controllers, with counts of protected, need a backup policy, rebuilt on recovery, unprotected, without an agent, and excluded. Domain controllers rebuilt on recovery are not backup targets and are not a shortfall in protection, except those the domain needs backed up because it has too few backup targets, which are counted as **need a backup policy**. Opens the Domain Controllers screen scoped to the domain.
SYSVOL	The SYSVOL replication engine recorded by the newest backup: **DFSR**, or **FRS** as a critical chip.

The tombstone lifetime is read from each domain. Until it has been collected, the screen assumes the Active Directory default and says so.

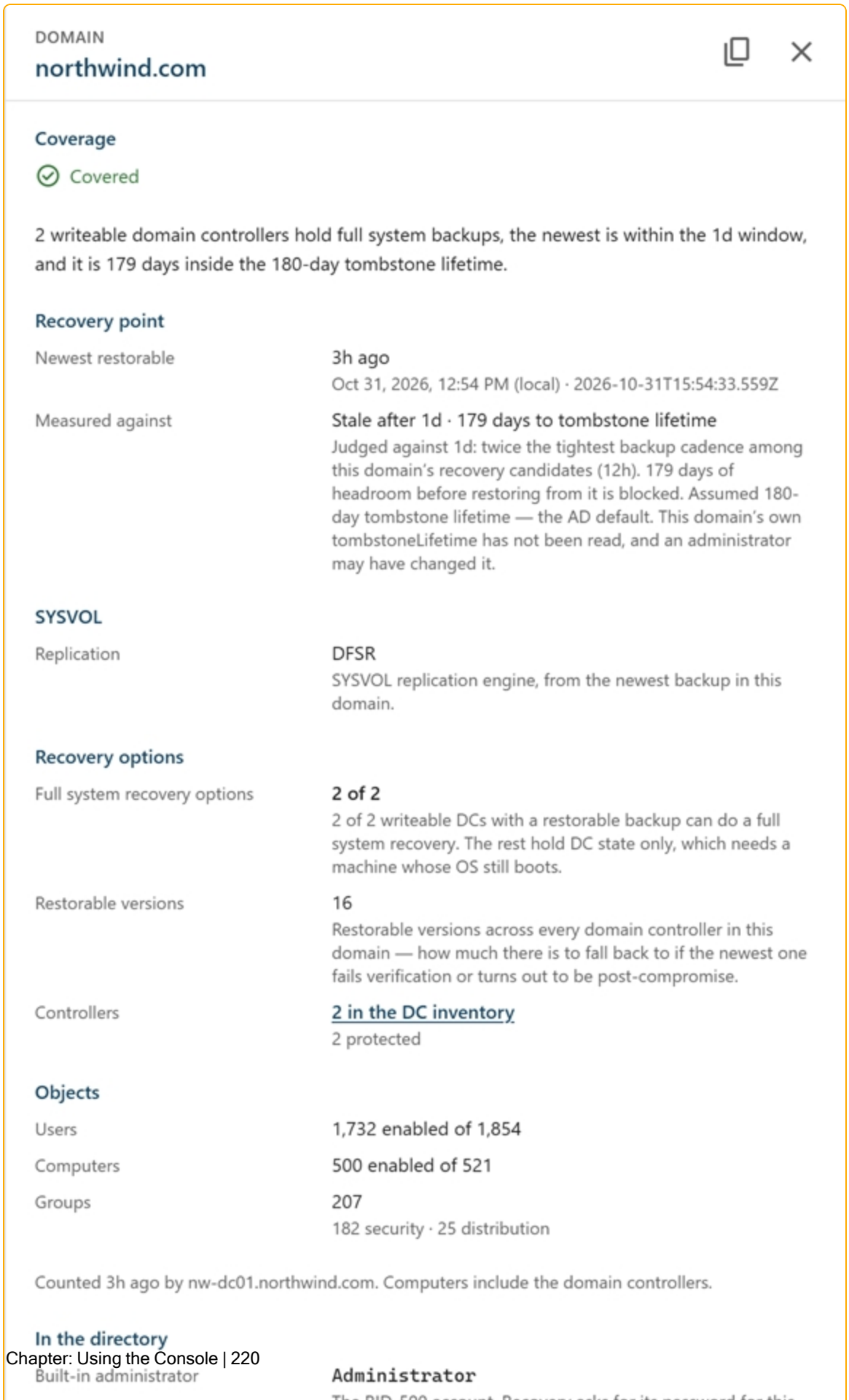
Row actions

The actions menu (⋮), shown to Backup Operators, has **Exclude from alarms…** or **Edit exclusion…**, and **Domain controllers**, which opens the Domain Controllers screen scoped to the domain. See Excluding and Retiring Objects.

Domain detail sheet

Section	Contents
Coverage	The verdict, its reasoning, and the exclusion if any.
Recovery point	**Newest restorable** and **Measured against**: the freshness window and tombstone lifetime the recovery point is judged by.

Section	Contents
SYSVOL	The replication engine and what it means for restore preflight.
Recovery options	**Full system recovery options** (writable domain controllers whose restorable backup is full system), **Restorable versions** across the domain, and **Controllers**.
Objects	Users, computers, and groups, as counted by an agent in the domain.
In the directory	**Built-in administrator** (the RID 500 account whose password recovery asks for), **Operations masters** and the last role move, **Functional level**, **Forest root**, **Seen in forest**, and **Known from**.
Identity	NetBIOS name, forest, domain SID, and naming context.


DOMAIN
northwind.com
Coverage
Covered
2 writeable domain controllers hold full system backups, the newest is within the 1d window, and it is 179 days inside the 180-day tombstone lifetime.
Recovery point
Newest restorable
3h ago
Oct 31, 2026, 12:54 PM (local) · 2026-10-31T15:54:33.559Z
Measured against
Stale after 1d · 179 days to tombstone lifetime
Judged against 1d: twice the tightest backup cadence among this domain's recovery candidates (12h). 179 days of headroom before restoring from it is blocked. Assumed 180-day tombstone lifetime — the AD default. This domain's own tombstoneLifetime has not been read, and an administrator may have changed it.
SYSVOL
Replication
DFSR
SYSVOL replication engine, from the newest backup in this domain.
Recovery options
Full system recovery options
2 of 2
2 of 2 writeable DCs with a restorable backup can do a full system recovery. The rest hold DC state only, which needs a machine whose OS still boots.
Restorable versions
16
Restorable versions across every domain controller in this domain — how much there is to fall back to if the newest one fails verification or turns out to be post-compromise.
Controllers
2 in the DC inventory
2 protected
Objects
Users
1,732 enabled of 1,854
Computers
500 enabled of 521
Groups
207
182 security · 25 distribution
Counted 3h ago by nw-dc01.northwind.com. Computers include the domain controllers.
In the directory
Built-in administrator
Administrator
The RID-500 account. Recovery asks for its password for this

Related topics

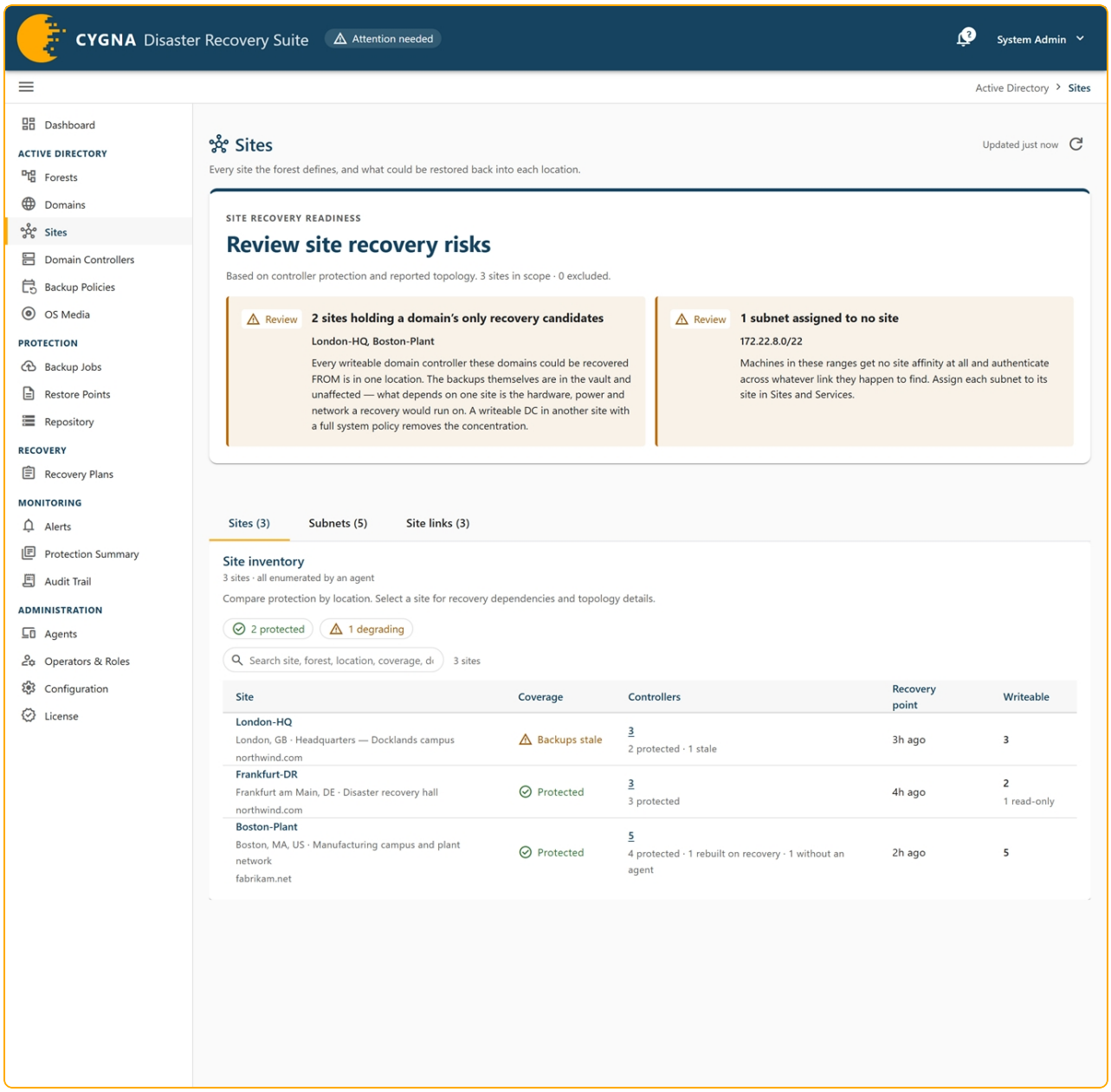
- Coverage and Readiness
- Backup Types
- Recovering a Domain
- Recovery Requirements
- Forests
- Domain Controllers
- Managing Backup Policies

Sites

The Sites screen lists every site the forest defines and says what could be restored back into each location. It also shows the subnet map and the site links across which a recovery repopulates each location.

Unlike the domain and forest verdicts, a site's verdict is a roll-up of the protection of the backup targets in it, the domain controllers you back up. Domain controllers that recovery rebuilds by promotion are counted beside them and are not findings. No site verdict is critical: a location comes back by promotion, and whether a domain could be recovered is that domain's verdict on the Domains screen. See Backup Targets and Rebuilt Domain Controllers.

To open the screen, select **Active Directory** > **Sites**, or select the site count in a forest's detail sheet. Every role can see the screen. Only a Backup Operator (or the break-glass account) sees the row actions menu, which holds the exclusion action.



Scope and notices

When you open the screen from a forest, it is scoped: a **Scoped** notice names the forest, and every count and finding describes that forest alone. Select **Clear filter** to see every site.

Notice	Meaning
No site attribution yet	No domain controller has a site recorded, so which sites are occupied is not known. Every site is reported as unjudged rather than empty until a forest enumeration arrives.
Partly enumerated	Some sites are known only because domain controllers name them; no site object was enumerated, so they have no description, subnets, or site links. Usually a topology report's site step failed; the Agents screen shows the last report.
Unplaced	Domain controllers with no site recorded. They count toward no site and are listed on the Domain Controllers screen. Domain controllers removed from the directory are not counted.

Findings

Findings appear in the **Site recovery readiness** panel at the top of the screen, under a headline that sums up the sites' state, and cover sites that are not excluded. Every site finding is a warning, marked **Review**. Sites with no domain controllers, sites whose domain controllers are all rebuilt on recovery, and sites whose occupancy is unknown raise no coverage finding.

Finding	Severity	What to do
Sites whose backup targets have nothing to restore	Warning	The location has domain controllers under a backup policy, and not one holds a restorable backup. Open the Domain Controllers screen scoped to each site to see why nothing is landing.
Sites only partly protected	Warning	Some backup targets in the location have nothing to restore from. Open the Domain Controllers screen scoped to each site to see why nothing is landing.
Sites where a recovery could not begin	Warning	Every domain controller in the location whose flag has been read is read-only, and a recovery starts by restoring a writable domain controller. Place a writable domain controller there, or accept that recovery starts elsewhere.

Finding	Severity	What to do
Sites holding a domain's only recovery candidates	Warning	Every writable domain controller a domain could be recovered from is in this one location. The backups are safe in the vault, but the recovery depends on this location's hardware, power, and network. Add a writable domain controller with a full system policy in another site.
Sites with no global catalog	Warning	Everything in the location that needs a global catalog crosses a site link to reach one. Promote a domain controller to global catalog, or confirm the link is dependable.
Occupied sites reached by no site link	Warning	Replication cannot be built for a site no link names, and after a recovery nothing would promote back into it. Add each site to a site link in Active Directory Sites and Services.
Occupied sites with no subnet	Warning	Nothing places clients in these sites. Assign the location's IP ranges to the site.
Subnets assigned to no site	Warning	Machines in these ranges have no site affinity. Assign each subnet to its site.

Site inventory

The **Site inventory** card on the Sites tab has filter chips for each verdict.

Coverage	Severity	Meaning
Protected	Good	Every backup target in the location is protected. Domain controllers rebuilt on recovery do not change this verdict.
Partly protected	Warning	Some domain controllers have a restore point, and some backup targets have nothing to restore from.
No schedule	Warning	No backup target lacks a restore point, and some domain controllers have a restore point

Coverage	Severity	Meaning
		that nothing is scheduled to replace.
Backups stale	Warning	Some domain controllers have a restore point older than their policy window.
Last backup failed	Warning	Restore points are fresh, and some machines' newest backup attempt failed.
Nothing to restore	Warning	The location has backup targets, and no domain controller in it holds a restorable backup. The location would come back by promotion; whether a domain could be recovered is that domain's verdict.
Rebuilt on recovery	Information	The site has domain controllers and none of them is a backup target, so a recovery brings the location back by promotion. This is normal in a large estate, not a fault.
No domain controllers	Information	The forest defines the site and no domain controller sits in it. This is normal design, such as a branch office, not a fault.
Occupancy unknown	Unknown	No domain controller anywhere has a site recorded, so the site cannot be judged.

The chips group **No schedule**, **Backups stale**, and **Last backup failed** under **degrading**. Two further chips filter the table: **cannot start a recovery** (every domain controller whose flag has been read is read-only) and **excluded**.

Sites tab

Column	Description
Site	The site name, its location description, and its forest. **Not enumerated** marks a site known only because domain controllers name it. Select the name to open the detail sheet.
Coverage	The coverage verdict and any exclusion. Hover for the reasoning.
Controllers	The number of domain controllers in the site, with counts by protection state, for example `2 protected · 4 rebuilt on recovery`. Opens the Domain Controllers screen scoped to the site.

Column	Description
Recovery point	The age of the newest restorable backup taken in this location. The backup itself is in the vault, so losing the location does not lose it. **None** means nothing in the location can be restored.
Writable	How many domain controllers are confirmed writable and how many are read-only. **Read-only only** means a recovery cannot begin here.

The actions menu (⋮), shown to Backup Operators, has **Exclude from alarms…** or **Edit exclusion…**, and **Domain controllers**. A site that has not been enumerated cannot be excluded yet. See Excluding and Retiring Objects.

Subnets tab

Column	Description
Prefix	The subnet.
Site	The site the subnet is assigned to. **No site** is a misconfiguration: machines in the range get no site affinity. **Unknown site** means the subnet names a site this installation has no record of, usually because an enumeration failed partway.
Forest	The forest that reported the subnet.
Location	The subnet's description and location attributes.
Seen in directory	When an enumeration last reported the subnet.

Site links tab

Column	Description
Link	The site link name and its forest.
Transport	**IP** or **SMTP**. SMTP cannot replicate a domain naming context, so nothing a domain recovery depends on travels over an SMTP link.
Cost	The link cost. Replication prefers the cheapest path.
Replicates	The replication interval.
Sites joined	The sites the link connects. Members this installation has no

Column	Description
	record of are counted as unknown.
Seen in directory	When an enumeration last reported the link.

Site detail sheet

Section	Contents
Coverage	The verdict, its reasoning, and the exclusion if any.
Recovery	Domain controllers, writable domain controllers, and the recovery point.
Topology	Global catalogs; domains with a domain controller in the site, and whether a domain's recovery candidates are all concentrated here; operations master roles held in the site; subnets; and site links with the cheapest cost and slowest interval.
Directory	Forest, location, where the site is known from, and when it was last seen in the directory.
Distinguished name	The site object's distinguished name. A site with none has not been enumerated and cannot be excluded.

Operations master roles concentrated in one site all have to be seized together if that location is lost. The Topology section shows this so you can plan for it.

Related topics

- Coverage and Readiness
- Forests
- Domain Controllers
- Recovering a Forest
- Excluding and Retiring Objects

Domain Controllers

The Domain Controllers screen lists every domain controller this installation knows to exist, including the ones it is not protecting. The list comes from the directory: a DC agent reports every domain controller its forest defines, including machines with no agent. For each

machine the screen shows its protection verdict, its newest restorable backup, its schedule, and its agent.

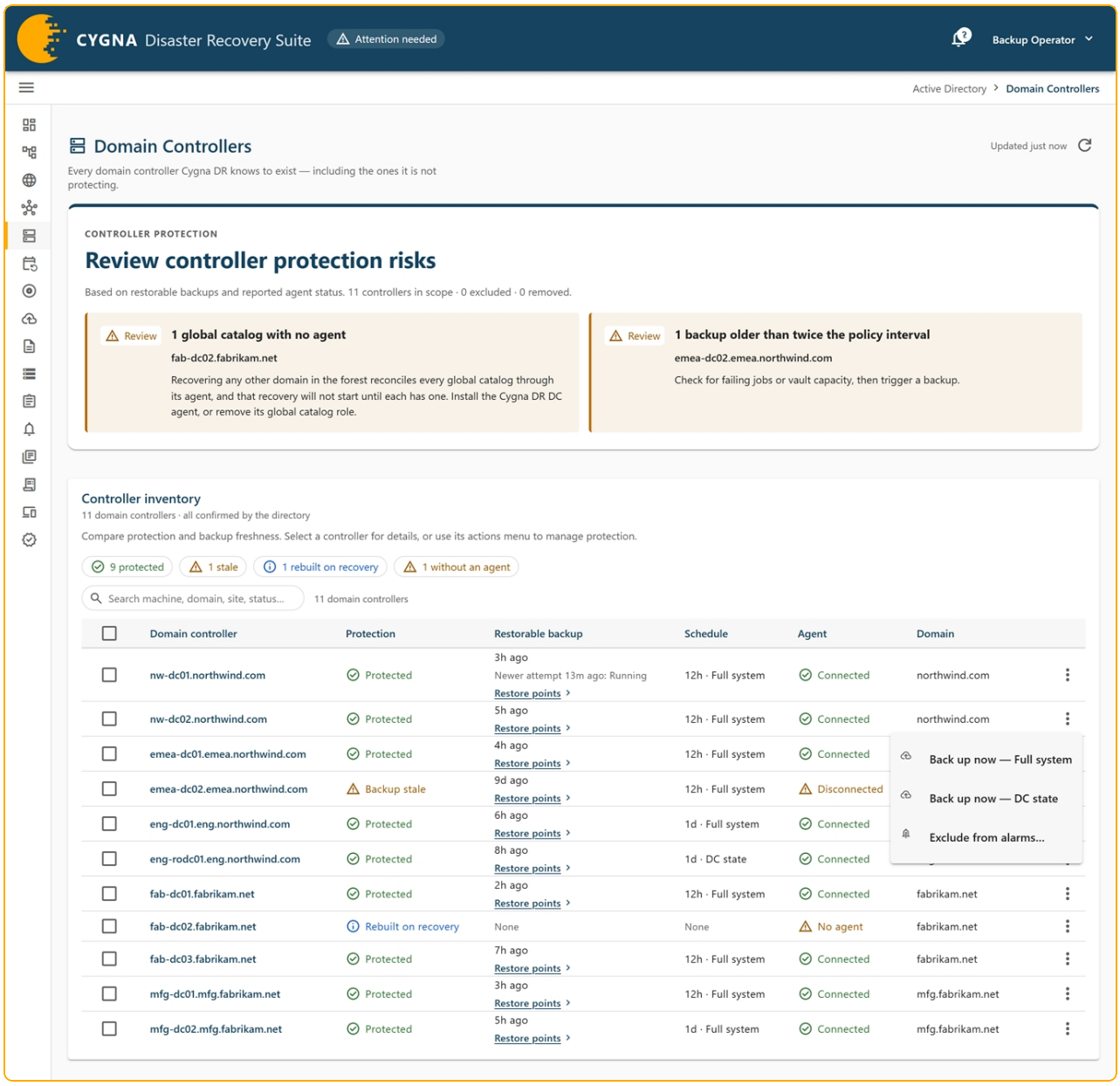
A candidate machine, which runs the DC agent but is not a domain controller, is not listed here and does not count in the screen's figures or alerts. It is listed on the **Agents** screen. A machine that the directory lists, or that has a backup, stays listed even if its agent reports that it is not a domain controller.

This is also where you start an ad hoc backup, install or update the DC agent from the console, exclude a machine from alerts, and handle domain controllers that have left the directory.

To open the screen, select **Active Directory** > **Domain Controllers**. You also reach it from the health chip in the top bar, and from the controller counts on the Forests, Domains, and Sites screens, which open it scoped. Every role can see the screen. The actions depend on your role:

Role	Actions on this screen
Backup Operator	**Back up now**, **Stop backup**, exclusions, and recording machines as removed from the directory.
System Admin	**Install agent** and **Update agent**.
Viewer / Auditor, Restore Operator	None. The selection checkboxes and actions menu are not shown.

The screen refreshes every 10 seconds.



Notices

Notice	Meaning
Scoped	The screen shows only the domain controllers in a forest, domain, or site, and every count and alert describes that scope. Machines with no forest, domain, or site recorded are left out rather than assumed to belong. Select **Clear filter** to see every machine.
Not verified against	No DC agent has reported a forest enumeration yet, so the list

Notice	Meaning
the directory	shows only machines that have enrolled or been backed up. Domain controllers without an agent appear after the first topology report.
Not started	The Manager did not start a backup you requested, and says why. If the attempt was recorded as a failed job, select **See the recorded failure**.
Not stopped	The Manager did not stop the backup, and says why.
Request failed	The request may not have reached the Manager. Check the Backup Jobs screen before you try again.

Controller protection panel

The **Controller protection** panel at the top of the screen sums up the scope in a headline, such as "*n* controllers have nothing to restore", **Review controller protection risks**, **Controller protection is not fully confirmed**, or **All scoped controllers have protected backups**. Below it, a scope line gives how many controllers are in scope, excluded, and removed, and how many the directory has not confirmed. A domain controller that recovery rebuilds raises no finding and does not count against the headline. Each raised alert is listed as a finding with the affected machines and what to do, marked **Action required** (critical) or **Review** (warning):

Alert	Severity	What to do
Global catalogs with no agent	Warning	A recovery of any other domain in the forest reconciles every global catalog through its agent, and does not start until each has one. Install the DC agent, or remove the machine's global catalog role.
Backup targets have no restorable backup	Warning	A policy covers each machine and nothing of it is in the vault. Install the agent where there is none; otherwise check the machine on the Backup Jobs screen. Whether this costs a domain its recovery is that domain's verdict on the Domains screen.
Backups older than twice the policy interval	Warning	Check for failing jobs or vault capacity, then run a backup.

Alert	Severity	What to do
DCs with a backup but no active schedule	Warning	Add or enable a backup policy, or the existing restore point only ages.
Domain controllers no agent can reach	Warning	Check the network path from the agents (DNS, and UDP and TCP 389). If the domain controller is gone, clean up its metadata or mark it as removed.
Machines backed up but no longer in the directory	Warning	Confirm each was decommissioned deliberately. A domain controller that left the directory unexpectedly needs investigating.

Controller inventory

The **Controller inventory** card has filter chips, a search box, and one row per domain controller. The chips count **protected**, **unprotected**, **stale**, **failing**, **unscheduled**, and **rebuilt on recovery** machines, plus machines **without an agent**, machines **not reachable** by any agent, **excluded** machines, and machines **removed** from the directory. The **without an agent** chip is a warning only when one of those machines is a global catalog that a recovery needs an agent on; otherwise it is information. Removed machines are listed only while the **removed** chip is selected.

Protection verdicts

The protection verdict is judged against a window of twice the machine's backup policy interval.

Protection	Severity	Meaning
Protected	Good	A restorable backup exists within the window, and the last attempt succeeded.
Last backup failed	Warning	The newest restorable backup is within the window, but the most recent attempt failed. A backup refused because the machine had no agent, made before its current agent enrolled, does not count as a failed attempt.
Backup stale	Warning	The newest restorable backup is older than the window.
No schedule	Warning	A restorable backup exists, but its policy is

Protection	Severity	Meaning
		disabled or no policy covers the machine, so nothing will replace it.
Unprotected	Warning	A backup target with no restorable backup: an enabled policy covers the machine, but no agent has enrolled, no backup has run, or every backup has been removed from the vault or never reached it.
Rebuilt on recovery	Information	Not a backup target: no enabled policy covers the machine and nothing of it is in the vault. A recovery rebuilds it by promotion, so this is not a finding. Whether its domain can be recovered is the domain's verdict.

Hover over the verdict to see which case applies. For how backup targets and rebuilt domain controllers are decided, see Backup Targets and Rebuilt Domain Controllers.

Columns

Column	Description
Domain controller	The machine's fully qualified name. Select it to open the detail sheet. An icon marks a machine no directory enumeration has confirmed, or one that was backed up in the past but that neither the directory nor an agent reports now.
Protection	The protection verdict, with the exclusion chip and **Removed from the directory** where they apply.
Restorable backup	The age of the newest restorable backup, or **Never** as a warning chip. A machine that is rebuilt on recovery shows **None** as plain text. Notes flag a backup whose content is no longer in the vault, one that never reached the vault, and a newer failed attempt. **Restore points ›** opens the machine's restore points.
Schedule	The backup interval and type, **Disabled**, or **None**. **Disabled** and **None** are warning chips, except on a machine that is rebuilt on recovery, where they are plain text. Hover to see when the next backup is due. For a machine whose DC agent has never come online, the cell ends in **· waiting for agent**: the schedule is held, and its first backup runs as soon as an agent connects.

Column	Description
Agent	**Connected**, **Disconnected**, **Never reported**, or **No agent**. **No agent** is a warning on a backup target and on a global catalog that a recovery of another domain in the forest needs an agent on; on any other machine it is information, because recovery rebuilds it and needs no agent there. Hover to see which case applies.
Domain	The machine's domain. Opens the Domains screen.

Tables show 50, 100, 250, or 500 rows per page.

When you open a domain controller from a link on another screen, such as the **Domain controller** item in a row menu or a breadcrumb, the list is filtered to that machine as well, with the notice **Showing** *machine* **only.** The counts and alerts above the card still describe every controller. Select **Clear filter** to list every domain controller again; the search, sort, and verdict filter are kept.

Row actions

The actions menu (⋮) at the end of each row shows only the actions your role allows.

Action	Role	What it does
Back up now — Full system / Back up now — DC state	Backup Operator	Queues an ad hoc backup of that type at once, without confirmation. Select **View backup jobs** on the confirmation to follow it. Not available when the machine has no agent. While a recovery is working on the machine, the backup is refused with **A recovery is working on this DC** and nothing is recorded; the machine's scheduled backup runs when the recovery is done. See Running a Backup Now.
Stop backup…	Backup Operator	Shown while a backup is queued or running. A queued backup is cancelled at once. For a running one, the agent stops the job cleanly and releases its snapshot. The run produces no restorable version, the machine keeps its last good one, and nothing already in the vault is touched.
Install agent… / Update agent…	System Admin	Opens the agent deployment dialog. When the agent is already current, the menu says

Action	Role	What it does
		so; reinstall from the Agents screen instead. Requires a DC agent package in the Manager's **Package folder**.
Exclude from alerts… / Edit exclusion…	Backup Operator	Opens the exclusion dialog. See Excluding and Retiring Objects.
Mark as removed from the directory…	Backup Operator	Records that the machine is no longer a domain controller. Offered only when the latest complete enumeration of its forest did not include it.
Withdraw removal record…	Backup Operator	On a removed machine: lists and counts it again.
Forget this machine…	Backup Operator	On a removed machine: deletes the Manager's record of it.

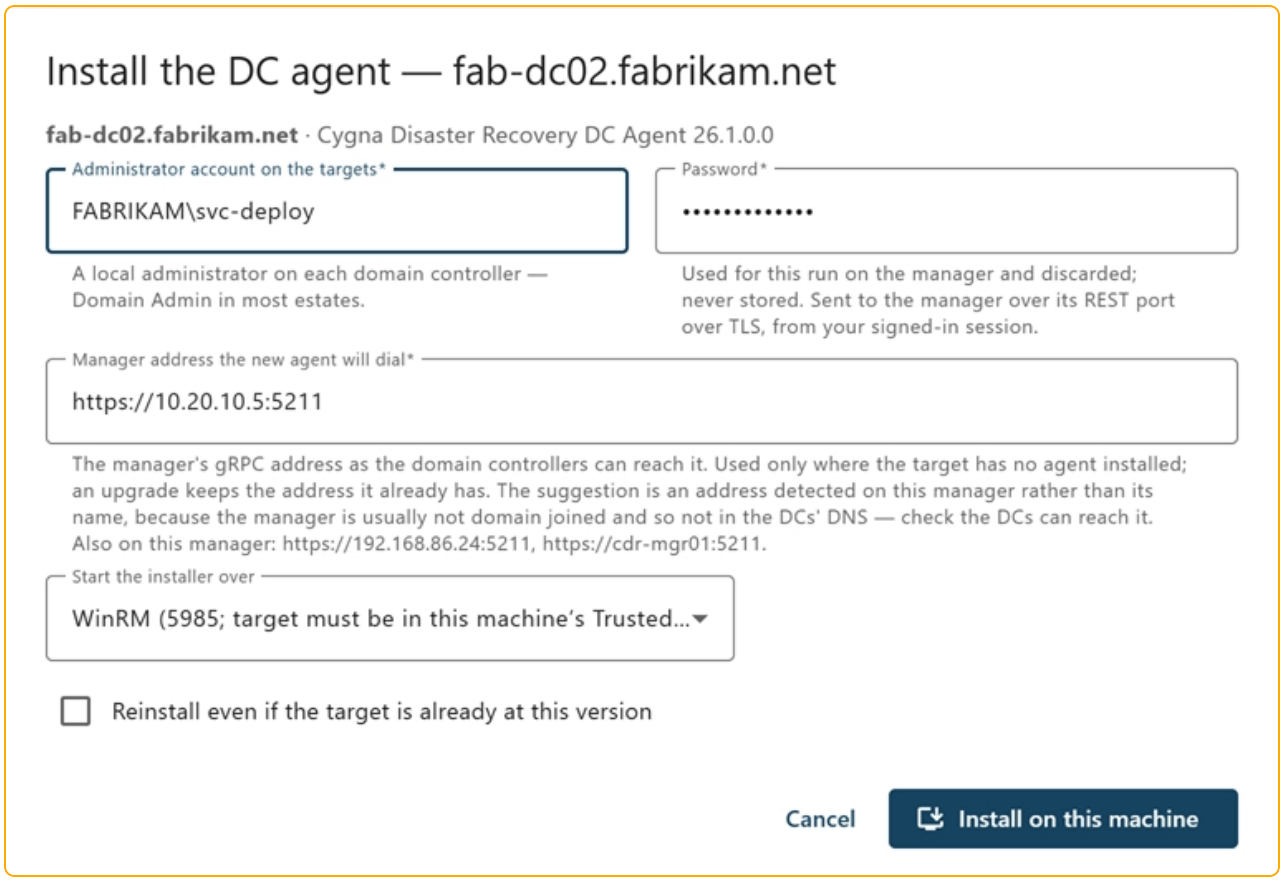
Bulk actions

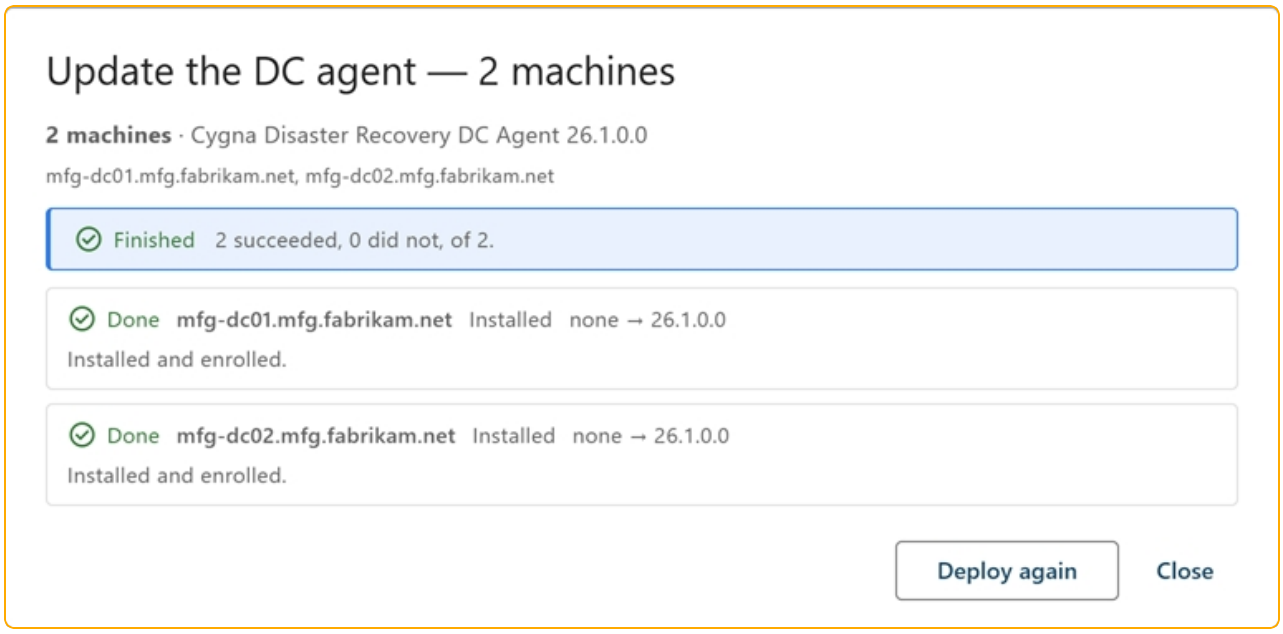
Select machines with the checkboxes to show the bulk bar. Depending on your role it offers **Back up *n* now** (choose **Full system** or **DC state**), **Stop *n* running backups…**, **Deploy the agent to *n*…**, **Exclude *n* from alarms…**, and **Clear *n* exclusions**. Select **Select all *n* matching** to include every machine that matches the current filter and search, not only the current page. If some machines are refused, a notice lists each with the reason.

Installing and updating the agent

The **Install the DC agent** or **Update the DC agent** dialog (for several machines, **Install or update the DC agent** when any of them has no agent yet) pushes the DC agent package from the Manager to the selected domain controllers. It asks for an administrator account on the targets and its password, the Manager address a new agent will dial, and whether to start the installer over WinRM or DCOM. When the agent can update itself, **Update the DC agent** asks for no account: the Manager sends the package over the agent's own connection. For each field, see Deploying DC Agents.

After you start, the dialog follows the run and shows each target's status and outcome, such as **Installed**, **Upgraded**, **Already current**, **Unreachable**, **Credential refused**, or **Install failed**, with the installer log path and transcript. You can close the dialog; the run continues. For prerequisites and the full procedure, see Deploying DC Agents and Maintaining Agents.





Domain controllers removed from the directory

When a domain controller is demoted or deleted, the inventory does not simply drop it. Discovery records a machine as removed from the directory once every agent with a current, complete read of the directory stops seeing it. If more than a quarter of a forest's

domain controllers disappear at once, discovery holds the removals for review and raises an alert instead, because that pattern can indicate an attack or an incomplete cleanup. You can then record each removal with `cdr cleanup retire`; see Directory Cleanup Commands.

A removed machine leaves every count and alert, is listed only under the **removed** chip, and shows **Removed from the directory** with the reason, who recorded it, and when. Nothing in Active Directory changes. If an enumeration includes the machine again, the removal record is withdrawn automatically.

- **Mark as removed**—Records the removal yourself. Enter **Why it is gone** (required) and select **Mark as removed**.
- **Withdraw the record**—The machine is listed and counted again, aging as an unreachable domain controller would. Discovery will not record it as removed again unless the directory lists it and then drops it once more.
- **Forget it**—Deletes the Manager's record of the machine. Type the machine's name to confirm. The audit trail keeps a copy of what was deleted, and if the directory ever defines the name again, it is discovered as a new domain controller. A machine can be forgotten only when it is recorded as removed, no agent is enrolled for it, no backup jobs exist for it, and no backup schedule names it; otherwise the menu lists what is keeping it.

For when to use each, see Excluding and Retiring Objects.

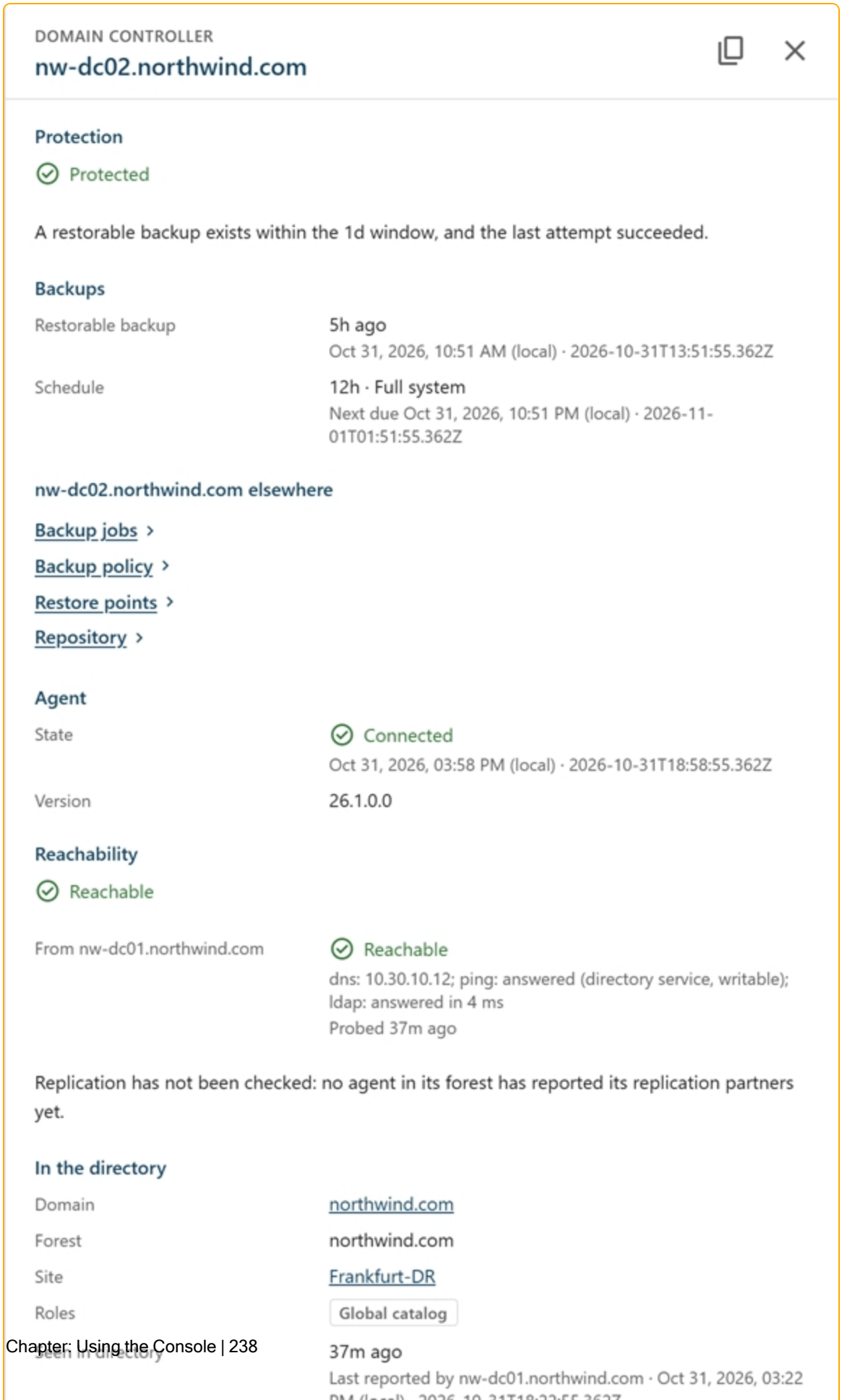
Domain controller detail sheet

Select a machine name to open its detail sheet.

Section	Contents
Protection	The verdict and its reasoning, the exclusion if any, and the removal record. Notes say whether the latest enumeration of the forest still includes the machine.
Backups	The restorable backup, and the schedule with when the next backup is due.
machine elsewhere	Links to the machine's **Backup jobs**, **Backup policy**, **Restore points**, and **Repository** entries.
Agent	The agent's state and version. A note appears when the Manager holds a newer DC agent package.
Reachability	An overall reachability verdict, then one row per agent that probed the machine, with the result and when. Below it, whether replication from the machine has been observed.

Section	Contents
In the directory	Domain, forest, and site; roles (global catalog, read-only domain controller, and operations master roles); when the directory last listed the machine, and by how many of the forest's agents; and where the machine is known from.
Operating system	The build and revision (UBR). Full system recovery preflight checks build and revision together, so a missing revision is flagged.

Reachability	Meaning
Reachable	Agents can reach the machine over LDAP.
Reachable from some agents	Some probing agents reach it and some do not.
Not reachable from any agent	No probing agent reached it. Individual probe results say why, for example **Name does not resolve** or **No answer on LDAP**.
A different DC answers at this name	The name resolves to another domain controller.
Not probed	No agent has probed the machine yet. Agents report probes with their next topology enumeration.


DOMAIN CONTROLLER
nw-dc02.northwind.com
Protection
Protected
A restorable backup exists within the 1d window, and the last attempt succeeded.
Backups
Restorable backup
5h ago
Oct 31, 2026, 10:51 AM (local) · 2026-10-31T13:51:55.362Z
Schedule
12h · Full system
Next due Oct 31, 2026, 10:51 PM (local) · 2026-11-01T01:51:55.362Z
nw-dc02.northwind.com elsewhere
Backup jobs
Backup policy
Restore points
Repository
Agent
State
Connected
Oct 31, 2026, 03:58 PM (local) · 2026-10-31T18:58:55.362Z
Version
26.1.0.0
Reachability
Reachable
From nw-dc01.northwind.com
Reachable
dns: 10.30.10.12; ping: answered (directory service, writable); ldap: answered in 4 ms
Probed 37m ago
Replication has not been checked: no agent in its forest has reported its replication partners yet.
In the directory
Domain
northwind.com
Forest
northwind.com
Site
Frankfurt-DR
Roles
Global catalog
37m ago
Last reported by nw-dc01.northwind.com · Oct 31, 2026, 03:22

Related topics

- Coverage and Readiness
- Running a Backup Now
- Monitoring and Cancelling Backup Jobs
- Backup Policies
- Deploying DC Agents
- Agents
- Excluding and Retiring Objects

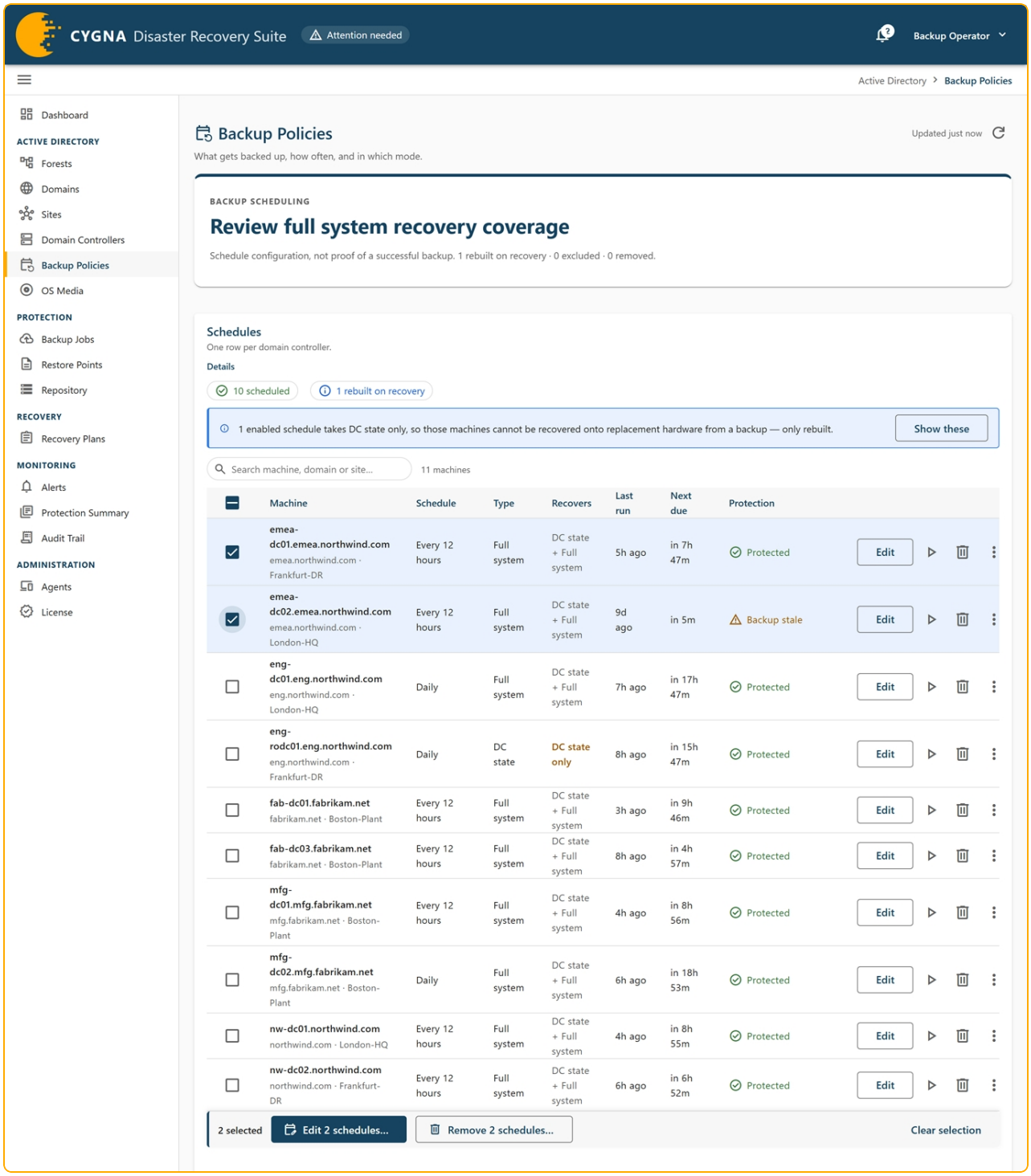
Backup Policies

The Backup Policies screen shows what gets backed up, how often, and with which backup type. It has one row per known domain controller. A policy belongs to one machine and sets its backup interval, an optional start time, its backup type, and whether it is enabled. No machine has a policy until you create one.

To open the screen, select **Active Directory** > **Backup Policies**. Every role can see it. Creating, changing, and removing schedules, and starting a backup from a row, require the Backup Operator role (or the break-glass account); the Manager refuses a schedule change from any other account with **The manager refused this request.**, and the back-up-now button is not shown to it.

For step-by-step procedures, see Managing Backup Policies. To set up policies on a new installation, see Creating Your First Backup Policies.

If no domain controller is known yet, the screen shows **No domain controllers are known to this manager.** A schedule is attached to a machine, so install the DC agent first; see Deploying DC Agents.



Schedules

The **Schedules** card lists first the machines that need a schedule: those with backups and no schedule, and those their domain needs backed up. Scheduled machines follow, then the machines rebuilt on recovery, which need none, and machines excluded from alerts come last.

Filter chip	Shows
scheduled	Machines with a schedule.
need a backup policy	Critical. Shown when a domain has fewer backup targets than a recovery needs. The count is how many more domain controllers need a schedule. Selected, it lists every domain controller that could make up the difference: writable, in scope, and rebuilt on recovery. Any of them will do.
rebuilt on recovery	Machines with no enabled schedule and no restore point, which a recovery rebuilds by promotion. Information, not a finding. Schedule one only to make it a backup target. See Backup Targets and Rebuilt Domain Controllers.
unscheduled	Machines with restore points and no schedule, so their restore points will age. A warning.
disabled	Machines whose schedule is disabled and that still have restore points.
due now	Machines whose next backup is due.
excluded	Machines excluded from alerts. They are left out of the other counts but still listed and can still be scheduled.
removed	Machines recorded as removed from the directory. Those that still have a schedule stay listed so you can remove it.

When a domain has fewer backup targets than a recovery needs, the panel at the top of the screen shows ***n* need a backup policy** as **Action required**, and names each domain with how many more it needs. Select **Review candidates** to list the domain controllers that could take the schedule. A schedule on a machine without a DC agent waits for one, so when the candidates have no agent the panel says to install one first, and each such row reads **Its domain needs another backup. Install an agent here before scheduling it.** Machines with backups and no schedule are a separate warning, ***n* with backups and no schedule**.

When enabled schedules take DC state backups, a notice says how many: those machines can be recovered only with DC state recovery, not onto replacement hardware. Select **Show these** to list them.

Columns

Column	Description
Machine	The domain controller, with its domain and site. Notes appear when no agent is enrolled (the schedule will not run), when the machine is excluded, and when it is **Removed from the directory** but backups are still being dispatched to it (remove the schedule).
Schedule	The interval, such as **Daily**, **Hourly**, or **Every 4 hours**, followed by the start time if the schedule has one, for example **Daily from 22:00**. The start time is the Manager's local time. A disabled schedule shows the same text followed by **· off**. **None** means no schedule.
Type	**Full system** or **DC state**.
Recovers	**DC state + Full system** for a full system backup, or **DC state only** for a DC state backup.
Last run	When the scheduler last dispatched a backup. This is not necessarily the last successful backup; see the Backup Jobs screen for outcomes.
Next due	How long until the next backup is due (for example **in 9h 40m**), **Due now**, **Not scheduled**, or **Waiting for agent**. **Waiting for agent** means no DC agent has come online for the machine yet: the schedule is held, and its first backup runs as soon as one does.
Protection	The machine's protection verdict, as on the Domain Controllers screen.

Row actions

- **Schedule**–Shown when the machine has no schedule. Opens the editor for a new schedule.
- **Edit**–Changes the state, interval, start time, or backup type.
- **Back up now** (play icon)–Shown on a row with a schedule, except for a machine recorded as removed from the directory or one whose DC agent has never connected. Starts a backup of the machine now, of the type its schedule takes; the tooltip names it, for example **Back up now (Full system)**. The backup does not change the schedule's last or next run. See Running a Backup Now.
- **Remove schedule** (delete icon)–Removes the schedule and discards its interval, start time, and type. To pause backups and keep the settings, disable the schedule instead.

- **Actions menu (⋮)**–Links to the same machine on other screens: **Backup jobs**, **Restore points**, and **Repository**, each filtered to the machine, and **Domain controller**, which opens its details on the **Domain Controllers** screen with the list filtered to it.

While a schedule is being saved, **Edit**, **Back up now**, and **Remove schedule** are unavailable on every row, so a backup never starts with the type being changed away from.

Schedule, **Edit**, **Remove schedule**, and the bulk bar are shown only to operators with the `backup.manage` capability (Backup Operator role). **Back up now** needs `backup.run`.

Select machines with the checkboxes to show the bulk bar, with **Edit** *n* **schedules…**, **Remove** *n* **schedules…**, and **Select all** *n* **matching**.

Schedule editor

The editor opens as **Schedule** *machine* for a new schedule or **Edit schedule for** *machine*. A new schedule starts as enabled, daily, Full system. Changes take effect as soon as you save them.

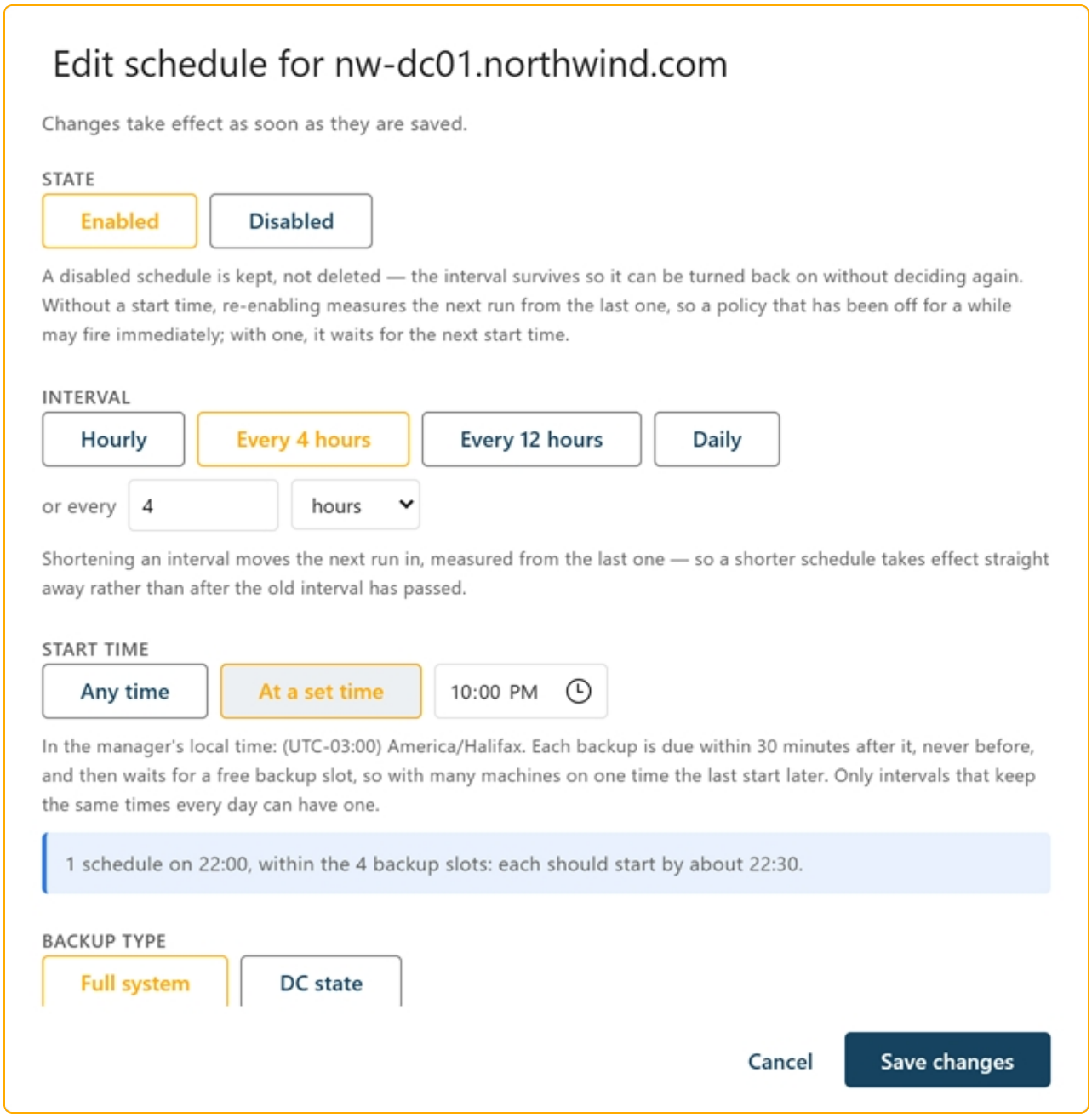
Field	Description
State	**Enabled** or **Disabled**. A disabled schedule keeps its settings so you can turn it back on later. When you re-enable a schedule without a start time, the next run is measured from the last one, so a schedule that has been off for a while may run immediately. A schedule with a start time waits for its next start time.
Interval	A preset (**Hourly**, **Every 4 hours**, **Every 12 hours**, **Daily**) or a custom value: **or every** a whole number of minutes, hours, or days, from 5 minutes to 90 days. Shortening an interval moves the next run in, measured from the last run.
Start time	**Any time** (the default) runs on the interval alone, with a small random spread either side of it. **At a set time** shows a time field, filled in with 22:00. The time is 24-hour, in the Manager's local time; the hint under the field names the Manager's time zone. Each backup is due within the **Start time spread** after the start time, never before, and then waits for a free backup slot. Only an interval that runs at the same times every day can have a start time; see Managing Backup Policies.
Backup type	**Full system** captures the critical volumes, including Windows, applications, and DC state, and supports both DC state recovery and full system recovery. **DC state** is smaller and faster and supports DC state recovery only. See Backup Types.

While an enabled schedule has a start time, the editor shows the Manager's estimate under the field. When the schedules on that time fit in the backup slots, it reads, for example, **3**

schedules on 22:00, within the 4 backup slots: each should start by about 22:30. When more schedules share the time than there are slots (**Maximum concurrent backups**), a warning says how late the last of them should start, based on the average length of recent backups, and suggests raising **Maximum concurrent backups** or giving some of them another start time. The warning does not stop the save.

Select **Create schedule** or **Save changes**. The confirmation says how the timing changed, for example **first backup is due immediately**, **first backup is due at** *time* for a new schedule with a start time, **next run moved from** *time* **to** *time*, **next run is now due at** *time* when an edit gives a schedule that has never run its first run, or **timing unchanged**. Times in the confirmation are shown in your browser's time zone. If the machine's agent was not reachable, the confirmation says the agent picks up the change when it reconnects. If no DC agent has ever come online for the machine, the confirmation says instead that no backup runs until one is installed and connects. It then names the first backup's due time, if that is still ahead, adding that the backup runs as soon as the agent connects if that is later; otherwise it says that the first backup then runs straight away.

The bulk editor (**Edit the schedules of** *n* **machines**) has the same fields, each with a **Keep current** option that leaves that setting unchanged. Machines without a schedule get the defaults for any field left on **Keep current**. A start time that some machines' intervals cannot keep is refused before the save, with the machines named. If some machines fail, the dialog lists them and offers to retry only those.



Related topics

- Managing Backup Policies
- How Backup Works
- Backup Types
- Coverage and Readiness
- Backup Jobs
- Domain Controllers

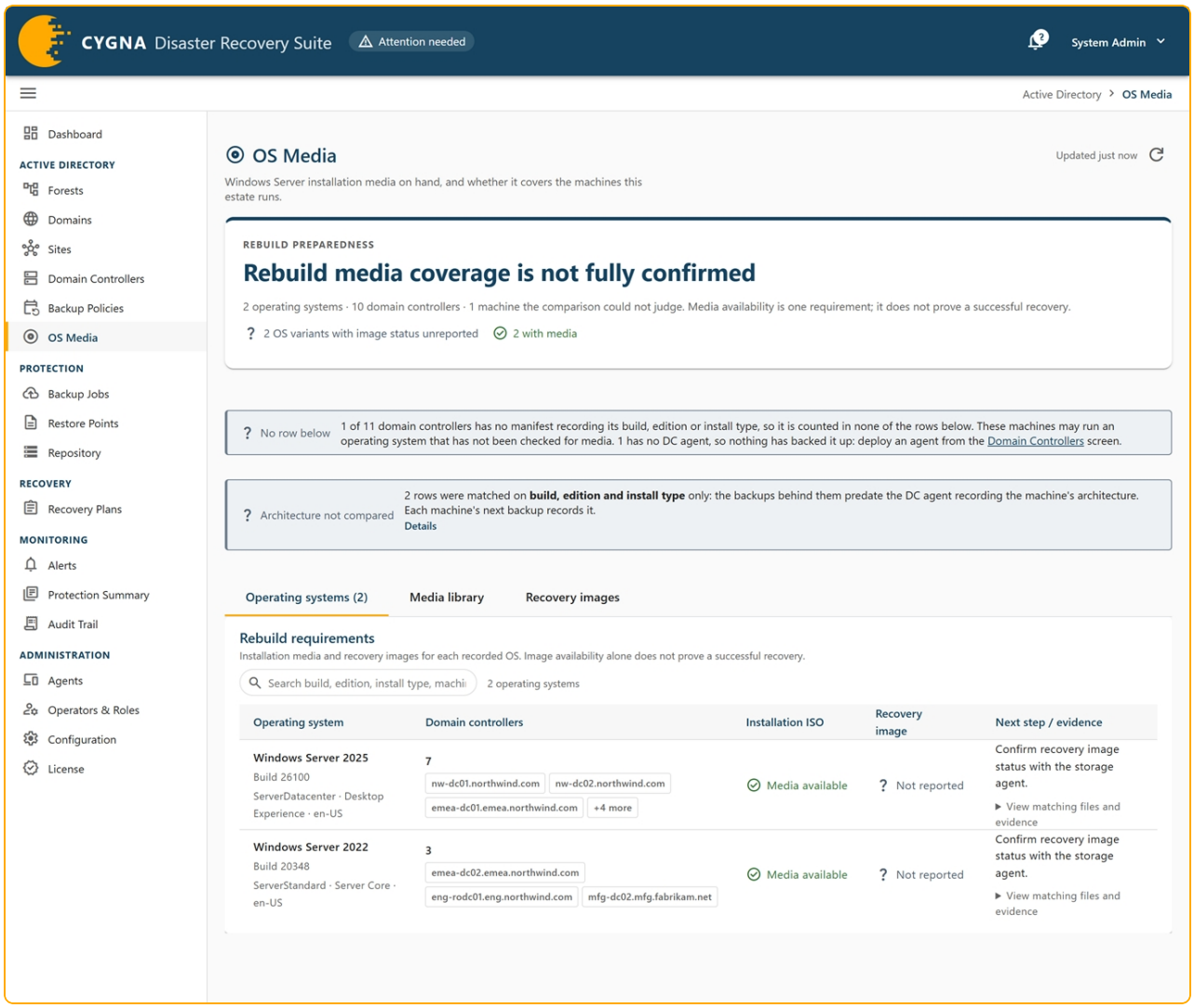
OS Media

The OS Media screen shows the Windows Server installation media you have on hand and whether it covers the operating systems your domain controllers run. It also shows, for each operating system, whether a domain controller running it can use full system recovery: restored as a whole server from a full system backup, in place or onto a replacement machine.

Both answers depend on the OS media library: a folder of Windows Server ISO files that the storage agent scans without mounting them. The storage agent also builds each operating system's recovery image from these ISOs. The library is optional for backups, but no domain controller can use full system recovery without a recovery image.

To open the screen, select **Active Directory** > **OS Media**. Every role can see it. The screen is read-only: you configure the media folder and recovery image folder for the storage agent on the Configuration screen. The screen refreshes every 60 seconds.

For setup, see Setting Up OS Media and Recovery Images. For how recovery images are used, see Recovery Images.



Notices

Notice	Meaning
Full system recovery unavailable	No storage agent has a recovery image folder set, so no recovery images are built and domain controllers running the listed operating systems cannot use full system recovery. DC state recovery is unaffected. Set **Recovery image folder** for the storage agent in Configuration and restart the agent. Use a folder separate from the media folder.
Optional feature, not configured	No storage agent has a media folder configured. This is not a gap in your protection; the operating systems table still lists what you would need to install to rebuild the estate.
Never scanned	A media folder is configured but no scan has completed yet,

Notice	Meaning
	so the library is still empty. The Media library tab shows which agent and any error.
No fleet OS collected	No backup has recorded a machine's build, edition, and install type yet, so the operating systems in use are not known. This fills in with the first completed backup.
No row below	Some domain controllers have no backup recording their build, edition, or install type, so they appear in no row. The notice counts them by cause: • No DC agent, so nothing has backed them up. Deploy an agent from the Domain Controllers screen. • An agent and no completed backup yet. A completed backup gives each a row. • Backed up, but the agent could not read the build, edition, or install type. Only domain controllers in scope are counted; the notice says how many excluded domain controllers are not counted.
Cannot be matched	Some rows have a build, edition, or install type that was only partly collected, so no media match can be attempted. The next backup fills it in.

Findings

The **Rebuild preparedness** panel (described below) lists media findings only when a media library is configured.

Finding	Severity	What to do
Operating systems with no installable media	Critical	A domain controller on one of these builds cannot be rebuilt from the media in the library. Add the matching Windows Server ISO to the media folder; build, edition, and install type must all match.
Operating systems covered only in another language	Warning	The media installs, but in a language your machines do not run, so tool output during recovery will not match the documentation. Add the ISO in your

Finding	Severity	What to do
		machines' language.
ISOs that could not be read	Warning	Check each file against its published hash and replace any that are truncated or corrupt. Files with no Windows image inside, such as driver discs, are not a fault.
Media folders could not be read	Warning	Check that the path exists and that the storage agent's service account can read it. The folder row shows the agent's error.

Rebuild preparedness

The **Rebuild preparedness** panel at the top of the screen opens with a headline that sums up whether the media on hand covers a rebuild. It counts operating systems by coverage and shows chips such as **with media**, **with no installable media**, **in the wrong language only**, and **new and uncovered** (operating systems that appeared recently with no media, as after an in-place upgrade or a new domain controller). Further chips count operating system variants by recovery image status. The panel then lists each finding.

Operating systems tab

The **Rebuild requirements** card has one row per operating system variant that your domain controllers run, taken from each machine's newest backup: the build, edition, install type, architecture, and language that a restore of that machine would put back.

Column	Description
Operating system	The product name with the build under it (**OS not collected** if neither was recorded), then the edition, install type (**Desktop Experience** or **Server Core**), architecture (**x64** or **ARM64**) when recorded, and language. **New** marks a variant first seen in a backup within the last 30 days.
Domain controllers	How many domain controllers run this operating system, with up to three names.
Installation ISO	Whether the library holds matching installation media. See the table below.
Recovery image	Whether a domain controller running this operating system can use

Column	Description
	full system recovery. See the table below. **Not reported** means the storage agent has not reported a recovery image status for this variant.
Next step / evidence	What to do next for this operating system. Expand **View matching files and evidence** to see the matching ISO files, the recovery image status, the update revision of the most-patched machine on this build against the best matching image, and the newest backup that reports this operating system. The revision is for information only. An ISO is a base image, so media behind the fleet is normal; the revision tells you how much patching a rebuild would need.

Installation ISO	Severity	Meaning
Media available	Good	The library holds an image matching the build, edition, install type, and language, and the architecture where the machines' backups record it.
Wrong language only	Warning	The library matches the build, edition, and install type, but not the language. It installs a working domain controller in the wrong display language.
No installable media	Critical	Nothing in the library matches, so a machine on this operating system could not be rebuilt from the media on hand.
Machine OS unconfirmed	Unknown	The machines' build, edition, or install type was never collected, so no match can be attempted.
Library not scanned	Unknown	A media folder is configured and no scan has completed yet.
No media library configured	Information	No storage agent has a media folder. The library is optional, so this is not a gap.

Recovery image	Severity	Meaning
Available, image proven	Good	A recovery image exists for this operating system, and at least one has been proven by a completed restore.
Available, image untested	Warning	A recovery image exists and is served, but none has been proven by a restore yet. This is the normal state until a full system recovery or a test restore completes with it. With an Isolated Recovery Host and the **Prove new recovery images** setting on, untested images are test-restored automatically.
Unavailable, no image built	Critical	There is no recovery image yet. The storage agent builds one on its next pass.
Unavailable, image failed its test	Critical	Every recovery image for this operating system failed its boot test, so none is served. Rebuilding produces the same result; investigate the failure.
Unavailable, no media to build from	Critical	No media in the library matches, so no image can be built. Add the matching ISO.
Unavailable, no image folder set	Warning	No storage agent has a recovery image folder set, so no image will be built. Set **Recovery image folder** in Configuration and restart the storage agent.
No restore point to recover	Information	No restore point exists for this operating system, so no recovery image is needed.
No storage agent has reported	Unknown	No storage agent has reported a recovery image library yet.

In every unavailable case, DC state recovery of a domain controller that still starts Windows is unaffected.

Media library tab

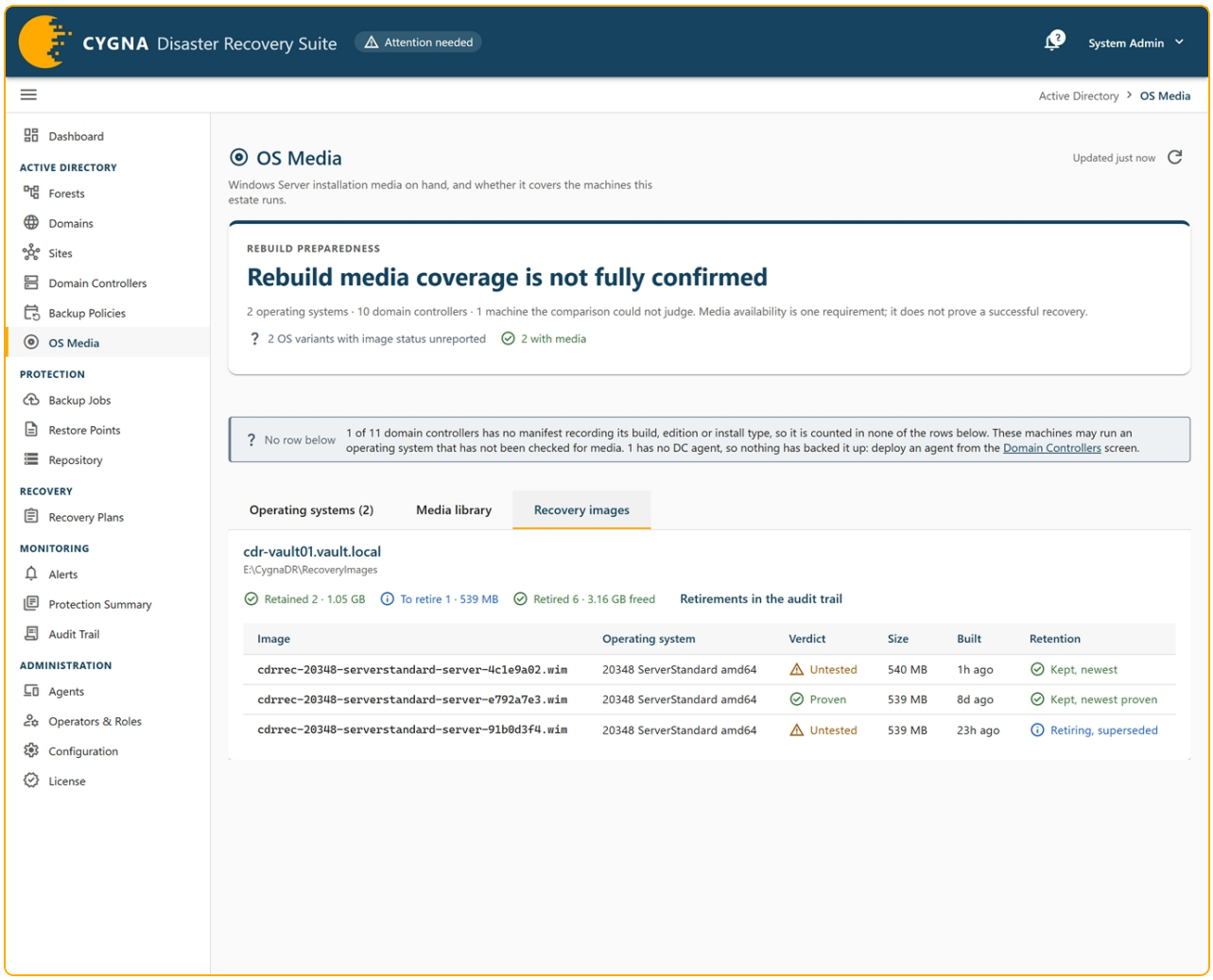
The **Media library** tab shows what the storage agents found in their media folders. Nothing here is required for backups to work.

- **Media folders**–One row per storage agent: the folder and whether it is a **Local folder** or **Network share**, the last scan and its outcome (**Read successfully**, **Partly read**, **Folder unreachable**, **Access denied**, or **No folder configured**), the number of ISO files and images, and whether the agent is **Connected** or **Offline**. A share depends on a network, a server, and a credential the same disaster may have taken, and a local folder is only as available as the storage machine.
- **ISO files**–Each file, whether it was read (**Read**, **No Windows image inside**, **Could not be read**, **Unsupported format**, or **Not read yet**), the images inside, how many of the operating systems in use it covers, its size and SHA-256 hash when computed, when it was modified, and when a scan last listed it. Compare the hash with Microsoft's published hash to confirm the file.
- **Images inside the media**–Each Windows image in the files: index and name, build, edition, install type, architecture, language, revision, and installed size.

A scan that cannot read the folder removes nothing from the inventory, so a failed scan never makes media look missing.

Recovery images tab

The **Recovery images** tab shows what each storage agent's recovery image library holds, which images it keeps, and which it retires. Each product build adds a new image for every operating system variant, so the Manager decides what each library keeps and tells the storage agent to delete the rest. For the rule, see Recovery Images.



There is one card per library, titled with the storage agent and the library's folder. Chips at the top of the card:

- **Retained**–How many images are kept, and their size. Per operating system variant the vault needs, these are the newest image and the newest proven one, and any image a restore is using.
- **To retire**–Images marked for retirement and their size. The storage agent is told straight away while it is connected, otherwise when it reconnects, and the images leave the list after its next report. Shown only when there are any.
- **Retired**–How many images this library has deleted on the Manager's instruction, and the space that freed.
- **Nothing retired**–The Manager is not deciding retirement for this library at the moment, and every image counts as retained. Hover to see why, for example when the library's last report did not succeed and the Manager's view of it may be stale.

The image table lists every image in the library:

Column	Description
Image	The image file name.
Operating system	The build, edition, and architecture the image serves.
Verdict	**Proven** (a completed restore, including a test restore, has used it), **Untested**, or **Failed**.
Size	The size of the image file.
Built	How long ago the storage agent built it.
Retention	What the Manager decided for the image. See the table below. **Not decided** means no decision has been made for it.

Retention	Meaning
Kept, newest	The newest image for its operating system variant, whatever its verdict.
Kept, newest proven	The newest image for its variant that a restore has proven.
Kept, newest usable	The newest image for its variant has failed, so the newest image that has not failed is kept as well.
Kept, a restore is using it	A restore that has not ended is using the image. An image in use is never retired.
Retiring, superseded	A newer image serves the same variant.
Retiring, failed and superseded	The image failed, and a newer one exists.
Retiring, no restore point needs it	No restore point in retention runs this operating system variant.

Images the library has already deleted are counted, not listed. When there are any, **Retirements in the audit trail** opens the **Audit Trail** screen filtered to `recovery-image.retired`, where each one is recorded.

The tab can also show:

- **No storage agent has reported a recovery image library.**—No storage agent with a recovery image folder has reported yet.
- **This library holds no recovery images.**—The library is reported but empty.

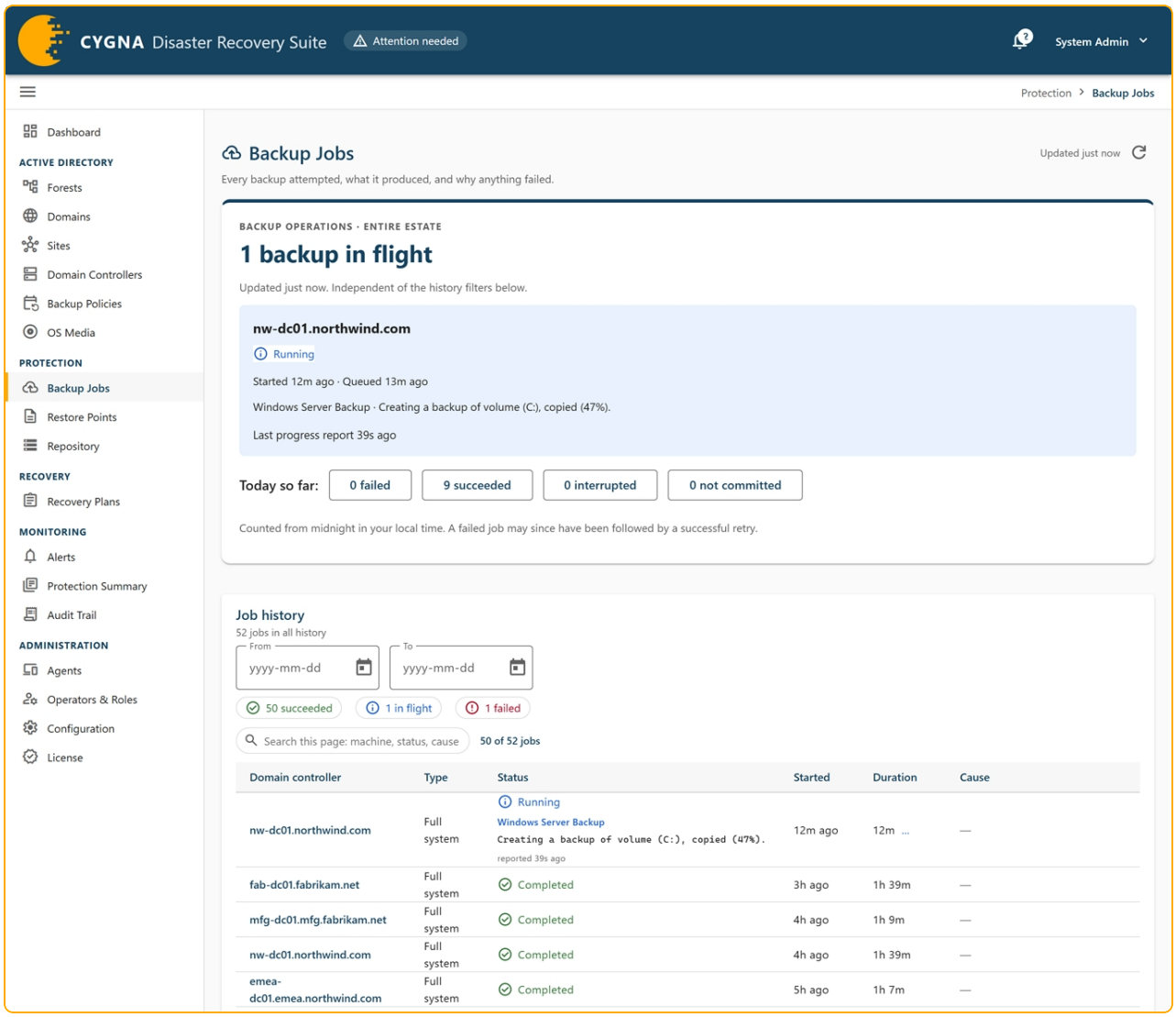
Related topics

- Setting Up OS Media and Recovery Images
- Recovery Images
- Restore Methods
- Test Restores
- Full System Recovery in Place
- Full System Recovery onto a Candidate
- Settings Reference

Backup Jobs

The **Backup Jobs** screen lists every backup the Manager has attempted, what each one produced, and why any of them failed. Use it to follow a backup while it runs, to check that scheduled backups are succeeding, and to find the cause of a failure.

To open it, select **Protection** > **Backup Jobs**. The screen requires the `backup.view` capability, which every role includes, and refreshes itself every few seconds. Operators with the `backup.run` capability (Backup Operator role) can also retry a job that ended without a restore point.



Backup operations

The card at the top of the screen covers the entire estate and ignores the history filters below it. It lists each backup in flight with its machine, status, when it was queued and started, the last phase the agent reported, and when that report arrived.

After **Today so far:**, buttons count the jobs created since midnight in your browser's time zone that **failed**, **succeeded**, were **interrupted**, or were **not committed**. Select one to filter the job history to those jobs. A failure today may already have been followed by a successful retry.

Job history

The **Job history** card holds the list of jobs. Its subtitle gives the number of jobs in all history, or the number that match your filters.

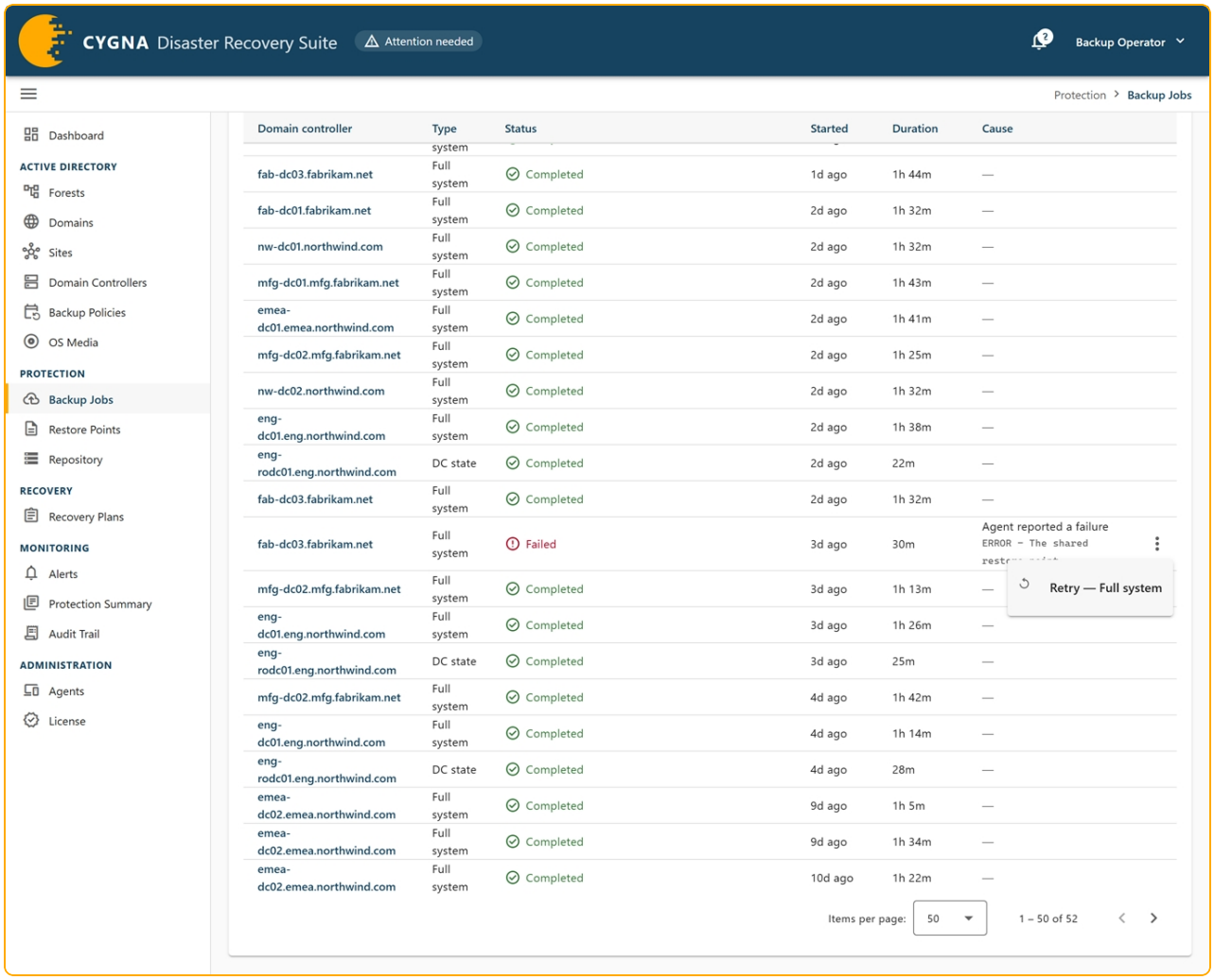
Filters

- **From and To**–Limit the list to jobs created in a date range (your browser's time zone). Select **Clear filter** to remove the range.
- **Status chips**–Select a chip to list only jobs in that group: **succeeded**, **in flight**, **failed**, **interrupted**, **not committed**, **in other states**, or **unreconciled**. The chip counts do not change when you filter, so they always show how many jobs are in each group.
- **Machine**–When you open the screen from a link on another screen, for example from a domain controller, the list shows only that machine's jobs. A notice says so; select **Clear filter** to see every machine.
- **Search**–Searches only the jobs on the current page, by machine, status, cause, or error text. To find a job on another page, narrow the list with the dates, a machine, or a status first.

Columns

Column	Description
Domain controller	The machine that was backed up. Select it to open the job's detail sheet.
Type	**Full system** or **DC state**. For the difference, see Backup Types.
Status	The job's status (see below). While a job is in flight, a line under the status shows the last phase the DC agent reported and how long ago it reported it, or **nothing reported yet**.
Started	When the job started. A backup still waiting for a backup slot reads **Waiting for a slot**; hover for when it was queued. A dash means the job never started.
Duration	How long the job ran. While it is still running, this is the time elapsed so far.
Cause	For a job that did not succeed, the error and the text the agent or tool reported. For a job that completed but was not committed, the staging path where its content is waiting.

With `backup.run`, a row whose job is **Failed**, **Interrupted**, or **Cancelled** has an actions menu (⋮) with **Retry – *type***. It queues a new backup of the same machine and type, as **Back up now** on the **Domain Controllers** screen does, and the new job appears in the list at once. If the Manager does not start it, for example because the agent is not connected or a backup is already running, a message gives the reason. A job that completed but was not committed has no retry, because its backup already exists in staging.



The list is paged, 50 jobs per page by default; you can choose 25, 50, 100, or 200.

Job statuses

Status	Meaning
Queued	Waiting for a backup slot. The Manager runs only as many backups at once as **Maximum concurrent backups** allows and starts this one when a slot frees. Nothing has been asked of the domain controller yet. A queued backup is kept across a Manager restart. While a recovery is working on the machine, or while the Manager has no active license, it stays queued until it can run.
Created	The backup has a slot and the Manager is starting it. The DC agent has not started it yet.

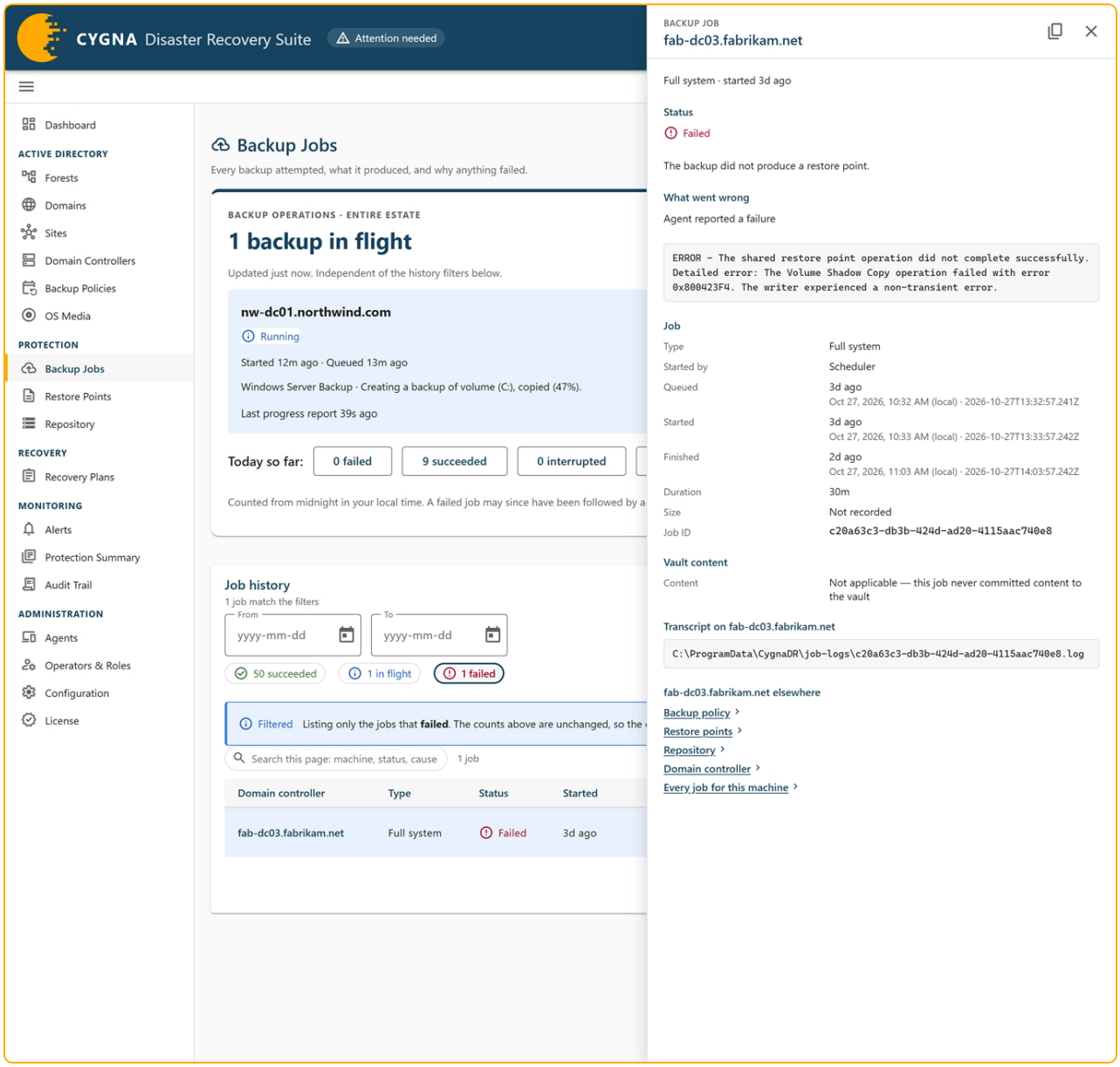
Status	Meaning
Staging	The staging share the backup will be written to is being prepared.
Running	The backup is running on the domain controller.
Committing	The backup finished on the domain controller and is being moved into the vault.
Completed	The backup finished and was committed to the vault. It is a restore point unless retention has since removed its content.
Completed (unreconciled)	The backup completed, but the Manager could not confirm it against the vault. Treat it as uncertain, not as a restore point, until this is resolved.
Completed, not committed	The domain controller took the backup, but the Manager did not see it reach the vault. The Manager checks the storage machine for these every hour and whenever the storage agent connects. Content still in staging is committed, content already in the vault is recorded as a restore point, and the job changes to **Completed** either way. If the content is in neither place any more, the job stays as the record that the backup was taken, its detail sheet says so, and it is not a restore point. The **Repository** screen lists these backups and can check now.
Failed	The backup did not produce a restore point.
Interrupted	The job stopped before it finished, so it did not produce a restore point.
Cancelled	An operator cancelled the job. It did not produce a restore point. A cancelled job is not counted as a failed attempt.

Job detail sheet

Select a job's domain controller to open its detail sheet. The sheet has these sections:

- **Status**—The status and a sentence explaining what it means for the backup.
- **What went wrong**—For a job that did not succeed, the cause and the raw error text, or **No error text was recorded**.

- **Content to rescue**—Shown when the backup content is in staging rather than the vault, with its staging path. The Manager commits it on its next check of the storage machine. Not shown once the storage agent has reported the content gone.
- **Last reported**—The **Phase** the DC agent last reported, **When** it reported it, and **Volume progress** as a percentage. Volume progress covers only the current volume and restarts for each volume.
- **Job**—**Type**, **Started by** (**Scheduler** or **Operator**), **Queued**, **Started**, **Finished**, **Duration**, **Size**, and **Job ID**.
- **Vault content**—**In vault**, or **Pruned** with the retention rule that removed it and when. A job that never committed content shows that the section does not apply.
- **Transcript on <machine>**—The path of the wbadmin transcript on the domain controller. The full transcript stays on the machine; read it there when you need the tool's own output.
- **<machine> elsewhere**—Links to the same machine on other screens, including **Every job for this machine**.



Why a job fails

The Manager records an error code on each job that does not succeed, so failures can be counted and grouped. The **Cause** column and the detail sheet show its label. Common causes include:

- **DC agent unreachable**–The Manager declined to start the backup because the domain controller's agent was not connected.
- **Storage agent unreachable**–The storage agent was not available to prepare staging or commit the backup.

- **Insufficient staging capacity** or **Staging account could not be created**–The storage machine could not reserve room or create the per-job account the domain controller writes with.
- **Agent reported a failure**–The backup ran and failed on the domain controller. The error text shows what wbadmin reported.
- **Backup not reported in time**–The Manager stopped waiting for the result. The backup may still be running and can still complete when the agent reconnects.
- **Vault commit failed** or **Vault commit not confirmed in time**–The backup finished but did not reach the vault, or the commit was not confirmed before the timeout.
- **Cancelled by an operator**–Someone stopped the backup.

Two codes can appear on a **Completed** job. **Backup version identifier not captured** means the backup is in the vault but the identifier a restore is addressed by was not recorded. **Vault commit confirmed late** means the restore point exists, but the commit was confirmed after the Manager stopped waiting for it. The error says why; raise the **Vault commit timeout** setting only when it says the commit outlasted that setting.

For every error code, what it means, and what to do, see Backup Job Errors.

Cancelling a backup

The **Backup Jobs** screen has no cancel action. To stop a running or queued backup, use **Stop backup…** on the Domain Controllers screen, which requires the `backup.run` capability. The agent stops a running backup cleanly, and the run produces no restore point. A queued backup is cancelled at once, because it never started. For steps, see Monitoring and Cancelling Backup Jobs.

Related topics

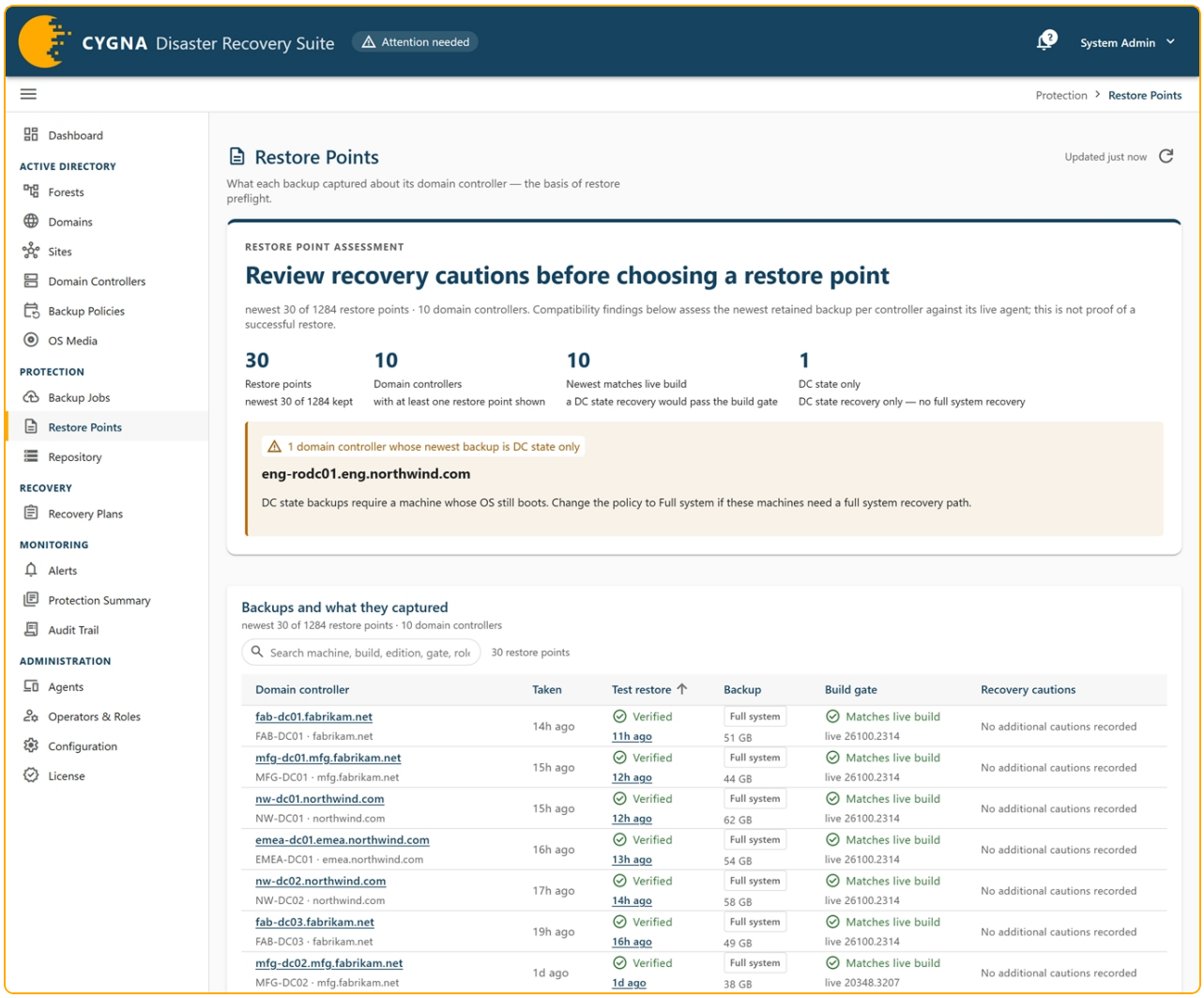
- How Backup Works
- Running a Backup Now
- Restore Points Screen
- Settings Reference

Restore Points Screen

The **Restore Points** screen shows what each backup captured about its domain controller: its operating system build, hardware, network settings, Active Directory roles, and backup details. These facts are what restore preflight checks a restore against, so this screen tells you in advance whether a backup could be restored and what might stand in the way.

To open it, select **Protection** > **Restore Points**. The screen requires the `backup.view` capability, which every role includes. It is read-only: you cannot start a restore from here. To restore from a restore point, or to test-restore it, create a recovery on the Recovery Plans screen. Test restore results are not shown on this screen; the restore point list of a new single-DC restore shows them.

For what a restore point is and what it records, see Restore Points.



Restore point assessment

At the top of the screen, the **Restore point assessment** panel opens with a headline and summarizes problems with each machine's newest restore point, which is the one a restore would start from:

- Domain controllers whose newest backup no longer matches the build the machine runs now. A DC state recovery of that backup would be refused; take a fresh backup.

- Newest restore points that report SYSVOL replicated by FRS. Restore preflight refuses FRS; migrate SYSVOL to DFSR, then take a fresh backup.
- Domain controllers whose newest backup holds an operations master role that a later backup lists on another DC. Confirm that the role was transferred, not seized: a DC whose roles were seized elsewhere after its backup must be rebuilt, never restored.
- Domain controllers whose newest backup is DC state only, so they have no full system recovery option.

Compact metrics in the same panel count the restore points kept, the domain controllers they cover, and how many newest restore points fall into each of the cases above. Metrics for a case that does not occur are hidden.

Backups and what they captured

This card lists the newest 200 restore points, one row per backup. Each machine's newest restore point is tagged **Newest**; older rows are shown dimmed. Use the search box to find rows by machine, build, edition, gate, role, test restore result, or hash. When you open the screen from another screen's machine link, every count and finding describes that machine alone; select **Clear filter** to see the whole fleet.

Backups that retention has pruned are not restore points, because their content is gone, so the card hides them and says how many are hidden. Select **Show them as history** to list them, each tagged **Pruned · not restorable**; select **Hide them** to hide them again. When every backup in view has been pruned, the card reads **No restore points: every backup in view has been pruned.** and offers **Show the pruned backups as history**. A restore point taken while a recovery of its forest was under way is tagged **During recovery**; hover for the plan and the step it had reached. Such a backup holds the directory part-way through that recovery, so restore it only if that is the state you mean.

Column	Description
Domain controller	The machine, with its NetBIOS name and domain. Select the name to open the restore point's detail page.
Taken	How long ago the backup was taken. Hover for the local and UTC time.
Test restore	The result of this version's latest test restore: **Verified**, **Failed**, **Running**, **Waiting to start**, or **Inconclusive** (the test restore ended without testing this version). Hover for what it means and the host it ran on. Below it, the time links to the test restore's page. A version never tested reads **Not tested**. Sorting by this column puts failed versions first and untested ones last. See Test Restores.
Backup	**Full system** (supports DC state recovery and full system recovery)

Column	Description
	or **DC state** (DC state recovery only), with the size.
Build gate	How the backup's build compares with the build the machine's DC agent reports now (see below).
Recovery cautions	What could stand in the way of restoring this backup: **Unsupported SYSVOL** (SYSVOL replicated by FRS, which recovery does not support), **SYSVOL not collected**, **Role holder changed** (a later backup of another DC lists one of this DC's operations master roles), and **Roles not collected**. An **RODC · not a first-DC candidate** tag marks a read-only DC, which is rebuilt rather than restored. Otherwise the column reads **No additional cautions recorded**.

Select a row, or press Enter or Space on a focused row, to open the **Captured backup** sheet with what the backup recorded: **Captured**, **OS at backup** (product name, build and update build revision, edition, and install type), **Test restore** (the latest result, when, what it means, and **Open the run**), **Build comparison**, **SYSVOL**, **Operations master roles at backup**, **Role changes** (when a role has since been listed elsewhere), **Machine shape** (platform, firmware type, disks, and memory), **Manifest schema**, and **SHA-256** of the stored restore point document. Select **Open full captured manifest** to open the restore point's detail page.

Build gate values

Value	Meaning
Matches live build	A DC state recovery of this backup onto the machine as it runs now would pass the build check.
Live build differs	A different Windows release is installed now. This backup cannot be restored onto that machine; rebuild it instead.
Live revision differs	Same build, different revision—usually Windows Update since the backup. A DC state recovery is refused until you take a fresh backup.
Build not captured	The backup recorded no build or revision, so nothing can be compared.
Live build not reported	The agent running under this name has not reported its build.
No live target	No DC agent is enrolled under this machine's name.

On older rows a difference is expected, because the machine has been updated since; only the newest restore point's gate is a finding.

Operations master holders over time

Expand **Operations master history · investigate role changes** at the bottom of the screen to show this card. It shows, for each operations master role, which backed-up domain controller held it and over which backups. Only backups provide evidence here, so a role move is known to have happened somewhere between one holder's last backup and the next holder's first. The **Domains** screen shows the directory's own account of role holders. When a role has no current holder attested by a backup, a **Role history incomplete** notice above the section says how many.

Columns: **Role**, **Of** (the domain or forest), **Held by**, **First attested** and **Last attested** (each with a link to the restore point), **Backups** (how many consecutive restore points listed the role), **Contradicted** (when a later restore point listed the role on another DC, or the holder stopped listing it), and **Standing**. **Latest record** means this is the newest evidence for the role and nothing contradicts it. **No current holder attested** means the current holder has not been backed up, or the role is unassigned.

Restore point detail

Select a domain controller in the list to open the detail page for that restore point. Select **All restore points** to return to the list, scoped to the same machine.

The **This restore point** card shows the SHA-256 of the stored document, when it was written, the backup job, the backup type, size, and agent version, the latest **Test restore** result with a link to its run (or **Not tested**), the **wbadmin version identifier** a restore is addressed by, and links to the same machine on other screens.

Notices

A notice above the tabs explains anything that limits what the page can show:

- **Backup pruned**—Retention deleted this backup's content. The restore point is kept as a record, but nothing is left to restore from it.
- **Hash changed**—The document changed between two requests. Refresh before trusting either.
- **Unreadable**—The stored document could not be parsed. The bytes are on the **Raw** tab.
- **Preflight unavailable**, **Not in the index**, or **No machine named**—The preflight comparison cannot be built. The captured facts and raw document are unaffected. **Not in the index** usually means the restore point is older than the newest ones the Manager returns for a machine.

Tabs

- **Preflight**–Compares the restore point with the machine as it runs now: **OS build and revision**, **Disk count and sizes**, **SYSVOL replication**, **Recovery mode** (**DC state and Full system** or **DC state only**), **Writeable DC**, and **Operations master roles**. Each check shows a verdict and the reason for it. Disks always show **Cannot be evaluated from here**, because the DC agent does not report the live machine's disks; the target machine's disks are checked when you plan a full system recovery.
- **Captured facts**–Everything the backup recorded, grouped as **Identity**, **Operating system**, **Active Directory role**, **Backup**, **Hardware** (including disks, and volumes with their BitLocker status), **Network** (adapters, addresses, DNS servers, and persistent routes), and **Roles, features and programs**. A value that was not collected shows **Not collected**.

  Roles, features and programs lists the Windows roles and features installed at backup time (name, feature name, and type) and the installed programs (name, version, and publisher). A rebuild of the domain controller from this backup reinstalls its roles and features before it promotes the machine; it does not bring back the roles' configuration or the programs.
- **Raw**–The stored document exactly as kept, with its SHA-256 and a **Copy** button.

Note: An **Encrypted** BitLocker status on a volume means a full system recovery of that volume needs extra steps. See Recovery Requirements.

Related topics

- Restore Points
- Preflight Check Reference
- Backup Jobs
- Repository

Repository

The **Repository** screen shows the state of backup storage: how much room the vault has left, how far back each machine's history goes, what retention has removed, and whether stored backups still match the hashes recorded when they were committed.

To open it, select **Protection** > **Repository**. The screen requires the `backup.view` capability, which every role includes. Apart from the requests described below, it is read-only. Capacity and retention are measured and enforced on the storage machine; this screen reports what the storage agent tells the Manager. To change retention or verification settings, use the **Storage agents** tab of the Configuration screen.

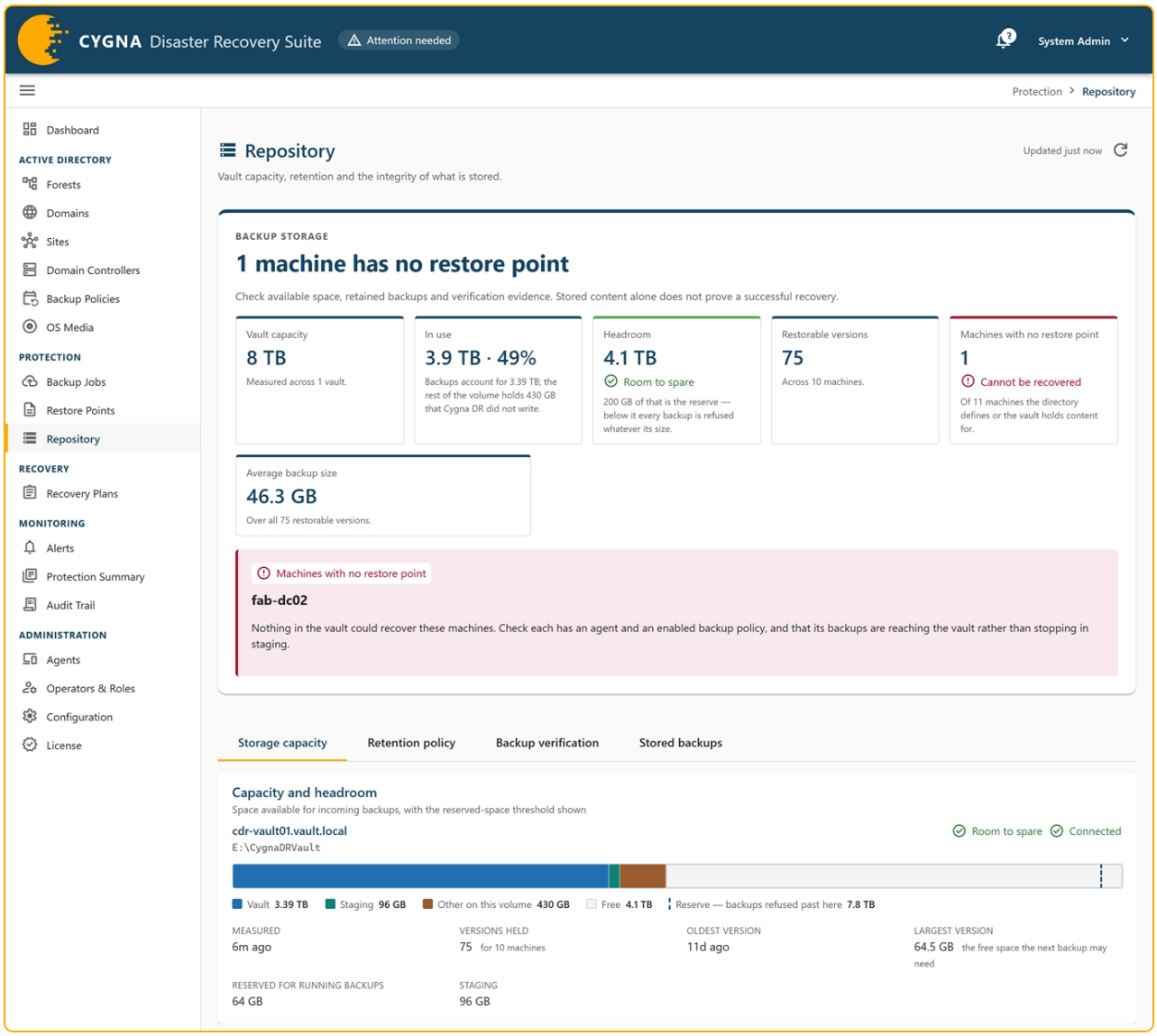
The capacity half of the screen appears when the storage machine enrolls and measures its volume. The history half appears with the first completed backup. The screen reads the storage machine alone; a candidate storage agent, or one that was replaced, is not counted.

Requests waiting for a second operator

When an operator who holds only one of the two capabilities it needs asks to replace the storage machine, the request waits here for an operator holding the other, usually a Backup Operator. The **Requests waiting for a second operator** card at the top of the screen names each pending request, who asked and when, when it lapses, the capability approving it needs, the asker's note, and how many restore points would have ended when it was asked. The buttons shown depend on who you are:

- **Review and approve…**—For an operator holding the missing capability who did not ask. It opens the replacement dialog as things stand now, which can differ from when it was asked, and asks for the candidate's agent ID before **Approve and replace the storage machine** carries it out.
- **Decline…**—For an operator who could approve it, with an optional reason.
- **Cancel request…**—For the operator who asked, or anyone holding one of the two capabilities.

Requests decided in the last week are listed under **Decided in the last week**, with who decided them. With no request pending or recently decided, the card is not shown. See Replacing the storage machine.



Backup storage panel and notices

The **Backup storage** panel at the top of the screen opens with a headline that sums up the state of storage, holds the tiles described below, and lists a finding when storage threatens recovery:

Finding	What to do
Vault is refusing backups (critical)	Free space is at or below the reserve, so every backup is refused. Add capacity, or lower the retention ceilings so the sweep can reclaim room. The retention floor is never crossed to make space.
No room for another	Usable space is below the size of the largest version held, so

Finding	What to do
version (warning)	the next backup of the biggest machine will not fit. Add capacity or lower the retention ceilings before backups are refused.
Vault volume could not be read (unknown)	The storage agent could not reach its storage root. Check that the volume is mounted. Capacity is unknown, not zero.
Abandoned staging is holding space (warning)	Staging folders from jobs that never committed are taking space. With **Delete abandoned staging folders** on, the default, they are deleted on the next sweep; a finished backup that was never committed is committed by the Manager before then. With it off, they are reported and kept; inspect them on the storage machine.
Machines with no restore point (critical)	Nothing in the vault could recover these machines. Check that each has an agent and an enabled backup policy, and that its backups reach the vault.
A backup was refused rather than history deleted (critical)	The retention floor stopped a prune short of the space a backup needed. This is the floor working as designed. Backups are refused until you add capacity or lower the ceilings.
Backups completed but never reached the vault (warning)	A domain controller finished these backups, but the Manager did not see them reach the vault, so they are not restore points. The Manager checks the storage machine for these every hour and whenever the storage agent connects. Content still in staging is committed, content already in the vault is recorded as a restore point, and content that is gone stops being counted.

When such backups exist, **Backups not in the vault** lists each one with its **Machine** and job, when it was **Taken**, **Why not committed**, the **Last check**, and what was **Found**: **In staging, being committed**, **In staging, commit failed**, **Could not be confirmed**, **No storage agent connected**, **Storage agent did not answer**, or **Not checked yet**. Backups the check resolved are not listed. With `backup.run`, select **Check now** to check at once instead of waiting for the hourly check.

Two notices can also appear. **Retention is in dry run** means prunes are selected and reported but nothing is deleted, so versions named in the prune history are still restorable. **No storage agent** means there is nowhere to commit backups; install the storage agent (see Setting Up the Storage Machine).

Tiles

- **Vault capacity**—The total measured across all vaults.
- **In use**—Space used, and how much of it is backups versus other data on the volume.
- **Headroom**—Free space, with the capacity verdict and how much of it is the reserve.
- **Restorable versions**—Versions held across all machines, and how many machines they cover.
- **Machines with no restore point**—Machines the directory defines, or the vault holds content for, that have nothing to restore from. Machines excluded by an operator are not counted.
- **Average backup size**—Over the restorable versions that have a recorded size.

Storage capacity tab

The **Capacity and headroom** card shows each vault as the storage machine measures it, with its storage root, a capacity verdict, and the storage agent's connection state. A bar divides the volume into **Vault**, **Staging**, **Other on this volume**, and **Free**, with a line marking the reserve below which backups are refused.

Capacity verdict	Meaning
Room to spare	The vault can take more versions.
No room for another version	The next backup of the largest machine will not fit.
Refusing backups	Free space is at or below the reserve.
Volume unreadable	The storage agent could not read the volume.
Never measured	The storage agent has not reported a measurement yet.

The facts under each vault are **Measured**, **Versions held**, **Oldest version**, **Largest version** (the free space the next backup may need, and the threshold for **No room for another version**), **Reserved for running backups**, and **Staging**. An **Abandoned staging** notice lists idle staging folders whose jobs never committed; an **Unmeasured versions** notice lists versions that could not be sized at the last measurement.

Retention policy tab

The **Retention policy** card shows the retention policy each storage agent is enforcing: **Age floor**, **Minimum per machine**, **Delete beyond age**, **Delete beyond count**, **Reserve**, **Sweep every**, **Abandoned staging**, and **Mode** (**Enforcing** or **Dry run**). The age floor and

minimum per machine are set on the storage machine itself; nothing the Manager sends can weaken them. A **Ceilings decide the real depth** notice appears when a version count ceiling means history is shallower than the floor suggests.

Expand **Cleanup history · why an older backup was removed** to see the **Retention activity** card. It lists the prunes the storage agent has reported: **When**, **Outcome**, **Vault**, **Trigger** (**Scheduled sweep** or **Space pressure**), **Rules**, **Versions**, **Reclaimed**, **Free after**, and **Machines**. A **Backup refused** chip marks a prune the floor stopped, and a **Retention was blocked** notice above the section names the storage agent, when, and why. Rows marked as a preview (from dry run) deleted nothing.

For how retention decides what to keep, see Retention.

Backup verification tab

The **Integrity of what is stored** card shows how many restorable versions carry a content hash, and how many have been re-read since commit and still match it. Notices explain any gap:

- **No results yet**–Scheduled verification is running and has not checked a version yet. This is normal on a new installation.
- **Not yet checked**–Versions a verification pass has not reached yet.
- **Could not be checked**–Versions the check could not run on, for example because no hash was recorded or a file would not open. This is not a finding about the content.
- **Verification failed** (critical)–Versions that no longer hash to the value recorded at commit. Each one also raises the alert **Vault content whose hash does not match**, whose **View repository** link opens this screen. Nothing is deleted or repaired. Check the storage volume for faults and the vault for unauthorized writes.
- **No hash recorded**–Versions committed without a content hash. They can never be verified and remain a gap until they age out.
- **Unreadable on disk**–Versions that could not even be sized at the last measurement.

A failures table lists each failing version, the expected and read hash, how long it has been failing, and what the storage agent reported. For how verification works, see Vault Protection and Integrity.

Verification finds changes to content at its next pass. Changes to committed backups that retention did not make, such as a deletion, are detected as they happen and raise **Committed vault content changed outside retention** on the Alerts screen instead; `cdr repository changes` lists them with their paths and who made them. See Detecting changes to committed backups.

Stored backups tab

The **Stored backups by machine** card shows what the Manager's own job history says is restorable for each machine. Filter chips select machines that are **recoverable**, have content stranded in staging, have no restore point, are excluded, or are no longer in the directory.

Columns: **Machine**, **Recoverable**, **Versions**, **Total**, **Storage details**, **Oldest**, **Newest**, and **Hashed**. Expand **Details** in the **Storage details** column for the machine's average and largest version size and how many versions retention removed. Each row's actions menu (⋮), at the right edge of the row, opens the same machine on the other backup screens, filtered to it.

Domain controllers that have been removed from the directory and have nothing in the vault are not listed; a line under the table says how many. The **Domain Controllers** screen lists them under its **removed** chip.

Below the table, a reconciliation line compares the number of restorable versions in the Manager's records with the number the storage agents counted on disk. The two are measured independently, so a disagreement means something has diverged and is worth investigating.

Related topics

- Sizing Backup Storage
- Settings Reference
- Alerts Reference

Recovery Plans

The **Recovery Plans** screen is where every recovery starts and where you follow it to the end. It lists every restore and recovery the Manager holds, and has a page for creating each kind of recovery and a page for each one you create.

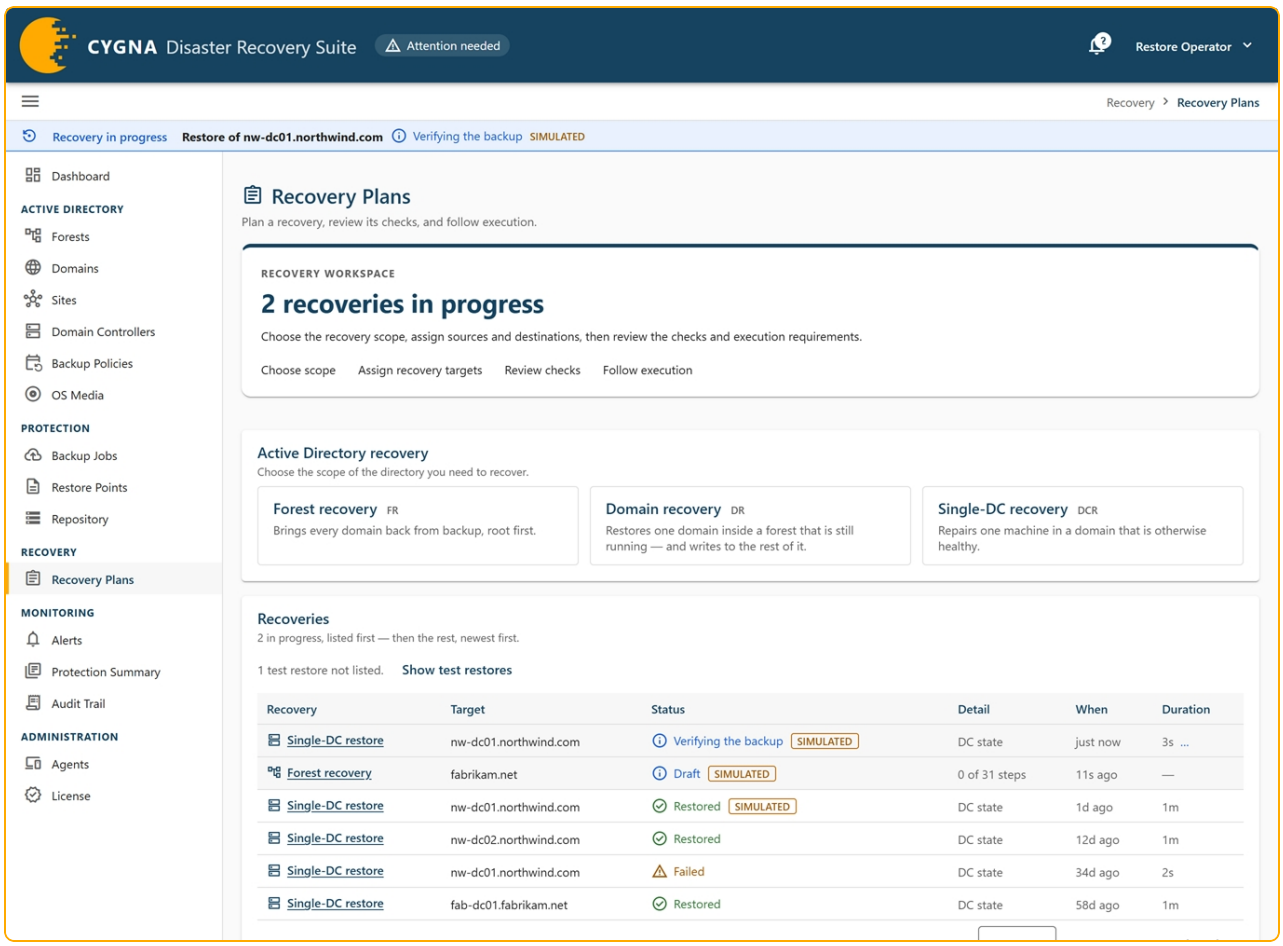
To open it, select **Recovery** > **Recovery Plans**. Viewing the screen and its pages requires the `plan.view` capability, which every role includes. Changing anything requires one of two further capabilities, held by the Restore Operator role and the break-glass account:

- **`plan.approve`**—Draft restores and recovery plans, approve them, withdraw an approval, ask a target machine to describe itself again, accept a restore's preflight checks that need acceptance, ask a restored machine to validate again, and discard a plan that was never executed.
- **`plan.execute`**—Execute a restore or plan, stop or abandon a restore, cancel a restore a plan holds, acknowledge a point of no return, attest or reject an operator step, retry a halted step, shorten a wait, supply domain credentials and the DSRM password, set a

plan's notification addresses, email its report, take back an unattended plan, and abort a plan.

No other role implies `plan.execute`. The System Admin role, which can configure the Manager, cannot run a restore. Where you lack a capability, the page says which roles hold it instead of showing the button, except where noted below.

For how a recovery moves from plan to completion, see Recovery Lifecycle. For step-by-step instructions, see Recovery Procedures.



Recovery Plans landing page

Active Directory recovery

This card has one tile for each recovery scope. Each scope is a different procedure:

- **Forest recovery**–Brings every domain back from backup, root domain first.
- **Domain recovery**–Restores one domain inside a forest that is still running, and makes changes in the rest of the forest to reconcile it.

- **Single-DC recovery**—Repairs one domain controller in a domain that is otherwise healthy.

To decide which to use, see Choosing a Recovery Approach.

Recoveries

This card lists every recovery on this Manager, a page at a time. Recoveries in progress are listed first, then the rest, newest first. Select a row to open its page. Use the paginator at the bottom of the card to change the page or the number of rows per page, up to 200.

When test restores have run, a line above the list says how many there are and whether they are listed. Test restores are left out until you select **Show test restores**; select **Hide test restores** to leave them out again. The choice is remembered in your browser. The dashboard's **View all test restores** link and the **View test restores** link of a test restore alert open the list with test restores shown.

Column	Description
Recovery	**Single-DC restore**, **Test restore**, **Forest recovery**, or **Domain recovery**.
Target	The domain controller, domain, or forest being recovered.
Status	The restore run status or plan status (see below).
Detail	For a restore run, **DC state** or **Full system**. For a plan, how many of its steps are done.
When	When the recovery was planned, started, or finished.
Duration	How long it took from execution to its end, including any time halted. A recovery in progress is measured to now and marked with an ellipsis. A dash means it was never executed.

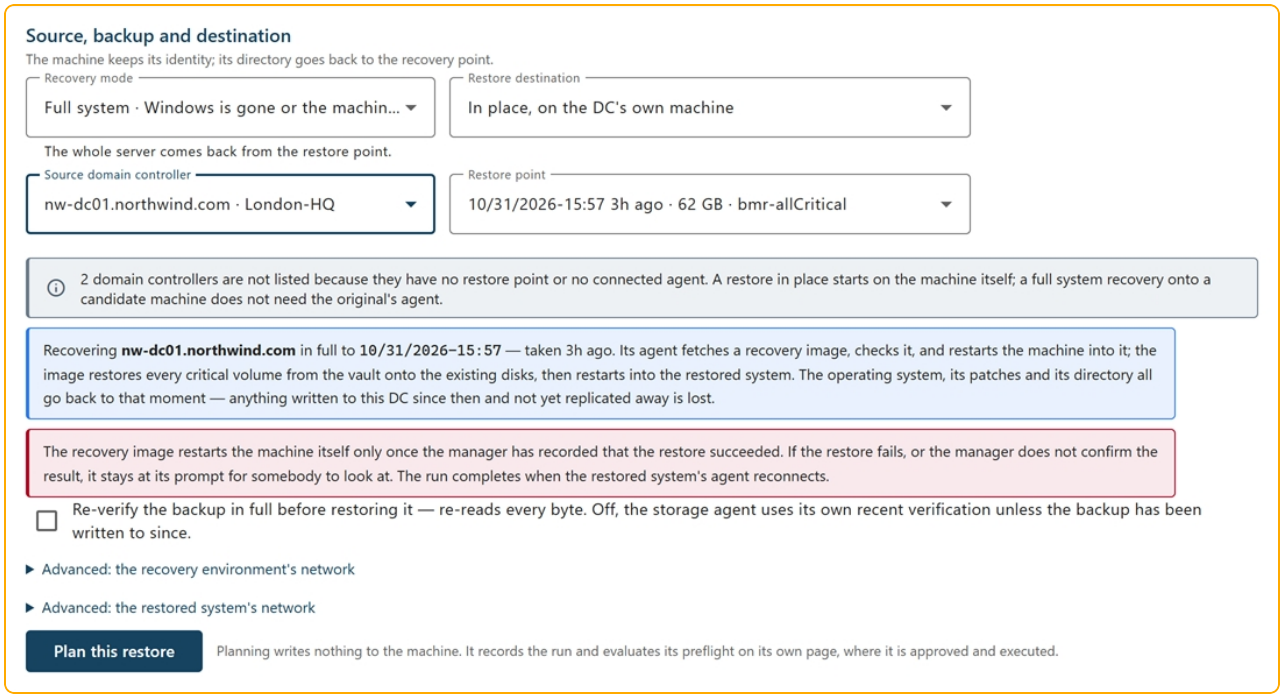
Planning a recovery writes nothing to any machine. It records the plan and evaluates its preflight checks.

New single-DC restore

Select **Single-DC recovery** to open the **New single-DC restore** page. It restores one domain controller to one of its own backups. The machine keeps its identity; its directory goes back to the restore point.

Which domain controllers are listed depends on **Restore destination**. In place, only domain controllers with a restore point and a connected agent are listed, because the restore is driven by the agent on the machine itself. Onto a candidate machine, every domain controller with a restore point is listed, and one whose agent is down is marked

agent not connected: the candidate's agent drives the restore. If nothing can be recovered, the page says **No domain controller can be restored in place, and no candidate machine is enrolled.** A notice counts the domain controllers left out and why.



Field	Description
Recovery mode	**DC state**–Windows still starts; the subtrees and objects you name go back to the restore point on every domain controller. This is a supported restore through Directory Services Restore Mode, not a VM snapshot revert. **Full system**–Windows is gone or the machine is being replaced; the whole server comes back from the restore point. For the difference, see Restore Methods.
Restore destination	Full system recovery only. **In place, on the DC's own machine**, the default, or a candidate machine: a machine with the DC agent enrolled that is not a domain controller. The storage machine and Isolated Recovery Hosts are never listed as candidates.
Source domain controller	The domain controller to restore, with its site. It is preselected when only one is connected.
Restore point	The backup to restore, shown with its version identifier, age, size, and type. The newest usable restore point, by when the domain controller completed it, is preselected. Restore points that cannot be used are listed but disabled, with the reason: no version identifier

Field	Description
	was captured; a DC state backup chosen for a full system recovery; the machine has been patched since the backup (DC state recovery); or the machine now runs a different Windows release (DC state recovery).
What to roll back	DC state recovery only. One or more rows of **Scope** (**Subtree** or **One object**) and **Distinguished name**; **Add a subtree or object** adds a row. Each is marked authoritative after the restore. The **Not accepted, and why** list shows what the Manager refuses. When a subtree holds `krbtgt`, the page warns that `krbtgt` and the users' passwords go back to the backup, and offers **Reset krbtgt twice once the rollback validates**, selected by default. See Restoring a DC with DC State Recovery.
Re-verify the backup in full before restoring it	Off by default. When on, the storage agent re-reads every byte of the backup and checks it against its recorded hash. When off, it relies on its own recent verification unless the backup has been written to since.

When you choose a restore point, the page describes the consequence: what goes back to that moment, and that anything written to this domain controller since then and not yet replicated to other DCs is lost. For a full system recovery it also explains that the recovery image restarts the machine into the restored system only after the Manager records that the restore succeeded. If the restore fails or the result is not confirmed, the recovery environment stays at its prompt for someone to look at.

If a restore of the same machine is already in progress, the page says so and offers **Open that run**. Follow, stop, or discard that run before planning another.

Network settings for a full system recovery

A full system recovery has two folded sections for network settings. Leave them at their defaults unless the recovery network needs something different.

Setting	Options
Recovery environment address (in **Advanced: the recovery environment's network**)	**DHCP** (default), **This machine's own address** (its static address, as it last described itself), or **Another address**. For another address, enter **Address, with prefix** (for example `10.99.0.10/24`), and optionally **Gateway** and **DNS servers, comma-separated**. The recovery environment reaches the vault at the storage agent's advertised address, its IP address by default; give it a DNS server only if the storage machine is advertised by name. If the machine has several adapters, enter the **Adapter MAC**.
Restored system address (in **Advanced: the restored system's network**)	**Automatic** (default: onto another machine, the backup's own address; in place, the backup's own address only when the machine's description says which adapter to use, otherwise unchanged), **As the backup left it** (nothing is changed), **The backup's own address** (put back on the target's adapter), or **Another address**, with the same address, gateway, DNS, and **Target adapter MAC** fields.

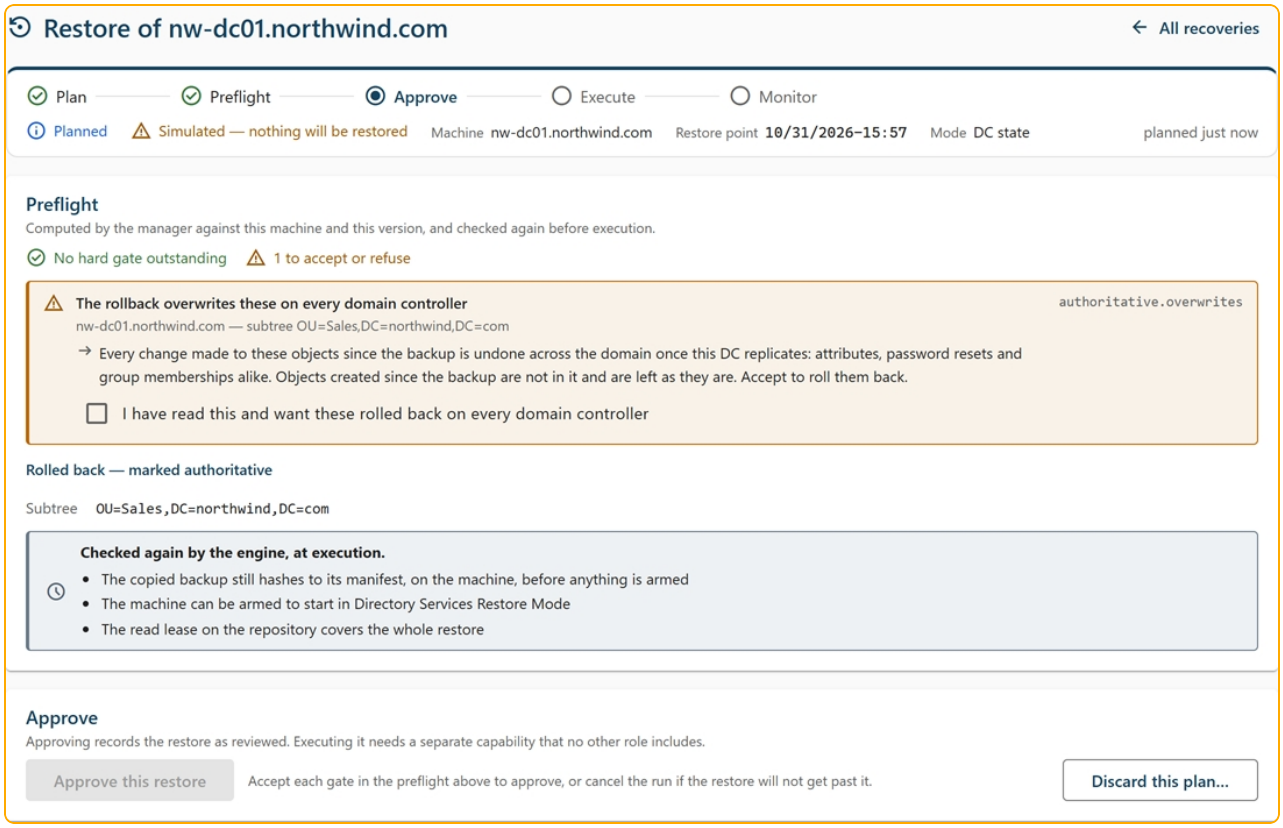
The recovery environment applies its address before it contacts anything. If it cannot, the run refuses before writing and the machine restarts into its own Windows; it never falls back to DHCP. The restored system's address is applied by the restored domain controller's agent at its first start, from a directive the Manager signs. If that fails, the failure is written to `POST-RECOVERY-README.txt` on the machine and reported when the agent connects.

Plan this restore

Select **Plan this restore** to record the run and open its page, where its preflight is evaluated and where it is approved and executed. Planning writes nothing to the machine and requires `plan.approve`. If you leave the page before planning, the console asks you to confirm, because the choices are not saved anywhere until the plan is built.

Restore run page

Each single-DC restore has its own page, titled **Restore of <machine>**. A stepper across the top shows the stage: **Plan** › **Preflight** › **Approve** › **Execute** › **Monitor**. The summary bar shows the status, **Machine**, **Restore point**, **Mode**, and when the run was planned, started, or finished. The page refreshes every few seconds; if the Manager stops answering, a **Not refreshing** notice says how old the view is.



Restore run statuses

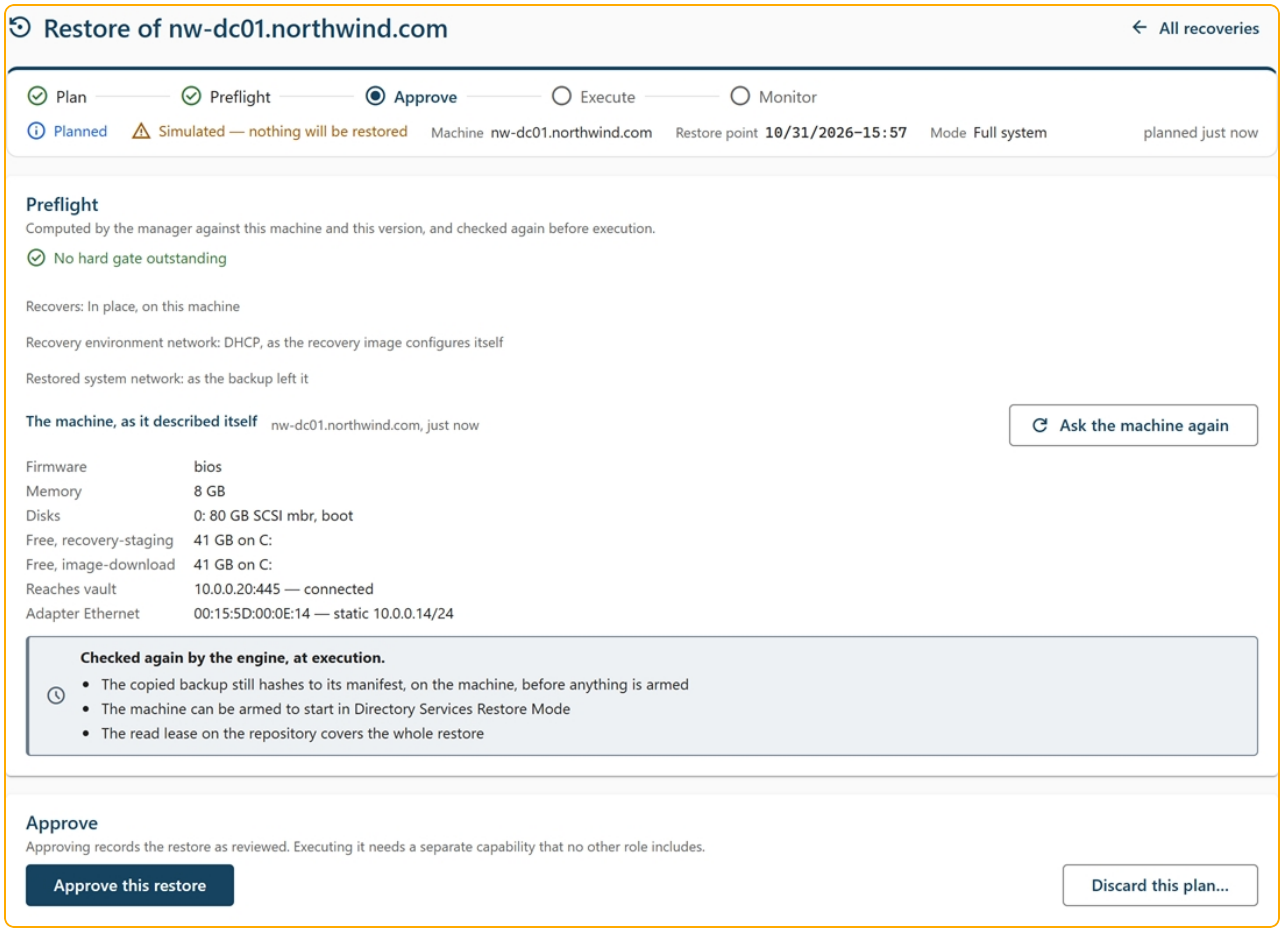
The summary bar shows the run's status, from **Planned** and **Approved** through the execution phases to **Restored**, **Failed**, **Cancelled**, or **Stopped — needs an operator**. For what each status means, see Recovery Lifecycle.

Preflight card

The **Preflight** card shows the checks the Manager computed against this machine and this restore point. They are checked again before execution. Each check (gate) shows a title, the subject, what to do, and a code:

- **Hard**—Blocks approval. The chip reads **N hard, blocking approval**, or **No hard gate outstanding**.
- **Advisory**—Read, not cleared. Advisory gates do not block approval; they say what the restore costs.
- **To accept or refuse**—Something that failed from where the Manager looked but may not fail from where the recovery environment runs, such as a storage machine name the target cannot resolve now. The chip reads **N to accept or refuse**. Each such gate has a checkbox, **I have read this and accept it: the restore will get past it, safely, from where it runs**; approval stays unavailable until you select it for every one, or you

cancel the run instead. After approval the chip reads **N accepted at approval**, and each gate shows who accepted it.



For a full system recovery, the card also shows the planned recovery environment and restored system network settings, the status of post-recovery directives, and **The machine, as it described itself**: the target's firmware, memory, disks, free space, whether it can reach the storage machine and Manager, its adapters, and whether the original domain controller still answers. Select **Ask the machine again** (`plan.approve`, while the run is planned or approved) to refresh this survey.

For a DC state recovery, the card shows **Rolled back — marked authoritative**: each subtree and object the run marks, or **Nothing: a nonauthoritative restore** for a restore started by a domain or forest plan. For a DC state recovery that resets `krbtgt`, approving asks for a **Domain administrator** and **Password**; if the reset later fails, **krbtgt has not been reset twice yet** offers **Reset krbtgt**.

The card ends with the checks the Manager makes again at execution: the backup's hash against the vault, that the version is present at the target, that the machine's boot configuration can be set and cleared, for a full system recovery the recovery environment's own view of firmware and disks, and, after the restore, the restored domain controller's own checks of AD DS, SYSVOL, replication, and dcdiag.

For every gate, what it means, and how to clear it, see Preflight Check Reference.

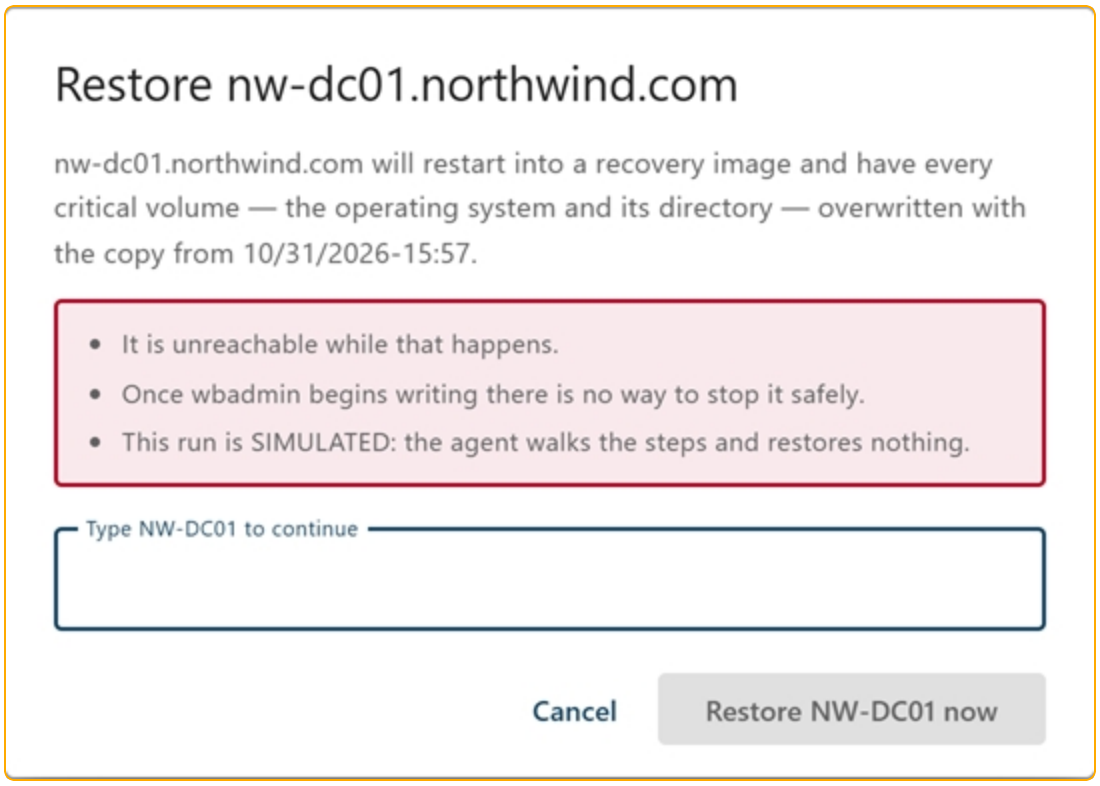
Approve and execute

Approving records the restore as reviewed. Executing it needs a separate capability that no other role includes.

Button	Capability	What it does
Approve this restore	`plan.approve`	Records the run as reviewed, with each gate you accepted. Disabled while any hard gate remains or any gate to accept is unaccepted; the Manager refuses an approval over a preflight that does not pass. If a forest or domain plan was waiting on the run and you also hold `plan.execute`, the plan carries on at once.
Withdraw approval	`plan.approve`	Returns an approved run to Planned.
Execute…	`plan.execute`	Starts the restore after a confirmation (see below).
Discard this plan…	`plan.approve`	Closes a run that was never executed as cancelled. Nothing has been sent to the machine, and it can be planned again straight away. Not offered for a run that belongs to a forest or domain plan.
Cancel this restore…	`plan.execute`	Shown for a planned or approved run that a running or halted forest or domain plan holds. Closes the run as cancelled; nothing has been sent to the machine. The plan step that started it fails, and the plan halts.
Ask the machine to stop…	`plan.execute`	Shown while the run is in flight. The machine abandons the restore only if wbadmin has not started writing yet; past that point the Manager refuses, because stopping a restore midway cannot be done safely.
Close as abandoned…	`plan.execute`	Shown once the run is past the point where it can be stopped. Use it when the machine is gone or will be rebuilt. The run

Button	Capability	What it does
		is closed as failed, nothing is sent to the machine, and the machine can be planned again. You must enter a reason, which is recorded in the audit trail. Do not use that machine as a domain controller until it is rebuilt or its transcript has been read.
Validate again	`plan.approve`	Shown for a finished run whose checks show **Not checked** or failed, when the machine's agent is connected and can make them. Asks the restored domain controller to check AD DS, SYSVOL, replication, and dcdiag again, waiting up to 15 minutes for replication. Changes nothing on the machine. If the checks cannot be made now, the page says why.
Plan another restore	`plan.approve`	Shown when the run has finished. Opens a new single-DC restore.

Executing asks for confirmation. The dialog, titled **Restore <machine>**, states what will happen: for a DC state recovery, the machine restarts into Directory Services Restore Mode, has its Active Directory database replaced, and restarts again; for a full system recovery, it restarts into a recovery image and has every critical volume overwritten. The machine is unreachable while that happens, and once wbadmin begins writing there is no way to stop it safely. To confirm, type the machine's short name in uppercase (for example `DC01`) and select **Restore DC01 now**.



In progress and outcome

While the run is in flight, the **In progress** card shows whether stopping is still possible: **Safe to stop — nothing was changed**, **Stopping needs cleanup — the machine was changed**, or **Cannot be stopped or resumed — needs an operator**. While the machine restarts, a notice says how long it is expected to be unreachable; silence during that time is the restore working, not the agent being lost.

When the run finishes, the card becomes **Outcome** and shows one of these results:

Outcome	Meaning
Restored and validated	The restore completed and the validation checks passed.
Restored, but a validation gate did not pass	The restore completed; a post-restore check failed.
Refused before anything was touched	Preflight refused the restore before the machine was changed.
Stopped before anything was written	The run was stopped in time.
wbadmin refused the restore and wrote nothing	wbadmin declined the restore.

Outcome	Meaning
wbadmin failed after it began writing	The restore failed partway through.
The machine restarted but did not enter restore mode	A DC state recovery could not reach Directory Services Restore Mode.
The machine could not be armed for restore mode	The boot configuration could not be set.
Restore mode was left before the restore began	The machine left Directory Services Restore Mode too early.
The read lease could not cover the work or **The read lease expired before the restore started**	The time-limited credential for reading the backup was too short.
The machine restarted while the restore was writing	The restore was interrupted.
The machine stopped reporting before the restore touched it	The agent went silent early.
The machine stopped reporting mid-restore — outcome unknown	Nothing outside the machine can tell what state it is in.
The recovery image could not be staged on the machine	A full system recovery could not stage the recovery image.
The machine restarted into its own Windows, not the recovery image	A full system recovery did not boot the recovery image.
Closed as abandoned by an operator — what is on the machine is unknown	Someone selected **Close as abandoned**.

When the outcome needs a person, the card shows **This machine needs you**. Do not start another restore of the machine until someone has looked at it: its directory may be inconsistent. The two ways forward are a fresh restore from a different restore point, or rebuilding the domain controller. The agent leaves a note and the full transcript on the machine's own disk.

Steps and what the machine reported

The **Steps** card lists the three checks resolved for this run: a hard block if the domain controller's operations master roles were seized elsewhere after the backup, checked when the run is dispatched; AD DS running and SYSVOL ready; and replication converging and `dcdiag` results coming back clean. Each step shows its state: **Not started**, **Running**,

Done, **Failed**, **Skipped**, or **Not checked**, which means the restored agent could not make the check. In a forest or domain plan, the first restored domain controller of a domain has no partners yet, so replication is not required of it.

The **What the machine reported** card shows each phase the agent reported, such as **Checking the machine**, **Copying the backup to this machine**, **Arming restore mode**, **Restoring DC state**, or, for a full system recovery, **Fetching the recovery image** and **Restoring volumes**. Times are shown on the Manager's clock. It then shows the last lines of the transcript and the path of the full transcript on the machine. After the run finishes, the preflight stays available as **Preflight, as approved**.

New forest or domain recovery

Select **Forest recovery** or **Domain recovery** to open the plan builder, titled **New forest recovery** or **New domain recovery**. The builder has two steps, shown in the summary bar along with the counts of domain controllers to restore, rebuild, and exclude, and a **Security breach** chip if you mark the plan as a breach response.

A forest can have only one plan at a time, whether of the whole forest or of a domain in it. If another plan is active, an **Already active** notice links to it.

Step 1: Choose

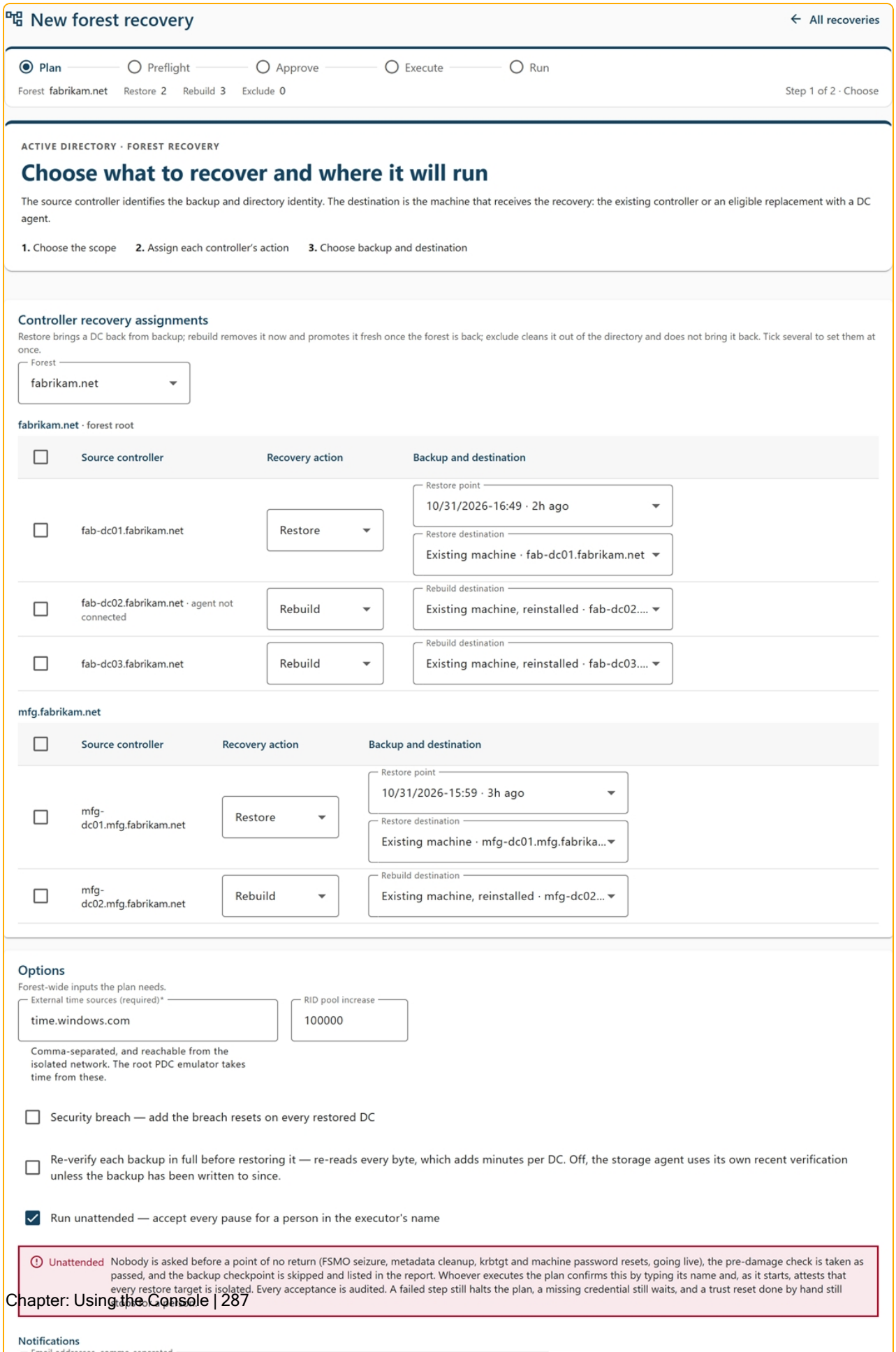
Choose the **Forest**, and for a domain recovery the **Domain to recover**. A domain recovery cannot recover a forest root domain; if the forest has only its root domain, use a forest recovery.

The **Controller recovery assignments** card has a table per domain, with these columns. Retired DCs are not listed. DCs of the forest with no known domain are not in the plan either; a notice above the tables counts them and points to the Domain Controllers screen, because this is usually a demotion that did not finish.

Column	Description
Source controller	The DC, marked **read-only**, **agent not connected**, or **excluded from protection** where that applies.
Recovery action	**Restore** brings the DC back from backup. **Rebuild** removes it now and promotes it fresh once the domain or forest is back. **Exclude** neither restores nor rebuilds it: its metadata is removed from the directory along with the domain's other DCs, and it is not brought back. Keep the machine off the network. **Restore** is not available for a read-only DC.
Backup and destination	For a DC set to Restore, the **Restore point** and **Restore destination** (**Existing machine** or a candidate machine, listed as

Column	Description
	Replacement). Restore points with no version identifier, or DC state backups, cannot be chosen, and a backup taken while an earlier recovery was under way is never the default. For a DC set to Rebuild, **Rebuild destination**: **Existing machine, reinstalled**, or a candidate machine (**Replacement**), which the plan renames to the DC's name and promotes.

By default, in each domain the connected writable PDC emulator (or the first eligible DC) is set to Restore and every other DC to Rebuild. A DC excluded from protection starts as Exclude; if only excluded DCs in a domain can be restored, one of them is still set to Restore. Select several rows to set their recovery action at once. Each domain must restore exactly one DC, and each machine can be bound to only one DC; the builder says so before review if not.


New forest recovery
All recoveries
Plan
Preflight
Approve
Execute
Run
Forest fabrikam.net
Restore 2
Rebuild 3
Exclude 0
Step 1 of 2 · Choose
ACTIVE DIRECTORY · FOREST RECOVERY
Choose what to recover and where it will run
The source controller identifies the backup and directory identity. The destination is the machine that receives the recovery: the existing controller or an eligible replacement with a DC agent.
1. Choose the scope
2. Assign each controller's action
3. Choose backup and destination
Controller recovery assignments
Restore brings a DC back from backup; rebuild removes it now and promotes it fresh once the forest is back; exclude cleans it out of the directory and does not bring it back. Tick several to set them at once.
Forest
fabrikam.net
fabrikam.net · forest root
Source controller
Recovery action
Backup and destination
fab-dc01.fabrikam.net
Restore
Restore point
10/31/2026-16:49 · 2h ago
Restore destination
Existing machine · fab-dc01.fabrikam.net
fab-dc02.fabrikam.net · agent not connected
Rebuild
Rebuild destination
Existing machine, reinstalled · fab-dc02....
fab-dc03.fabrikam.net
Rebuild
Rebuild destination
Existing machine, reinstalled · fab-dc03....
mfg.fabrikam.net
Source controller
Recovery action
Backup and destination
mfg-dc01.mfg.fabrikam.net
Restore
Restore point
10/31/2026-15:59 · 3h ago
Restore destination
Existing machine · mfg-dc01.mfg.fabrika...
mfg-dc02.mfg.fabrikam.net
Rebuild
Rebuild destination
Existing machine, reinstalled · mfg-dc02...
Options
Forest-wide inputs the plan needs.
External time sources (required)*
time.windows.com
Comma-separated, and reachable from the isolated network. The root PDC emulator takes time from these.
RID pool increase
100000
Security breach — add the breach resets on every restored DC
Re-verify each backup in full before restoring it — re-reads every byte, which adds minutes per DC. Off, the storage agent uses its own recent verification unless the backup has been written to since.
Run unattended — accept every pause for a person in the executor's name
Unattended
Nobody is asked before a point of no return (FSMO seizure, metadata cleanup, krbtgt and machine password resets, going live), the pre-damage check is taken as passed, and the backup checkpoint is skipped and listed in the report. Whoever executes the plan confirms this by typing its name and, as it starts, attests that every restore target is isolated. Every acceptance is audited. A failed step still halts the plan, a missing credential still waits, and a trust reset done by hand still
Notifications

For a domain recovery, an **Outside the domain** card lists what the plan changes in the rest of the forest: the global catalogs outside the domain it will reconcile (a global catalog with no agent is flagged and blocks the plan), the trusts it will reset on both sides (by the agents where the partner domain has a live DC with an agent, otherwise by hand), and a confirmation it will ask for that each domain's default administrator can sign in while the global catalogs are being reconciled.

The **Options** card holds the plan's inputs:

Option	Description
Global catalog reconciliation (domain recovery)	**Remove lingering objects — GCs stay online** (default) removes the recovered domain's lingering objects from each global catalog, in advisory mode first. **Unhost, then rehost — for small forests** has every global catalog drop the domain and take it back site by site; they cannot answer for it in between.
External time sources (required) (forest recovery)	One or more time sources, comma-separated, reachable from the isolated network. The root PDC emulator takes time from them.
RID pool increase	How much to raise each domain's available RID pool. The default is 100000; it must be positive.
Security breach	Off by default. Adds the breach resets on every restored DC. For a forest recovery, it also adds the trust resets; without it, a forest recovery resets no trust. For a domain recovery, it also re-creates the domain's group managed service accounts; a domain recovery resets the domain's trusts either way.
Re-verify each backup in full before restoring it	Off by default. Re-reads every byte of each backup, which adds minutes per DC.
Run unattended	Off by default. The plan does not stop for a person at points of no return, attestations, the backup checkpoint, or restore preflight confirmations. A warning lists what is accepted. The acceptances are made, and recorded, in the name of whoever executes the plan, who confirms it at execution. A failed step still halts the plan, a missing credential still waits, and a trust reset carried out by hand—an external trust, or a partner with no DC agent—still stops for a person. Trust resets the agents carry out are accepted. See Recovering a Forest.
Notifications: **Email**	Empty by default: nothing is emailed. These addresses get

Option	Description
addresses, comma-separated	the plan's halts, aborts, and waits for a person, and its report on completion. Shown only with `plan.execute`; without it, a note says who can add addresses later.
New DSRM password: **Password** and **Password again**	Empty by default: the DSRM step waits for one. The new Directory Services Restore Mode administrator password the plan sets on every DC it recovers. **Password again** appears once **Password** is filled in; if the two differ, the builder says so. Held by the Manager in memory only and never shown again. Shown only with `plan.execute`.

Select **Review the plan**. The button stays disabled, with the reason beside it, until you have chosen a forest (and domain), at least one DC is set to Restore, every DC being restored has a restore point, and, for a forest recovery, at least one external time source is given.

Step 2: Review

The **Recovery of <forest or domain>** card states what the plan will do in each domain (which DC is restored from which restore point and where, which are rebuilt, and which are left out) and across the forest (RID pool, time source, backup verification, whether this is a breach response, whether it runs unattended, who it emails, and whether it holds a new DSRM password). Select **Back to the choices** to change anything, or **Build the plan** to record it and open its page. Building requires `plan.approve` and writes nothing to any machine.

Forest or domain plan page

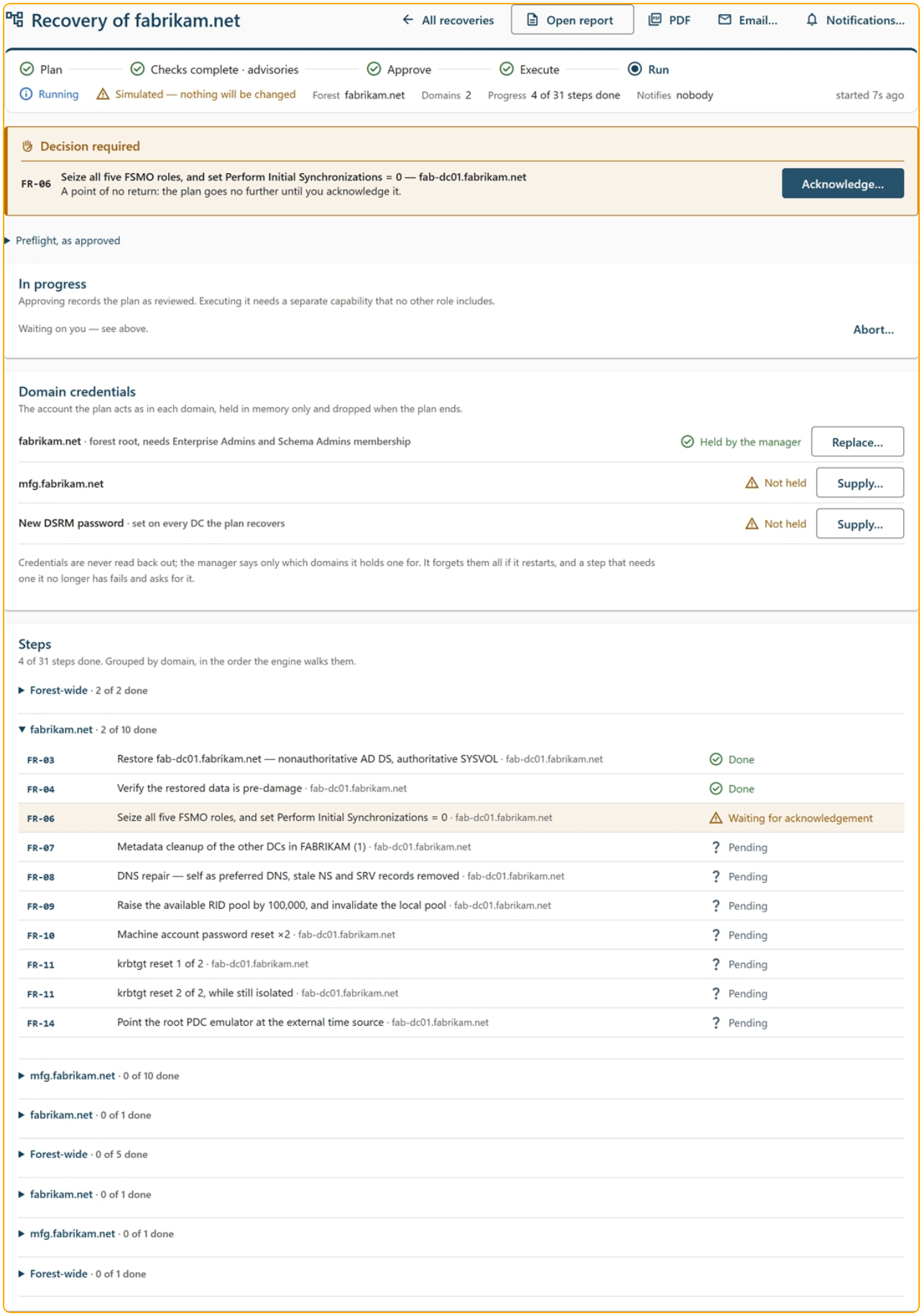
Each forest or domain plan has its own page, titled **Recovery of <forest or domain>**. The stepper shows **Plan** › **Preflight** › **Approve** › **Execute** › **Run**; the **Preflight** stage reads **Checks complete · advisories** when no hard gate is open but advisories remain. The summary bar shows the status, the forest and domain, the number of domains, progress (**Not started** until the plan starts, then steps done of total), **Notifies** (the plan's notification addresses, or **nobody**), and, for an unattended plan, an **Unattended** chip whose tooltip names whose name pauses are accepted in.

Banners below the summary bar show the plan's state when it matters:

- **Halted**—names the step that failed, the machine, and the reason, and offers **Retry <step>…**.
- **Live on reduced DCs**—from the reconnect until the last rebuilt DC is confirmed, the recovered directory is in production on fewer domain controllers than planned. The

banner says since when and which DCs each domain is serving on, and warns when a domain is on one DC with no redundancy. Afterward, a note says how long the directory was live on reduced DCs.

- **Report**–the result of the last report download or email from this page.


Recovery of fabrikam.net
All recoveries
Open report
PDF
Email...
Notifications...
Plan
Checks complete · advisories
Approve
Execute
Run
Running
Simulated — nothing will be changed
Forest fabrikam.net
Domains 2
Progress 4 of 31 steps done
Notifies nobody
started 7s ago
Decision required
FR-06
Seize all five FSMO roles, and set Perform Initial Synchronizations = 0 — fab-dc01.fabrikam.net
A point of no return: the plan goes no further until you acknowledge it.
Acknowledge...
Preflight, as approved
In progress
Approving records the plan as reviewed. Executing it needs a separate capability that no other role includes.
Waiting on you — see above.
Abort...
Domain credentials
The account the plan acts as in each domain, held in memory only and dropped when the plan ends.
fabrikam.net · forest root, needs Enterprise Admins and Schema Admins membership
Held by the manager
Replace...
mfg.fabrikam.net
Not held
Supply...
New DSRM password · set on every DC the plan recovers
Not held
Supply...
Credentials are never read back out; the manager says only which domains it holds one for. It forgets them all if it restarts, and a step that needs one it no longer has fails and asks for it.
Steps
4 of 31 steps done. Grouped by domain, in the order the engine walks them.
Forest-wide · 2 of 2 done
fabrikam.net · 2 of 10 done
FR-03 Restore fab-dc01.fabrikam.net — nonauthoritative AD DS, authoritative SYSVOL · fab-dc01.fabrikam.net Done
FR-04 Verify the restored data is pre-damage · fab-dc01.fabrikam.net Done
FR-06 Seize all five FSMO roles, and set Perform Initial Synchronizations = 0 · fab-dc01.fabrikam.net Waiting for acknowledgement
FR-07 Metadata cleanup of the other DCs in FABRIKAM (1) · fab-dc01.fabrikam.net Pending
FR-08 DNS repair — self as preferred DNS, stale NS and SRV records removed · fab-dc01.fabrikam.net Pending
FR-09 Raise the available RID pool by 100,000, and invalidate the local pool · fab-dc01.fabrikam.net Pending
FR-10 Machine account password reset ×2 · fab-dc01.fabrikam.net Pending
FR-11 krbtgt reset 1 of 2 · fab-dc01.fabrikam.net Pending
FR-11 krbtgt reset 2 of 2, while still isolated · fab-dc01.fabrikam.net Pending
FR-14 Point the root PDC emulator at the external time source · fab-dc01.fabrikam.net Pending
mfg.fabrikam.net · 0 of 10 done
fabrikam.net · 0 of 1 done
Forest-wide · 0 of 5 done
fabrikam.net · 0 of 1 done
mfg.fabrikam.net · 0 of 1 done
Forest-wide · 0 of 1 done

Plan statuses

Status	Meaning
Draft	Built, with gates evaluated. Can be edited with **Edit this draft**. Not yet approved.
Approved	Approved and waiting to be executed.
Running	The Manager is working through the steps.
Halted — a step failed	A step failed. Nothing more is issued until the step is retried or the plan is aborted.
Completed	Every step finished. Each DC the plan restored or rebuilt that has an enabled backup policy is made due for a backup, which starts when its agent connects.
Aborted	An operator aborted the plan. This is final: build a new plan rather than resuming.

Actions

Button	Capability	What it does
Edit this draft	`plan.approve`	Reopens the builder with the draft's choices. If a chosen restore point is no longer available, the builder picks another and tells you.
Approve this plan	`plan.approve`	Records the plan as reviewed. Every hard gate must clear first.
Execute…	`plan.execute`	Starts the plan after you type the forest or domain name to confirm. Each point of no return still stops for your acknowledgment. For an unattended plan, the dialog is **Execute the UNATTENDED recovery of <forest or domain>**: it lists what will be accepted without asking, and you also attest that the restore targets are isolated, then select **Execute unattended**. Aborting later stops the plan but does not undo steps that have run.

Button	Capability	What it does
Acknowledge…	`plan.execute`	Lets a point-of-no-return step proceed. Type the step code shown in the dialog to confirm. Past this step the plan cannot be walked back.
Attest	`plan.execute`	Confirms a check only a person can make, such as that the restore targets are isolated. A single selection, with no dialog.
Reject…	`plan.execute`	Rejects an operator step and halts the plan. The dialog asks for a reason.
Run it now…	`plan.execute`	Shown only when the plan was created with a gap between the two krbtgt resets. Ends the wait early. Kerberos tickets issued between the resets are invalidated. You must enter a reason.
Supply… or **Replace…**	`plan.execute`	Hands the plan the administrator credential for a domain, or the new DSRM password (see below).
Retry <step>…	`plan.execute`	Shown on a halted plan. Runs the failed step again from its checkpoint, after you select **Retry the step**, and the plan carries on. What the step already did is not repeated, and its point-of-no-return acknowledgment stands. Fix the cause of the failure before you retry.
Take back	`plan.execute`	Shown on an unattended plan that has not finished. After you select **Take it back**, every pause waits for a person. It cannot be made unattended again.
Notifications…	`plan.execute`	Opens **Who this recovery emails**. Edit **Addresses**, comma-separated, and select **Save**. Empty means nothing is emailed. If the plan had no addresses and is already halted or aborted when you add some, the new addresses are emailed that it halted or was aborted.

Button	Capability	What it does
Open report	`plan.view`	Shown once the plan has started. Opens the recovery report in a new browser tab. It is built from the plan's records each time, so it shows the plan as it is now.
PDF	`plan.view`	Downloads the report as a PDF, made by the Manager's own PDF renderer. If the Manager cannot make it, the page shows its reason.
Email…	`plan.execute`	Opens **Email the recovery report**. Enter addresses in **Send to**, or leave it empty for the plan's notification addresses, and select **Send the report**. The report is the email itself, with its PDF attached when the Manager can make one. Needs the Manager's email settings. Recorded in the audit trail with its recipients.
Abort…	`plan.execute`	Stops the plan where it is. It cannot be resumed. Steps already run are not undone; each restore the plan started is asked to stop, and one already writing carries on. The dialog says what aborting now leaves—which points of no return are behind the plan, and which changes stay to finish or undo by hand—and asks for a reason.
Plan another recovery	`plan.approve`	Shown when the plan has finished.

Reasons you enter are recorded in the audit trail.

Decision required

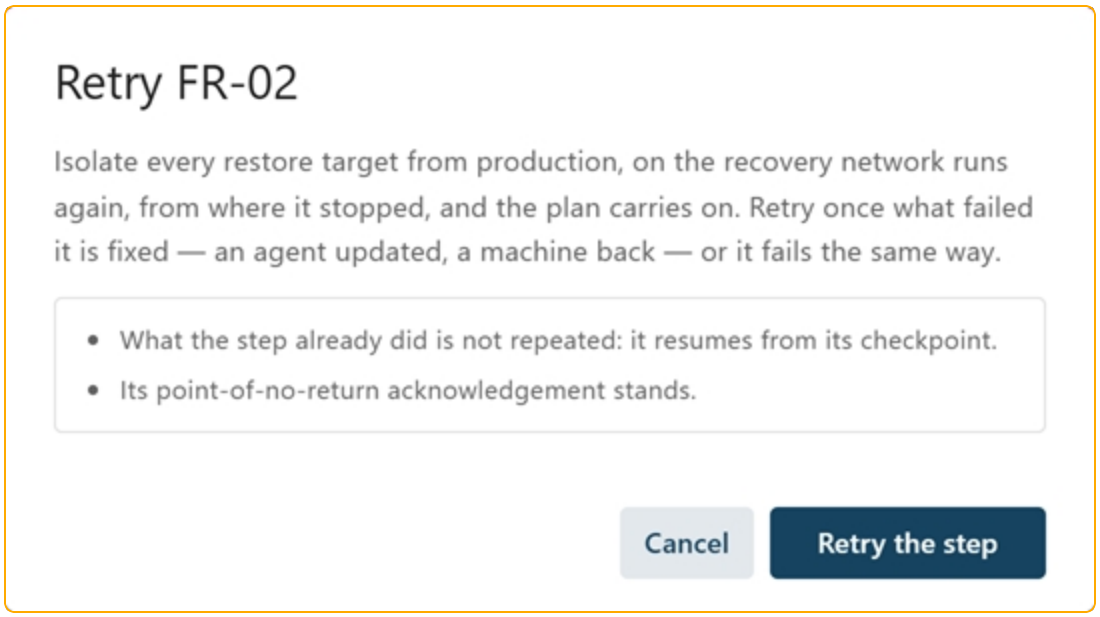
When the plan is waiting on a person, a **Decision required** section at the top of the page says what it needs, and the recovery banner at the top of the other console screens, except the Dashboard, reads **A recovery is waiting for you**:

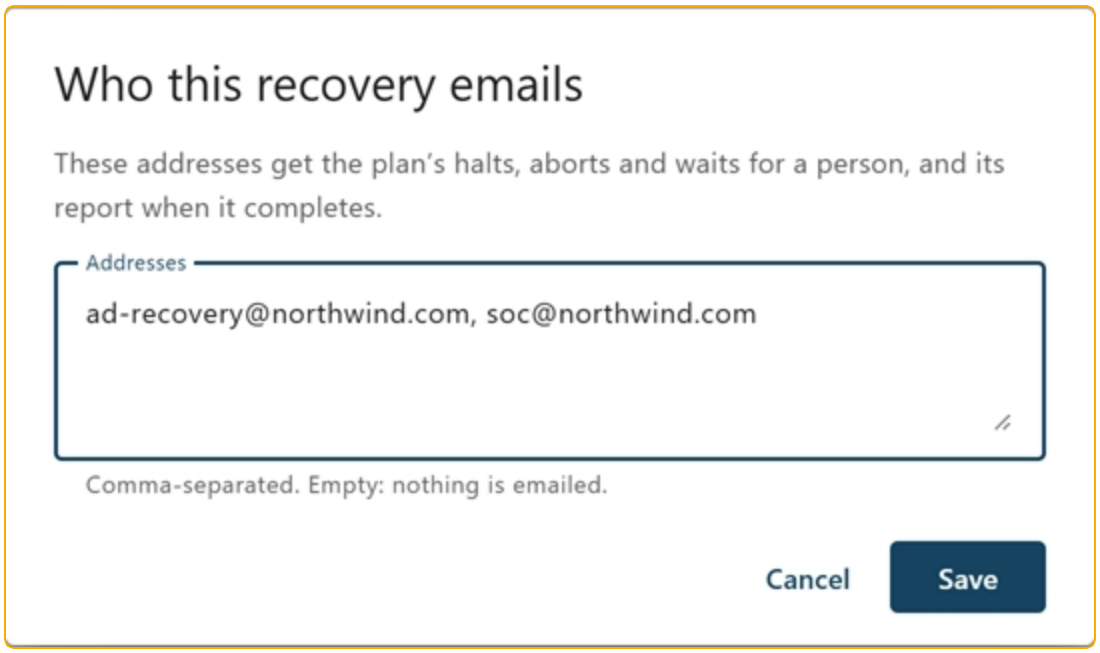
- **Waiting for acknowledgement**—A point of no return. Select **Acknowledge…**.
- **Waiting for you**—Something only a person can check, with a prompt that says what. Select **Attest** or **Reject…**. If the step is a restore whose preflight raised a gate only a

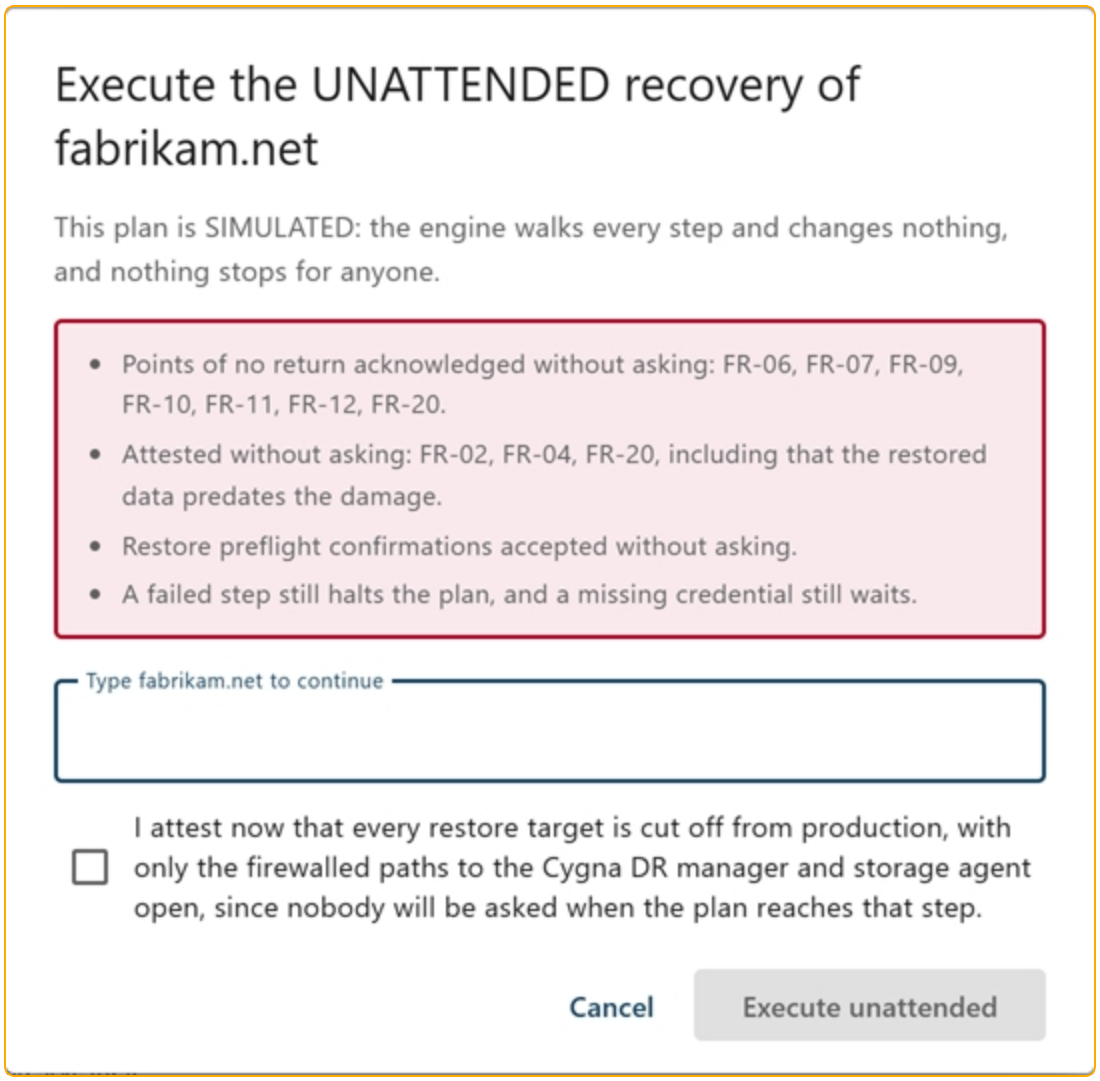
person can accept, select **Decide on the restore** instead, and approve or cancel the restore run.

- **Waiting for a credential**—The step cannot run without a domain's administrator credential, or the new DSRM password, and the Manager holds none. Select **Supply credentials…** or **Supply DSRM password…**. The step runs by itself once it is supplied.
- **Waiting (krbtgt)**—The plan was created with a gap between the two krbtgt resets, and is waiting it out; it shows when it will continue by itself. Select **Run it now…** to continue early. By default there is no gap.
- **Waiting for its machine to be built**—The machine a rebuilt DC is to be promoted onto has never connected, so it still has to be built, with a base operating system and the DC agent enrolled. The step runs by itself once its agent connects.

A pending step's row says which steps it is waiting for, and a rebuilt machine restarting between steps shows **Restarting**, which needs nothing from you. If the plan has notification addresses, each wait for a person is emailed to them once.




Who this recovery emails
These addresses get the plan's halts, aborts and waits for a person, and its report when it completes.
Addresses
ad-recovery@northwind.com, soc@northwind.com
Comma-separated. Empty: nothing is emailed.
Cancel
Save

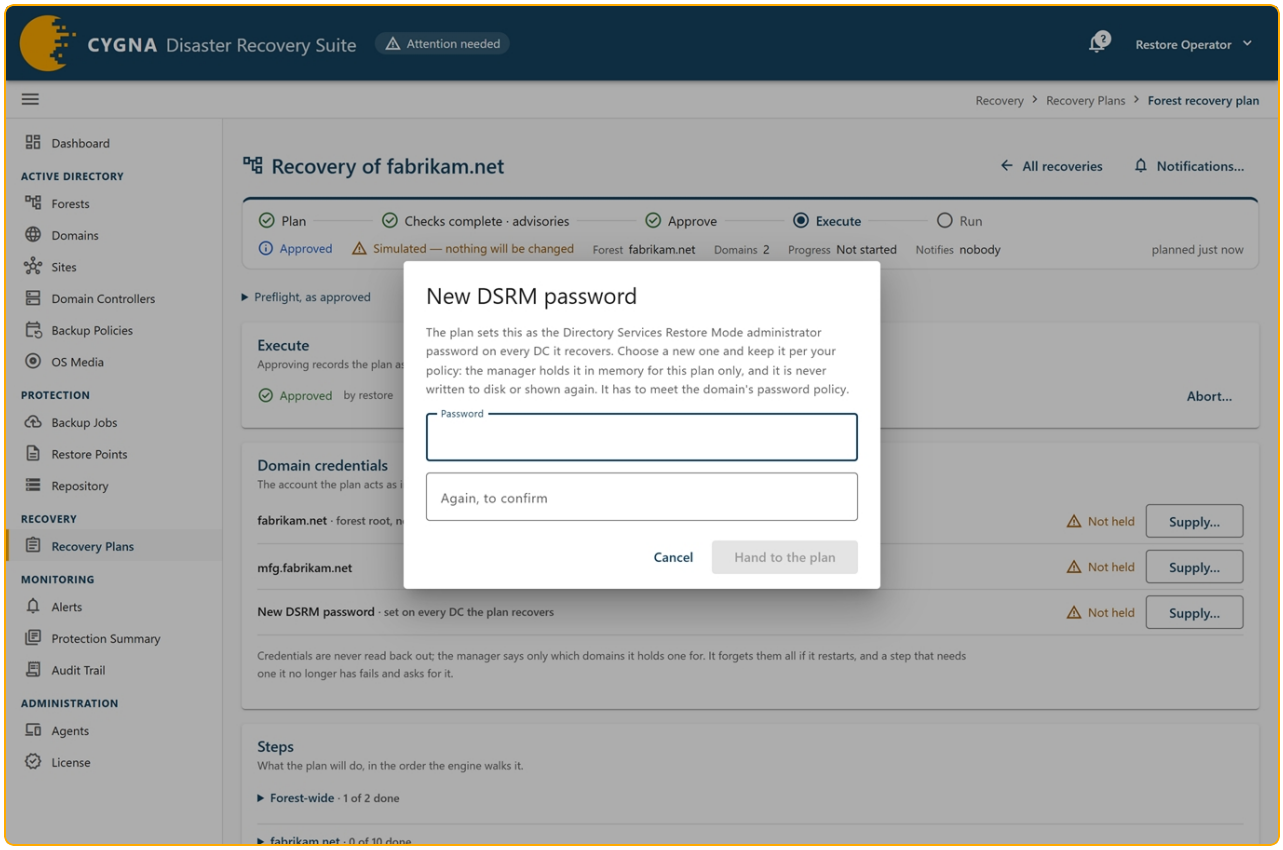

Execute the UNATTENDED recovery of fabrikam.net
This plan is SIMULATED: the engine walks every step and changes nothing, and nothing stops for anyone.
Points of no return acknowledged without asking: FR-06, FR-07, FR-09, FR-10, FR-11, FR-12, FR-20.
Attested without asking: FR-02, FR-04, FR-20, including that the restored data predates the damage.
Restore preflight confirmations accepted without asking.
A failed step still halts the plan, and a missing credential still waits.
Type fabrikam.net to continue
I attest now that every restore target is cut off from production, with only the firewalled paths to the Cygna DR manager and storage agent open, since nobody will be asked when the plan reaches that step.
Cancel
Execute unattended

Domain credentials

While the plan is approved, running, or halted, the **Domain credentials** card lists the account the plan acts as in each domain. In a domain recovery, the list also includes each partner domain whose live DC resets its side of a trust. Each domain shows **Held by the manager** or **Not held**. A step in a domain that is not held, and that needs a credential, is not sent: it shows **Waiting for a credential** until one is supplied, and then runs. The card reads this from the plan itself, so it is accurate when you open the page and after a Manager restart. Credentials are never read back out: the Manager says only which domains it holds one for. Select **Supply…** and enter the **Administrator account** (for example `NETBIOS\Administrator`) and **Password**, then select **Hand to the plan**. For the forest root domain, the account must be an Enterprise Admin and Schema Admin; for other domains, a Domain Admin of that domain.

The Manager holds credentials in memory for this plan only. They are never written to disk or shown again, and they are forgotten if the Manager restarts; a step that then needs one waits for it. Each supply is recorded in the audit trail and the recovery report, without the secret.

The card also has a **New DSRM password** row, set on every DC the plan recovers, showing **Held by the manager** or **Not held**. Select **Supply…** or **Replace…** to open **New DSRM password**, enter the **Password** and **Again, to confirm**, and select **Hand to the plan**. The password must meet the domain's password policy. Record it as your policy requires: the Manager holds it in memory only and never shows it again.



Plan as built, preflight, and steps

- **Plan as built**–What the plan was built from; it does not change. Columns: **Domain controller**, **What happens** (restored from backup, as a DC state recovery or a full system recovery, or rebuilt after the recovery), **Restore point**, and **Destination**. Each DC says at which step it is handled, for example **At FR-03** for a restore or **At FR-21, after FR-20** for a rebuild; each rebuild also says what it reinstalls, which roles come back without their configuration, and how many programs are not reinstalled. Expand **Recovery settings** below the table for the time source, RID pool increase, **Gap between krbtgt resets** (**None: both resets run while isolated** unless one was set when the plan was created), backup check before restoring, first global catalog, and global catalog strategy.
- **Preflight findings**–The plan's hard and advisory gates, shown while the plan is a draft and folded as **Preflight, as approved** afterwards. For each DC restored onto a candidate machine, they include the checks of the candidate's firmware and disks against the chosen restore point. Approval evaluates those again against the machines as they are then, saves the result on the plan, and is refused if one of them blocks. In a forest plan without the security breach flag, an advisory warns about each trust whose two sides are restored from points 30 days or more apart. This card and **Plan as built** sit above the approve and execute buttons. See Preflight Check Reference.
- **Steps**–Every step, grouped by domain and forest-wide, in the order the Manager runs them. Each group can be expanded or collapsed; a group opens by itself when one of its steps is running, has failed, or is waiting on a person. Each row shows the step's code and title, the machine, a **Follow the restore** link for restore steps, and its state, with the steps a pending step waits for and the reason for each check that failed.

Step states are **Pending**, **Waiting for acknowledgement**, **Waiting (krbtgt)**, **Waiting for its machine to be built**, **Restarting**, **Waiting for a credential**, **Running**, **Waiting for you**, **Done**, and **Failed**. A point of no return needs a typed acknowledgment and cannot be walked back.

For what each step does, see Recovering a Forest and Recovering a Domain.

Related topics

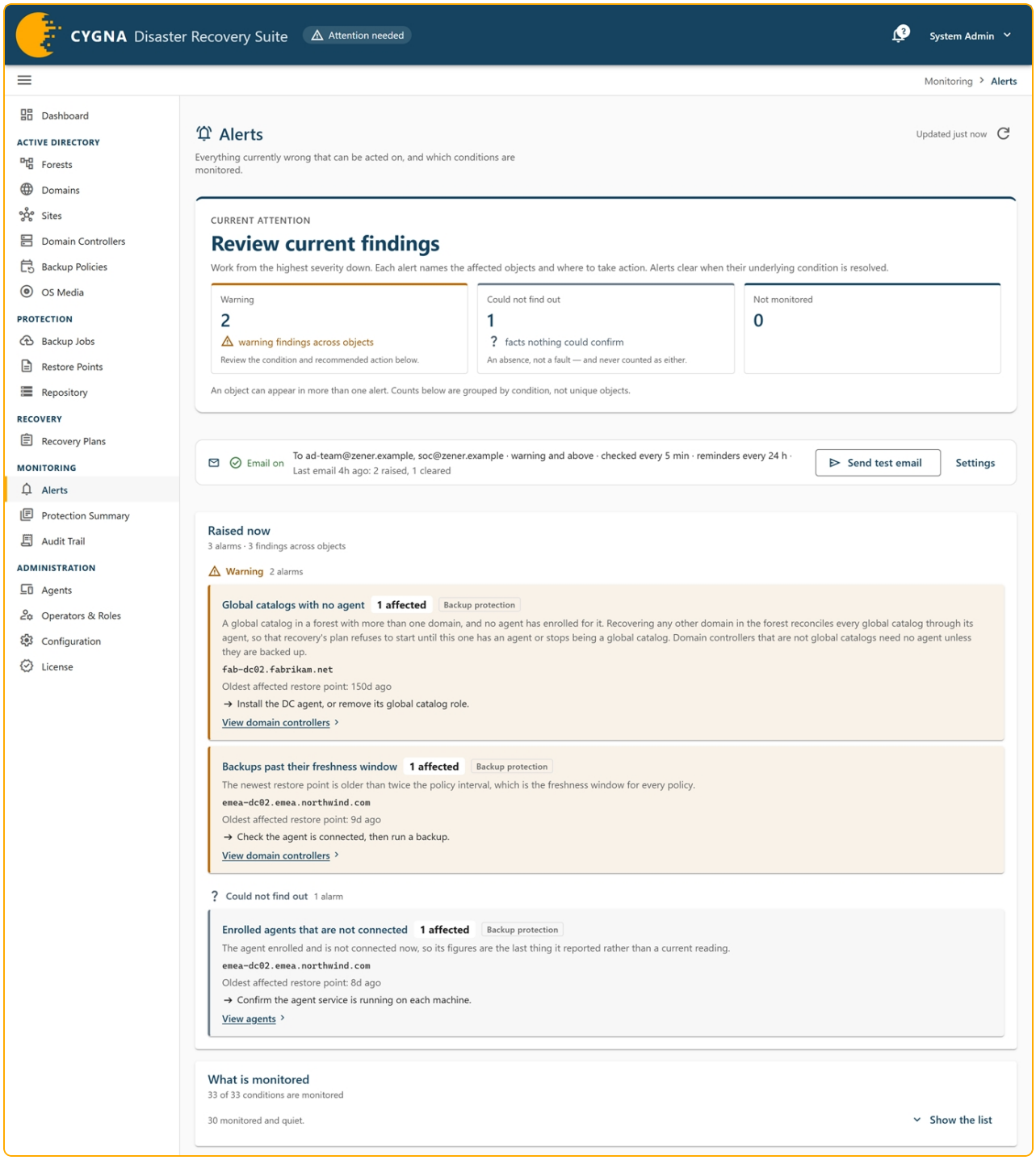
- Restoring a DC with DC State Recovery
- Full System Recovery in Place
- Full System Recovery onto a Candidate
- After a Recovery
- Recovery Requirements

Alerts

The **Alerts** screen lists everything currently wrong that you can act on: each alert the Manager raises, which objects it affects, and what to do about it. Alerts are evaluated continuously from what the agents report, and an alert disappears when its condition is no longer true. There is nothing to acknowledge, with one exception: **Committed vault content changed outside retention** stays raised until someone records a review of the change with `cdr repository changes review`.

To open it, select **Monitoring** > **Alerts**, or select **Open Alerts** in the notifications menu (the bell in the top bar). The screen requires the `dashboard.view` capability, which every role includes. It refreshes itself every 30 seconds.

The bell's badge counts the alert types active at critical or warning severity. Its menu lists recoveries in progress and up to six active alerts.



Tiles

The **Current attention** panel at the top of the screen opens with a headline, such as how many critical alerts need attention, and holds these tiles:

Tile	Meaning
Critical	Critical findings across objects. Review these first: recovery or data protection is at risk.
Warning	Warning findings across objects. Review the condition and recommended action below.
Could not find out	Facts nothing could confirm, such as an agent that is not connected. An absence of information, counted neither as a fault nor as healthy.
Not monitored	Confirms that the Manager evaluates every alert condition: it shows 0 and **every condition is monitored**. See **What is monitored** below.
Excluded	Objects an operator has marked as out of scope. They are in none of the other counts, and each stays on its own inventory screen with its reason and author.

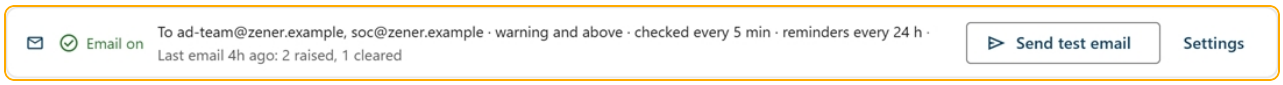
Severity tiles are shown only when their count is not zero.

Alert email

The line below the tiles says whether alerts are emailed, and whether that is working:

- **Email off**–No recipients are set, so nobody is emailed.
- **Email on**–The recipients, the minimum severity (such as **warning and above**), the check interval, and the reminder interval or **no reminders**.
- **Email not working**–The same, with the reason: the mail server is not set up, the last check failed, or the last email was not sent. Alerts are held and sent at the next check once the problem is fixed.
- **Email status unknown**–The console could not read the status from the Manager.

After the last email sent, the line shows when it went and what it listed, such as **Last email 5 hours ago: 2 raised, 1 cleared**. **Settings** opens the **Configuration** screen, where the **Alert notifications** settings are. When email is on and you have the `config.manage` capability (System Admin), **Send test email** sends the recipients an email in the real format, marked `[Test]`, listing what is raised now. It does not change what the next check reports. One test email can be sent every 30 seconds.



For how alert email works, see Alerts Reference; for the settings, see Settings Reference; for `cdr alert notify status` and `cdr alert notify test`, see Alert, Audit, and Report Commands.

Raised now

The **Raised now** card lists active alerts grouped by severity: **Critical**, **Warning**, **Could not find out**, and **Notice**. Each alert shows:

- Its title and how many objects it affects (***n* affected**). An object can appear in more than one alert.
- An origin tag saying which part of the system noticed it: **Backup protection**, **Topology**, **Storage**, **Security**, or **Recovery**.
- What it watches, and the affected objects by name (the first few, then **and N more**).
- Where it applies, a date with a label that says what it is: **Oldest affected restore point** for most alerts, **Earliest removal** for objects removed from the directory, **Longest silent: last replicated** for domain controllers silent past the tombstone lifetime, **First seen** for unexpected SMB connections to the staging share, **First change** for committed vault content changed outside retention, **Journal gap or last report** for vaults not being watched for changes, **First refusal since the last credential issued** for staging that could not be created, **First reset** for a krbtgt second reset due, and **Roles seized** for a recovery protection override.
- What to do, and a link to the screen where you do it, such as **View domain controllers**, **View repository**, **View backup jobs**, **View agents**, **View audit trail** for the staging share and vault change alerts, whose first sighting is an audit trail entry, **View test restores** for the test restore alerts, or **View recoveries** for the recovery plan alerts.

When nothing is active, the card reads **No monitored condition is raising an alert.** Every monitored condition was evaluated and none matched.

For every alert, its severity, what raises it, and how to resolve it, see Alerts Reference.

What is monitored

This card says how many of the alert sources the Manager evaluates are quiet, as ***n* monitored and quiet**. Select **Show the list** to see them: **Monitored and quiet** lists each source that was evaluated and is not currently raised, with what it watches.

Two sources report on the watching itself, so that a quiet storage machine is never mistaken for a watched one: **Storage machines not fully watching the staging share** and **Vaults not being watched for changes**. When the reason is the storage machine's audit policy, for example **Audit File Share is off in this machine's audit policy**, turn the auditing on as described in Turning on auditing on the storage machine. For what the storage agent watches, see Vault Protection and Integrity.

Excluded objects

An operator with the `backup.manage` capability can exclude a forest, domain, site, or domain controller from alerts, with a reason and an optional review date. Excluded objects are left out of every alert and count, but remain visible on their inventory screens. When the review date passes, or the object changes, the exclusion lapses and an alert says so. For how to exclude an object, see Excluding and Retiring Objects.

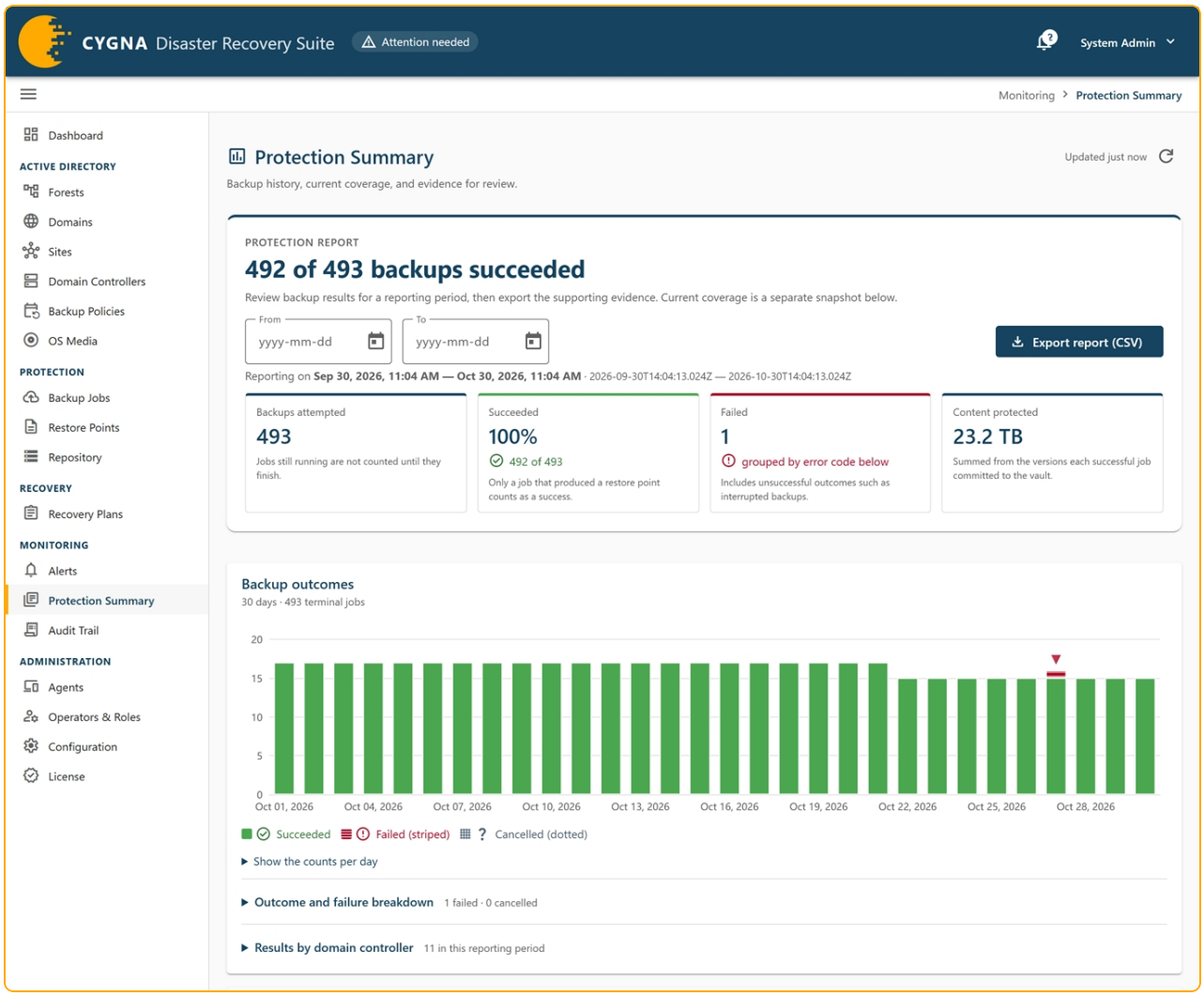
Related topics

- Dashboard
- Coverage and Readiness
- Alert, Audit, and Report Commands
- Settings Reference

Protection Summary

The **Protection Summary** screen gives the backup history, current coverage, and evidence for review: how many backups ran over a period and what they produced, and how well each domain and forest is covered now. You can export what is on the screen as a CSV file, for example as evidence for an audit.

To open it, select **Monitoring** > **Protection Summary**. The screen requires the `reports.view` capability, which every role includes.



Tiles

The **Protection report** panel at the top of the screen opens with a headline such as how many of the window's backups succeeded, and holds the report window, the export button, and these tiles:

Tile	Meaning
Backups attempted	Jobs that finished in the window. Jobs still running are not counted.
Succeeded	Jobs that produced a restore point, with the success rate. The tile is green at 100%, amber at 90% or more, and red below 90%.

Tile	Meaning
Failed	Jobs that failed, including unsuccessful outcomes such as interrupted backups, grouped by error code in the card below. Shown only when there are failures.
Content protected	The total size committed to the vault by successful jobs in the window. Shown only when something was committed.

Backup outcomes

This card reports on a window of time. Set **From** and **To** in the **Protection report** panel at the top of the screen to choose it, or select **Default window** to return to the last 30 days. The window can be at most 366 days. The **Protection report** panel shows the window in both local time and UTC.

- **Chart**–Succeeded, failed, and cancelled backups per day. Select **Show the counts per day** for the same figures as a table, by UTC day.
- **Outcomes by status**–Jobs per final status. Expand **Outcome and failure breakdown** to see this table and the next.
- **Failures by error code**–The recorded reasons for unsuccessful backups, grouped by the error code the Manager recorded, not by tool text, so they can be counted. Failures recorded without a code are shown as **Uncoded** rather than dropped.
- **Results by domain controller**–Expand this section, which shows how many domain controllers had results in the window. For each domain controller: **Succeeded**, **Failed**, **Cancelled**, **Protected** (size committed), and **Last success**.

For what each error code means, see Backup Job Errors.

Coverage and freshness

This card is a snapshot of coverage as it stands now, not a history over the window.

- **Per domain**–**Domain**, **Coverage**, **DCs**, **Protected** (with the numbers that **need a backup policy** and are **rebuilt on recovery** beneath it, when there are any), **Full system recovery options**, **Recovery point**, and **Judged against** (the freshness window the recovery point is measured against).
- **Per forest**–**Forest**, **Readiness**, **Domains recoverable** (could be recovered today, including domains with a finding against them), **Domains covered** (recoverable and meeting every coverage rule), and **DCs protected**. **DCs protected** counts protected domain controllers against the backup targets, so domain controllers rebuilt on recovery are named beside the figure rather than counted as a shortfall. Those a domain needs backed up because it has too few backup targets stay in the total and

are named as **need a backup policy**. Domains and domain controllers excluded by an operator are named the same way, for example `2 of 4 · 1 need a backup policy · 4 rebuilt on recovery · 1 excluded`, and are left out of the total.

For the coverage and readiness verdicts, see Coverage and Readiness.

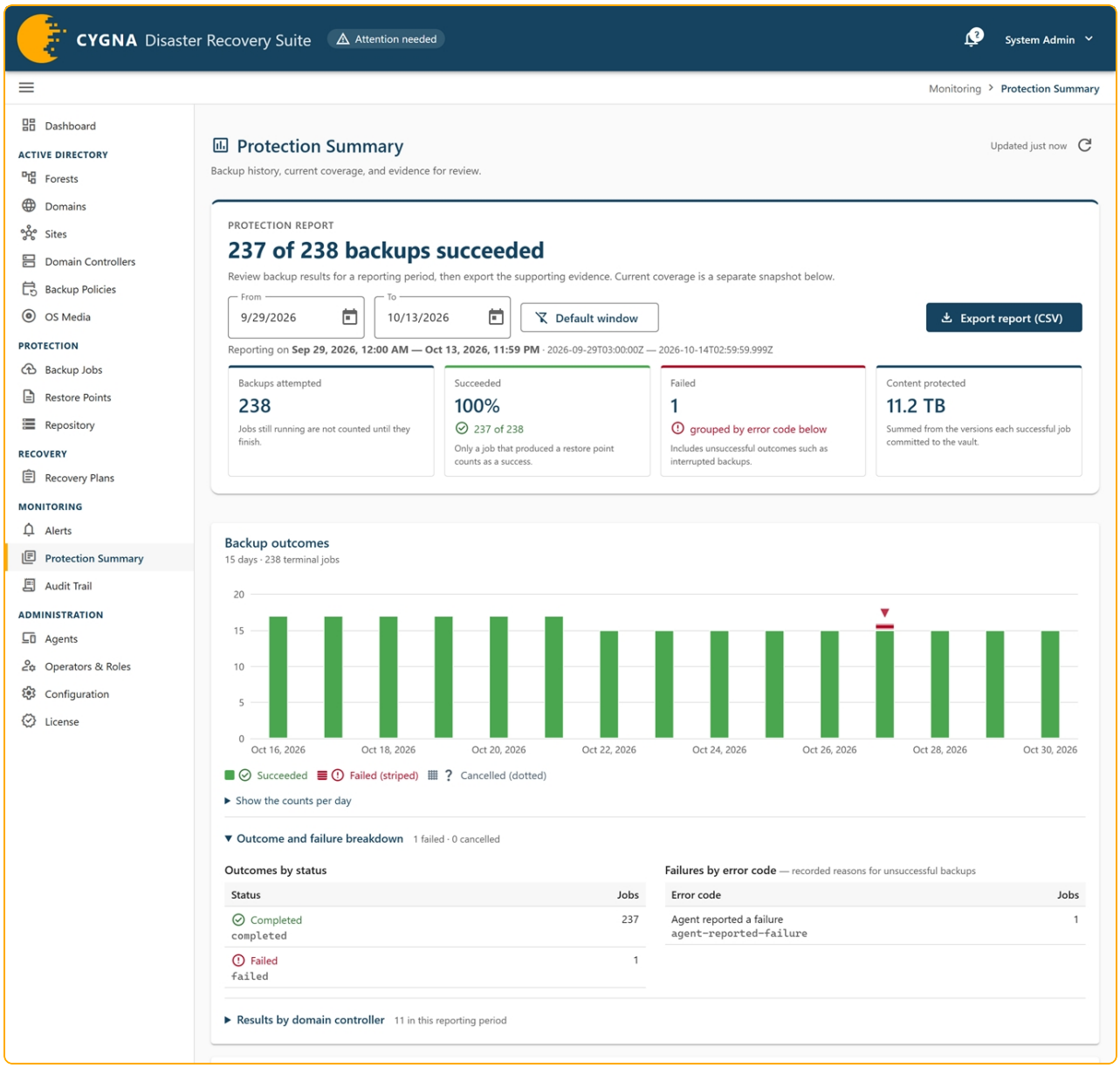
Recovery history

Restores and recoveries are recorded in the audit trail, with their gate results and each operator acknowledgment, and every run and plan is listed on the Recovery Plans screen. The card links to both: **View audit trail** and **Recovery plans**.

Exporting the report

Select **Export report (CSV)** to download what is on the screen as a CSV file named `cygnadr-report-<from>-to-<to>.csv`, with the window's UTC dates in the name. The file is UTF-8 and every timestamp is given in both UTC and local time. It contains these sections:

Section	Contents
Report header	The window start and end, and when the report was generated.
Backups per day	Day, Succeeded, Failed, Cancelled.
Outcomes by status code	Status code, its wording, and the count.
Failures by error code	Error code, its wording, and the count.
Per machine	Machine, Succeeded, Failed, Cancelled, Bytes protected, Last succeeded.
Domain coverage	As at generation: Domain, Forest, coverage code and wording, DCs, Protected, Need a backup policy, Rebuilt on recovery, first-DC candidates, full system recovery capable, recovery point.
Forest readiness	As at generation: Forest, readiness code and wording, Domains, Recoverable, Covered, DCs, Protected, Domains excluded, DCs excluded, DCs needing a backup policy, DCs rebuilt on recovery.



Related topics

- Audit Trail
- Backup Jobs
- Alert, Audit, and Report Commands

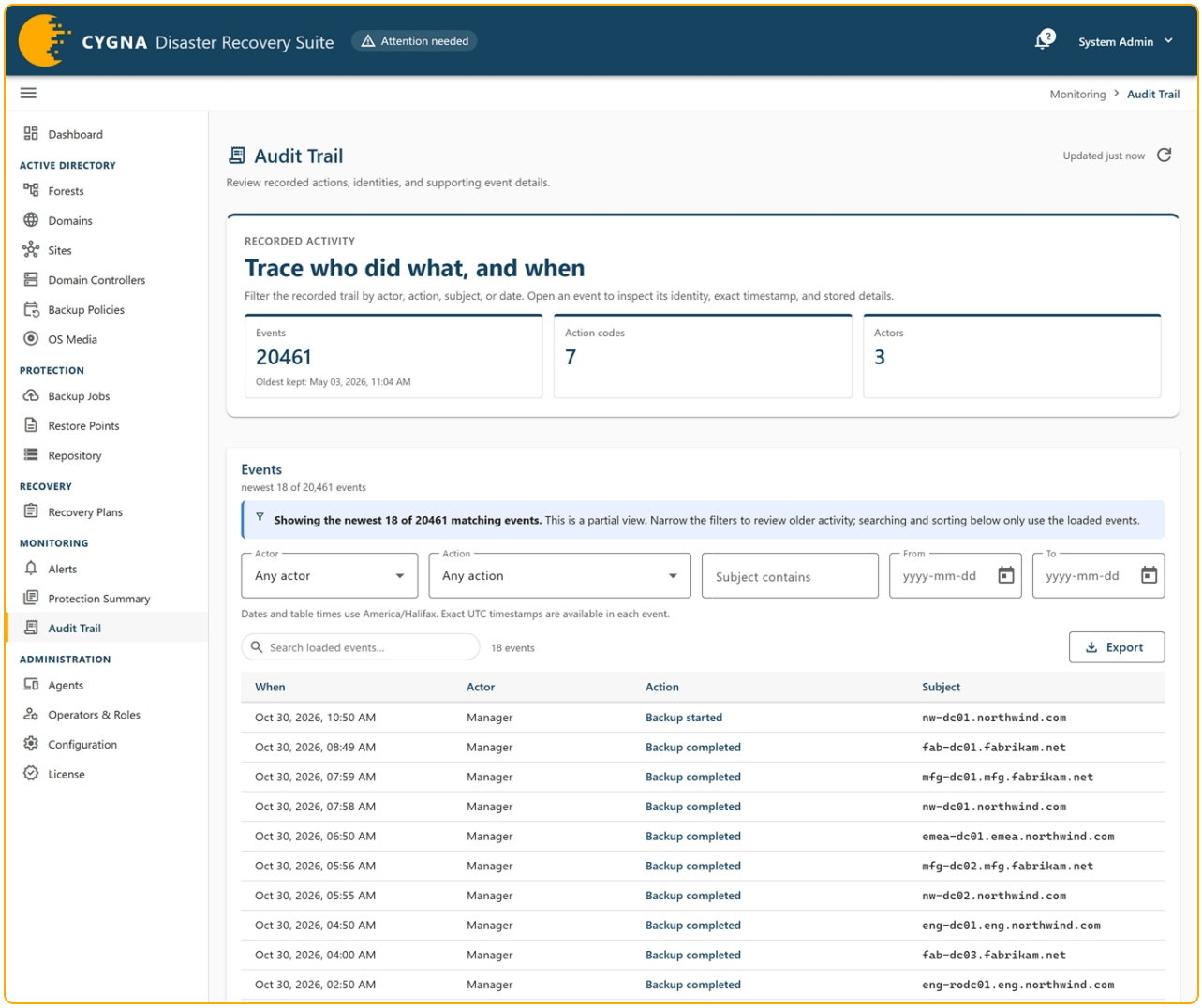
Audit Trail

The **Audit Trail** screen lists every gate result and state change the Manager has recorded, with who or what caused it. The Manager writes an audit entry for events such as an agent enrolling, a backup starting, completing, or failing, a policy or setting changing, an operator

signing in or changing roles, and every step of a restore or recovery, including each approval and acknowledgment.

To open it, select **Monitoring** > **Audit Trail**. The screen requires the `audit.view` capability, which every role includes. It changes nothing, and it refreshes itself every 30 seconds.

For what is audited and how long entries are kept, see Auditing Operator Activity.



Tiles

- **Events** (or **Events matching** when filtered)–How many entries there are, whether all of them are shown, and the date of the oldest entry kept.
- **Action codes**–How many distinct kinds of action appear across the whole trail.
- **Actors**–How many distinct actors appear across the whole trail.

Filters

The filters on the **Events** card are applied by the Manager, which returns at most the newest 200 matching entries. To find older entries, narrow the filters.

Filter	Description
Actor	One actor, or **Any actor**.
Action	One kind of action, shown with its label and code, or **Any action**.
Subject contains	Text the entry's subject must contain, such as a machine name.
From and **To**	A date range.

Select **Clear filters** to remove them all. The search box narrows only the entries already returned; to ask the Manager for a different set, change the filters. If the Manager did not apply a filter as requested, a **Filter not applied** notice warns that the rows are not narrowed the way the controls say.

Events

The table has four sortable columns:

Column	Description
When	When the entry was recorded.
Actor	The operator's username, or **System**, **Manager**, **Storage agent**, or **Unauthenticated caller**.
Action	What happened, for example **Restore approved**, **Configuration changed**, or **Sign-in failed**. An action the console has no wording for is shown as its code.
Subject	The object the action applied to, such as a machine, policy, plan, or operator.

Audit entry detail

Select a row to open the entry's detail sheet. It shows **Recorded (local)** and **Recorded (UTC)**, **Actor** (with the stored identity beneath it), **Subject**, **Action code**, and the entry number. The **Fields** section lists the entry's detail as key and value pairs, such as the reason an operator gave. The **Detail as stored** section shows the same detail exactly as recorded. It starts collapsed for each entry you open; select it to expand it. When the detail could not be read into fields, it starts expanded, because it is all there is.

Exporting the audit trail

Export, beside the search box on the **Events** card, saves the audit trail as a file to keep or send. Exporting needs only the `audit.view` capability, so every role can export.

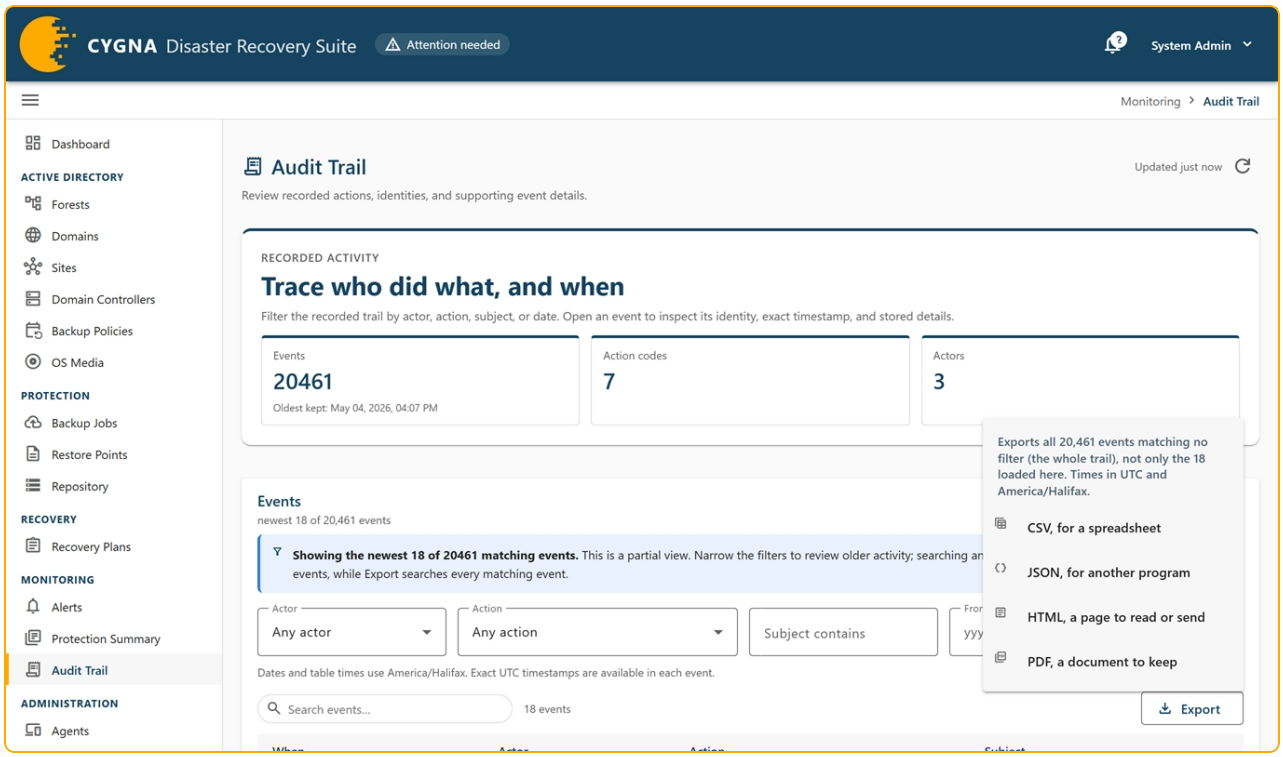
The Manager makes the file from every entry that matches the filters and the search box, across the whole trail–not only the 200 entries loaded on the screen. Before you choose a format, the menu says what the file will hold, for example **Exports all 4210 events matching the filters, not only the 200 loaded here. Times in UTC and Europe/Berlin.** The time zone is your browser's.

To export the audit trail:

1. Set the filters, and a search if needed, to match the entries you want.
2. Select **Export**.

   The menu shows what the export holds.
3. Select a format.

   The button reads **Exporting…** until the file is made, and your browser then saves it. You can export one format at a time.



Format	At most	What you get
CSV, for a spreadsheet	100,000 entries	One header row and one row per entry: **Event ID**, **Time (UTC)**, the time in your time zone, **Actor**, **Actor code**, **Action**, **Action code**, **Subject**, and **Detail**. UTF-8 with a byte order mark, so Excel reads it correctly. A value that starts with `=`, `+`, `-`, `@`, a tab, or a carriage return is written with a leading apostrophe, so a spreadsheet shows it as text instead of running it as a formula; Excel does not display the apostrophe. The filters are not written into the file.
JSON, for another program	100,000 entries	When and by whom the file was exported, the Manager, the filters and search, the time zone, the number of entries, and the entries themselves.
HTML, a page to read or send	5,000 entries	A page titled **Audit trail export** that says when, by whom, and from which Manager it was exported, lists the filters and search, and then the entries with their time, actor, action, subject, and detail. Times are shown in your time zone with UTC beneath.
PDF, a document to keep	5,000 entries	The same document as the HTML page, on A4 landscape pages. The Manager makes it with its own PDF renderer.

The file uses the console's wording for actors and actions, with the codes beside them. It takes the Manager's name for it, `cygnadr-audit-<yyyyMMdd-HHmmss>.<format>`, from the UTC time of the export.

If more entries match than the format holds, nothing is exported, so you never get a truncated file. A **Not exported** notice gives the Manager's reason, for example **More than 5,000 events match, and a PDF export holds at most 5,000. Narrow the dates or the filter, or export CSV or JSON, which hold up to 100,000.** The same notice reports a PDF the Manager could not make. Narrow the filters or the dates and export again.

Each export is recorded in the audit trail as **Audit trail exported** (`audit.exported`) before the file leaves the Manager. The entry's subject is the file name, so you can match a saved file to the record of its export.

To export from a script, use `cdr audit export`. See Alert, Audit, and Report Commands.

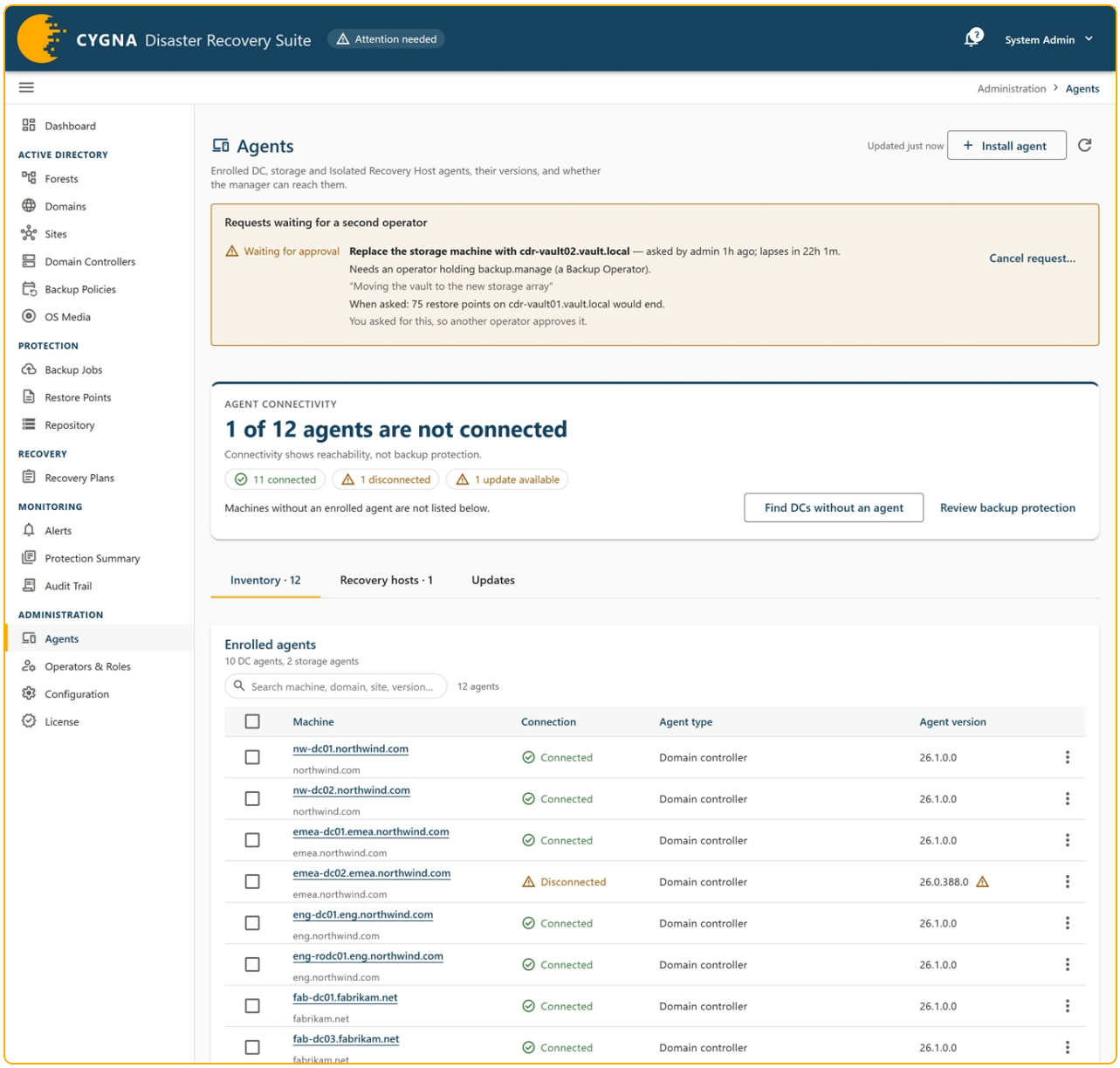
Related topics

- Protection Summary
- Settings Reference

Agents

The **Agents** screen lists every enrolled DC agent, storage agent, and Isolated Recovery Host agent, the version each runs, and whether the Manager can reach it. It shows whether each Isolated Recovery Host is ready to run test restores and whether it is approved for them, says what automatic agent updates are doing, gives you the command lines and packages for installing an agent by hand, and lets you update, restart, and uninstall agents of every kind from the console. A storage agent that is not the storage machine can be made the storage machine from here.

To open it, select **Administration** > **Agents**. Viewing the screen requires the `environment.view` capability, which every role includes. Installing, updating, restarting, and uninstalling agents, and downloading agent packages, requires `agents.manage`, held by the System Admin role and the break-glass account. Without it, the selection checkboxes, row menus, download buttons, and update and restart buttons are not shown. Approving and draining an Isolated Recovery Host also need `agents.manage`. Replacing the storage machine needs both `agents.manage` and `backup.manage`; an operator holding either one can ask for it, and the row menu of a candidate storage agent is shown to them. See Moving the Storage Machine or an Isolated Recovery Host.



Layout

The **Agent connectivity** panel sits at the top of the screen. Below it, three tabs divide the rest:

- **Inventory · *n***–Every enrolled agent. This is the tab the screen opens on. See Inventory tab.
- **Recovery hosts · *n***–Each Isolated Recovery Host, whether it can run test restores, and what it is doing. The tab shows ***n* need attention** while a host is not ready, cannot be read, or is awaiting approval. See Recovery hosts tab.

- **Updates**–What automatic agent updates are doing, the agent package the Manager holds, and the install commands. The tab shows ***n* to review** while agents need an operator. See Updates tab.

The tab you choose is kept in the page address, so a refresh or a shared link opens the same tab. The page header holds **Install agent**, which opens the package line and the install commands in a panel at the side of the screen, and a refresh button.

When no agent is enrolled

Before any agent enrolls, the screen shows no tabs and reads **No agents are enrolled.** It says which agent goes where: the DC agent on a domain controller, the storage agent on the backup host, and the Isolated Recovery Host agent on the Hyper-V host that backups are verified on. It also gives the address the Manager answers on, which you pass as `MANAGERADDRESS`. Each agent enrolls itself using the Manager address, CA fingerprint, and enrollment token it is installed with.

Below it, the **Agent packages** card holds the **Install an agent manually** section, already expanded, with the install command and package download for each agent, filled in with the Manager's address and CA fingerprint. If the package folder cannot be used, the card also shows the package line saying why. See Installing an agent by hand.

Agent connectivity panel

The **Agent connectivity** panel sums up the connections in a headline, such as "*n* of *m* agents are not connected" or "*n* of *m* enrolled agents connected". A connected agent means the Manager can reach it, not that its machine has a usable backup. The panel's filter chips select agents on the Inventory tab that are **connected**, **disconnected**, **never reported**, or have an update available (***n* updates available**). While agents need an operator after an automatic update, ***n* update outcomes to review** opens the Updates tab. Machines without an enrolled agent are not listed on this screen; select **Find DCs without an agent** to open the Domain Controllers screen filtered to them, or **Review backup protection** to open it unfiltered.

Inventory tab

The **Enrolled agents** card lists every agent. Use the search box to find agents by what the grid shows: machine, domain, version, or a chip such as **Candidate** or **Not reported as a domain controller**.

Candidate machines, which run the DC agent but are not domain controllers, are listed here like any other DC agent, with **Not reported as a domain controller** under the machine name. Hover over it, or focus it with the keyboard, for why: a member or standalone server, such as one kept for a full system recovery to restore a domain controller onto. The agent reports the same when it could not read the machine's directory role, so check any server you expect to be a domain controller. Candidate machines are not listed on the Domain

Controllers screen and are not counted in coverage or alerts. A DC agent on the storage machine or on an Isolated Recovery Host is never offered as a candidate machine.

Column	Description
Machine	The machine the agent runs on, with a DC agent's domain beneath it. Select the name to open the agent's details. Hover over it to see when the agent record was first enrolled; a reinstall or a new enrollment keeps the record and that date.
Connection	**Connected**, **Disconnected**, or **Never reported**.
Agent type	**Domain controller** for a DC agent, **Storage** for a storage agent, or **Isolated Recovery Host** for an Isolated Recovery Host agent. A chip beside a storage agent marks a candidate as **Candidate, given no work** and a replaced one as **Replaced**; while either is listed, the storage machine is marked **Storage machine** so the rows can be told apart. With a single storage agent, it has no chip. A chip beside an Isolated Recovery Host agent marks it **Awaiting approval** or **Draining**; an approved host has none. Hover over the chip for what it means.
Agent version	The installed agent version. A warning icon means the Manager holds a newer package for this kind of agent; hover to see the version (**Update available: *version***). For a connected agent, hovering over the version shows what its session says it can do (**Its session says it can: …**).

Select a machine name to open its **Agent details** panel at the side of the screen: the connection, **Agent type**, **Role** for a DC agent not reported as a domain controller, **Domain** and **Site** (not applicable to a storage agent or an Isolated Recovery Host agent, which are not domain-joined), **Operating system**, **Build and revision**, **Agent version**, **Available version**, **Last heartbeat**, **Enrollment**, the storage machine or host **Designation**, and the **Capabilities** its session reports.


AGENT DETAILS
nw-dc01.northwind.com
Connected
Agent type
Domain controller
Domain
northwind.com
Site
London-HQ
Operating system
Windows Server 2025
Build and revision
26100.2314
Agent version
26.1.0.0
Available version
26.1.0.0
Last heartbeat
59s ago
Enrollment
First enrolled May 16, 2026, 04:01 PM (local) · 2026-05-16T19:01:53.618Z
Capabilities
Its session names no capabilities

An agent shows as disconnected after it misses a few heartbeats. For causes and fixes, see Agent Enrollment and Connectivity.

The Manager keeps one record for each machine name and agent kind. When a recovery rebuilds a domain controller onto another machine, the replaced machine's record is removed. To remove the record of an agent whose machine is gone, select **Forget this agent record…** in its row menu, or use `cdr agent forget`. The menu item needs `agents.manage` and is offered only when nothing stops the deletion; the Manager checks again when you confirm. See Maintaining Agents.

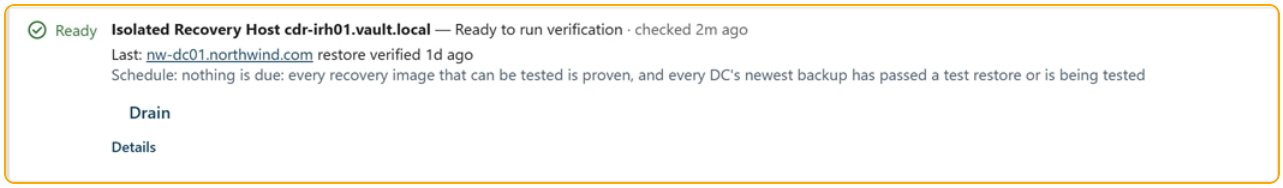
The storage machine and candidates

The first storage agent to enroll is the storage machine, the one every backup is written to. A storage agent that enrolls after that is a candidate and is given no work. If it is meant to replace the storage machine, open its row menu and select **Replace the storage machine with *machine*…**. The dialog says how many restore points end, what else changes, and how the two machines' locally set retention values compare, and asks for the candidate's agent ID. An operator holding both `agents.manage` and `backup.manage` replaces the storage machine with **Replace the storage machine**; one holding only one of them selects **Ask for the replacement**, and an operator holding the other approves it. If nobody meant to install the candidate, uninstall it and delete its record. See Moving the Storage Machine or an Isolated Recovery Host.

While a replacement waits for its second operator, a **Requests waiting for a second operator** card at the top of the screen names it, who asked, when it lapses, and which capability approving it needs, with **Review and approve…**, **Decline…**, and **Cancel request…** for the operators who may use them. The **Repository** screen shows the same card.

Recovery hosts tab

The **Recovery hosts** tab shows a panel for each Isolated Recovery Host: whether it can run test restores, and what it is doing. The panels refresh every 30 seconds. With no host enrolled, the tab reads **No recovery hosts reported.**



- **Verdict**—**Ready**, **Not ready**, or **Unknown**, or **Test restore running** while one runs on the host. A host is not ready while anything blocks it, and unknown while something could not be read.
- **Headline**—**Ready to run verification**, **Running verification**, **Not connected to the manager**, **Has not reported whether it can run verification**, or the problem that blocks the host, followed by when it last checked.

- **Findings**–Each problem and warning, with what to do about it. Information findings, such as where test restore virtual machines are stored and which network they restore over, and the host's own notes are under **Details**.
- **Test restores**–The test restore running now (**Testing *machine* now**, scheduled or started by an operator), the last one and its result (**Last:**), and what the schedule last decided for this host (**Schedule:**): the test restore it started here, why it could start nothing here (for example that the host is busy, awaiting approval, or draining), that it is waiting for running backups, or that nothing is due. With more than one host, each host's line is its own. A host the schedule has not looked at yet says so; the schedule looks every five minutes.
- **Approval**–A host that is **Awaiting approval** is given no test restores, however ready it is. With `agents.manage`, select **Approve…** and confirm with **Approve the host**. An approved host offers **Drain**, which stops new test restores going to it while the one running finishes; it then reads **Draining**, and **Drained** once nothing runs on it, when you can uninstall its agent and delete its record. **Use it again** puts a draining host back in use. See Moving an Isolated Recovery Host.

The host checks itself when it connects and at the **Readiness check interval**. These are the problems it reports most often:

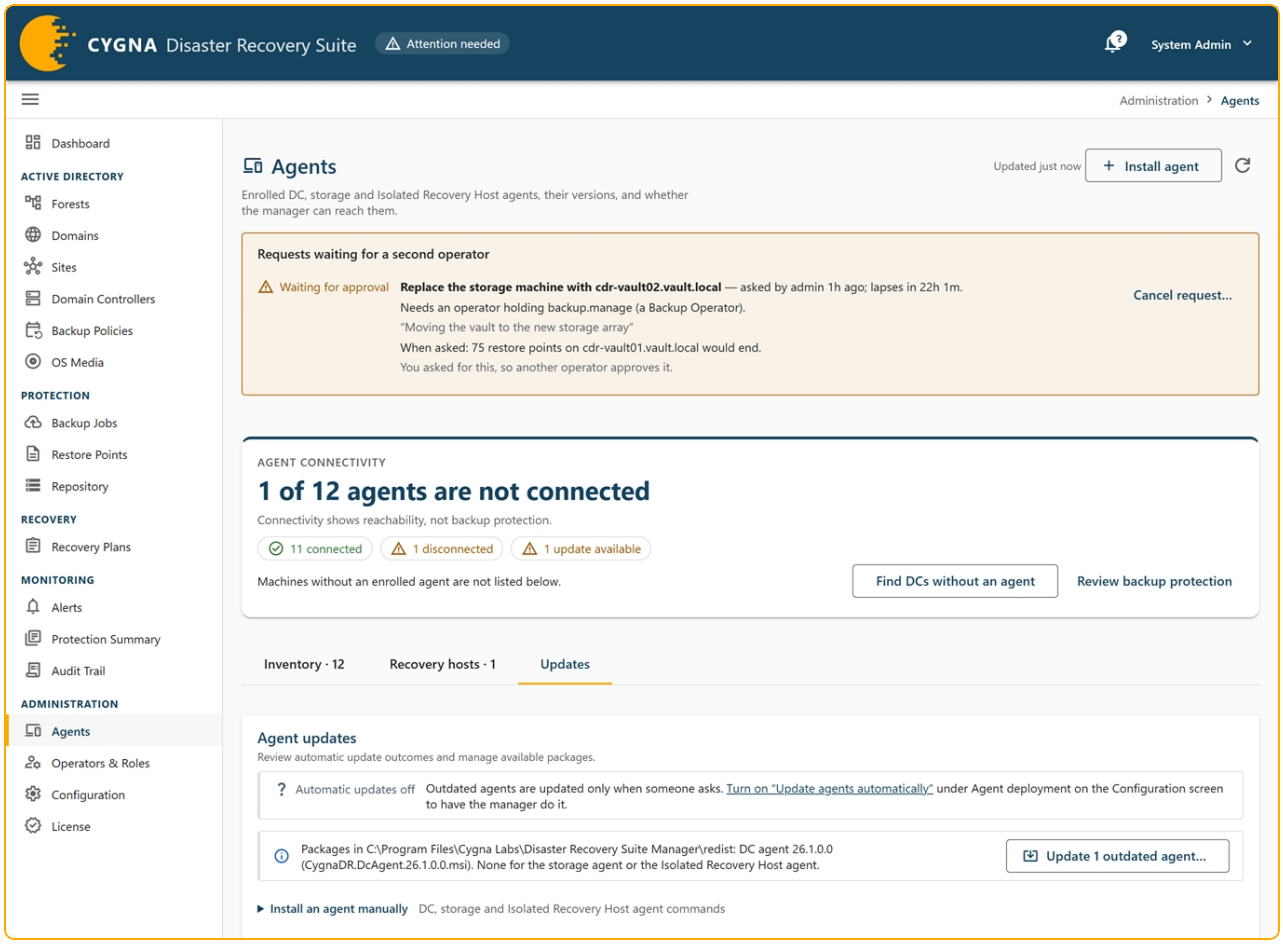
Finding	What to do
Hyper-V is installed and waiting for a restart	The agent has scheduled a restart, which `shutdown /a` cancels within five minutes, or it says why it will not restart the host itself. Restart the host when it suits you; it reports again on its own. A role that Windows reports installed before the Hyper-V service appears is also waiting for this restart.
Hyper-V is not installed	With **Install Hyper-V** on, the agent installs it. Otherwise, install the Hyper-V role with its management tools and restart. When the agent does not install it, its service log says why.
This host is a VM without nested virtualization, or **The processor does not offer Windows hardware virtualization**	On a virtual machine, turn on nested virtualization while it is shut down, for example with `Set-VMProcessor -VMName <vm> -ExposeVirtualizationExtensions $true` on Hyper-V. On a physical machine, turn on Intel VT-x or AMD-V in the firmware.
The host is joined to *domain*, or **The host is a domain controller**	Use a host that is not a member of any domain.

Finding	What to do
The agent could not create the recovery network	The finding says why, for example that the **Recovery network prefix** overlaps an address or a NAT already on the host, or that another switch is named **CygnaDR Recovery**. Choose another prefix, or rename the other switch.
The folder for verification VMs cannot be used, or **The host has no drive that can hold virtual disks**	Change **Folder for verification VMs**, or leave it empty for the agent to choose. Virtual disks need an NTFS or ReFS drive.
Low disk space or memory (warnings)	The host warns below 100 GB free where test restore virtual machines go, and below 8 GB of memory. A small host may not be able to start a restored domain controller.

A host that shares a machine with the storage agent or the Manager says so under **Details**; that arrangement is supported. For the settings named here, see the Isolated Recovery Host agent settings in Settings Reference. For setup, see Setting Up an Isolated Recovery Host.

Updates tab

The **Agent updates** card on the **Updates** tab holds, from the top, the automatic updates line, the package line, and **Install an agent manually**. Updating itself is started from here or from the Inventory tab; see Updating agents.



Automatic updates

The first line says whether the Manager updates agents automatically. While **Update agents automatically** is off, the default, the line reads **Automatic updates off** and links to the Configuration screen where you turn it on. While it is on, the line reads **Automatic updates on** and counts the agents updating, queued, and waiting for a later check, or says **Every agent is up to date.**, followed by when the Manager last checked. Hover over the time for the interval, the next check, and how many agents are updated at once. With `agents.manage`, **Check now** runs a check at once.

Agents that need an operator, because they cannot update themselves or their last automatic update failed, are listed beneath the line, each with its versions, why, and when a failed update may be retried. If the inventory already reports the version the update was going to, the entry says so: it is the previous outcome, and the next check reconciles it. See Updating agents automatically.

Package line

The package line names the package folder and each agent package it holds, with its version and file name, and says which kinds of agent have none. While the install lines are

shown and no agent is outdated, the line is left out, because the install lines already carry each package and its version. It is always shown when the folder has a problem or an agent is outdated. The package folder is set by the **Package folder** setting on the **Manager** tab of the Configuration screen, and defaults to the `redist` folder in the Manager's program folder. If the folder is not set, does not exist, or holds no DC agent package, the line says so and links to the Configuration screen; agents cannot be installed or updated from the console until it is fixed.

The package folder also holds the storage agent and Isolated Recovery Host agent packages, which the Manager uses for downloads and for updating those agents. When some agents are older than the package for their kind, the line offers **Update *n* outdated agents…**, which opens the deploy dialog for all of them that can be updated from the console. Update DC agents before any recovery plan runs: the Manager does not send recovery plan steps to an agent that is behind the package.

Installing an agent by hand

Select **Install agent** in the page header to open the install commands and packages in a panel at the side of the screen. They are also under **Install an agent manually** on the Updates tab, and, before any agent is enrolled, on the **Agent packages** card, already expanded. Each agent's ready-made `msiexec` command line sits under the agent's name and icon (**Domain controller agent**, **Storage agent**, **Isolated Recovery Host agent**), filled in with this Manager's address and CA fingerprint. Select **Copy command** to copy one, then run it on the agent's machine, replacing the placeholders:

```
msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
MANAGERCAFINGERPRINT=<fingerprint> STORAGEROOT=<path> ENROLLMENTTOKEN-
N=<token>
msiexec /i CygnaDR.HostAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
```

With `agents.manage`, each line also has a download button that saves that agent's package from the Manager. Hover over it to see the package version and its SHA-256 hash. You can also save a package with `cdr agent package download`.

- **Manager address**–An IP address rather than the Manager's name, because the Manager is usually not domain-joined and so not in the agents' DNS. If the Manager has other addresses, **Details** lists them, to put in `MANAGERADDRESS` instead when the agent's machine cannot reach the one in the command.
- **CA fingerprint**–Identifies the Manager's certificate authority; the agent connects only to a Manager that presents it. The fingerprint is public, not secret. Before relying on it, compare it with a second source; see Certificates and Trust.

- **Enrollment token**–Not shown, because it is a secret. Use the token the Manager was installed with.
- **Storage root**–For the storage agent, the folder on the storage machine that holds staging and the vault.

These values are needed on a first install only; an upgrade keeps the values already there. A first install of the storage agent or the Isolated Recovery Host agent is always by hand, because only the DC agent can be pushed from the console. Once installed, any agent updates itself from the Manager. For full instructions, see Deploying DC Agents, Setting Up the Storage Machine, and Setting Up an Isolated Recovery Host.

Updating agents

Agents of every kind are updated from this screen. The Manager chooses how for each agent:

- **Self-update**–An agent that is connected and able to update itself is sent the package over its own connection to the Manager. It installs the package only after checking that it is signed by Cygna Labs, that it is the package for its own kind of agent, and that it is newer than the version it runs. No credential is needed, and nothing connects into the machine.
- **Push install**–A DC agent that cannot update itself is installed remotely from the Manager, with an administrator account you supply, as for a first install.
- **By hand**–A storage agent or Isolated Recovery Host agent that cannot update itself must be installed by hand, with the package from its install line.

An agent cannot update itself when it is not connected or when its machine's policy turns updates from the Manager off. For that policy, see Maintaining Agents.

To update agents:

- On the Updates tab, select **Update *n* outdated agents…** on the package line to update every agent that is behind the package for its kind, except those that must be installed by hand.
- On the Inventory tab, select the checkboxes of the agents you want, and then select **Update 1 agent…** or **Update *n* agents…** in the bar at the bottom of the screen. The count is of the selected agents that can be updated from the console. A checkbox is offered for every agent that can update or restart itself, and for every DC agent.
- Use a row's menu: **Update the agent on <machine> to <version>…**, or, for a DC agent that is already current, **Reinstall the agent on <machine> (<version>)…**. For an agent that must be installed by hand, the menu says so and points to **Install agent**.

To install the agent on a domain controller that has none yet, use **Install agent…** on the Domain Controllers screen.

The Manager refuses to update an agent while it is busy: a domain controller while it is being backed up or restored, or while a recovery plan is active in its forest; the storage agent while backups or a restore are using it; an Isolated Recovery Host agent while a test restore runs on its host. The storage agent also declines an update itself while it builds a recovery image or reads a backup to check it. Update it when that work has finished. Agents that share a machine, such as a storage agent and an Isolated Recovery Host agent on one server, are updated one after another, never at once.

Restarting agents

A setting marked **Restart required** on the Configuration screen is saved at once but comes into force only when the agent's service restarts. You can restart agents from the console instead of signing in to each machine. Restart is offered for a connected agent whose session says it can restart its own service.

- To restart one agent, open its row menu and select **Restart the agent on *machine*...**.
- To restart several, select their checkboxes, and then select **Restart 1 agent...** or **Restart *n* agents...** in the bar at the bottom of the screen. The count is of the selected agents that can be restarted.

The dialog says that each agent's service stops and starts again and reconnects in a few seconds, and that settings saved since it last started come into force. Select **Restart the agent** or **Restart *n* agents**. A message then says which agents are restarting, and names each one that was not restarted with the reason. If any agent was not restarted, the message stays on screen until you dismiss it.

The Manager does not restart an agent whose work a restart would cut short:

- A DC agent while a backup of its domain controller or a restore involving it is running, or while a recovery plan is running.
- The storage agent while any backup is running or any restore is reading from its vault.
- An Isolated Recovery Host agent while a test restore is running on its host.

An agent that is not connected starts on the saved settings whenever its service next starts. The Configuration screen offers the same restart for every agent waiting on one. Each restart request is recorded in the audit trail.

Uninstalling an agent

For a connected agent that can remove itself, the row menu offers **Uninstall the agent from *machine*...**. It is not offered for the storage machine, which is replaced rather than removed. The agent can remove itself when it runs as its service, was installed by its own package, and its machine's policy allows changes from the Manager (the same `AllowRemote` policy that allows updates; see Maintaining Agents).

The dialog says that nothing on the machine is backed up, restored, or reported until an agent is installed again. To confirm, enter the machine's name and select **Uninstall the agent**. The Manager refuses, and the dialog shows why, in these cases:

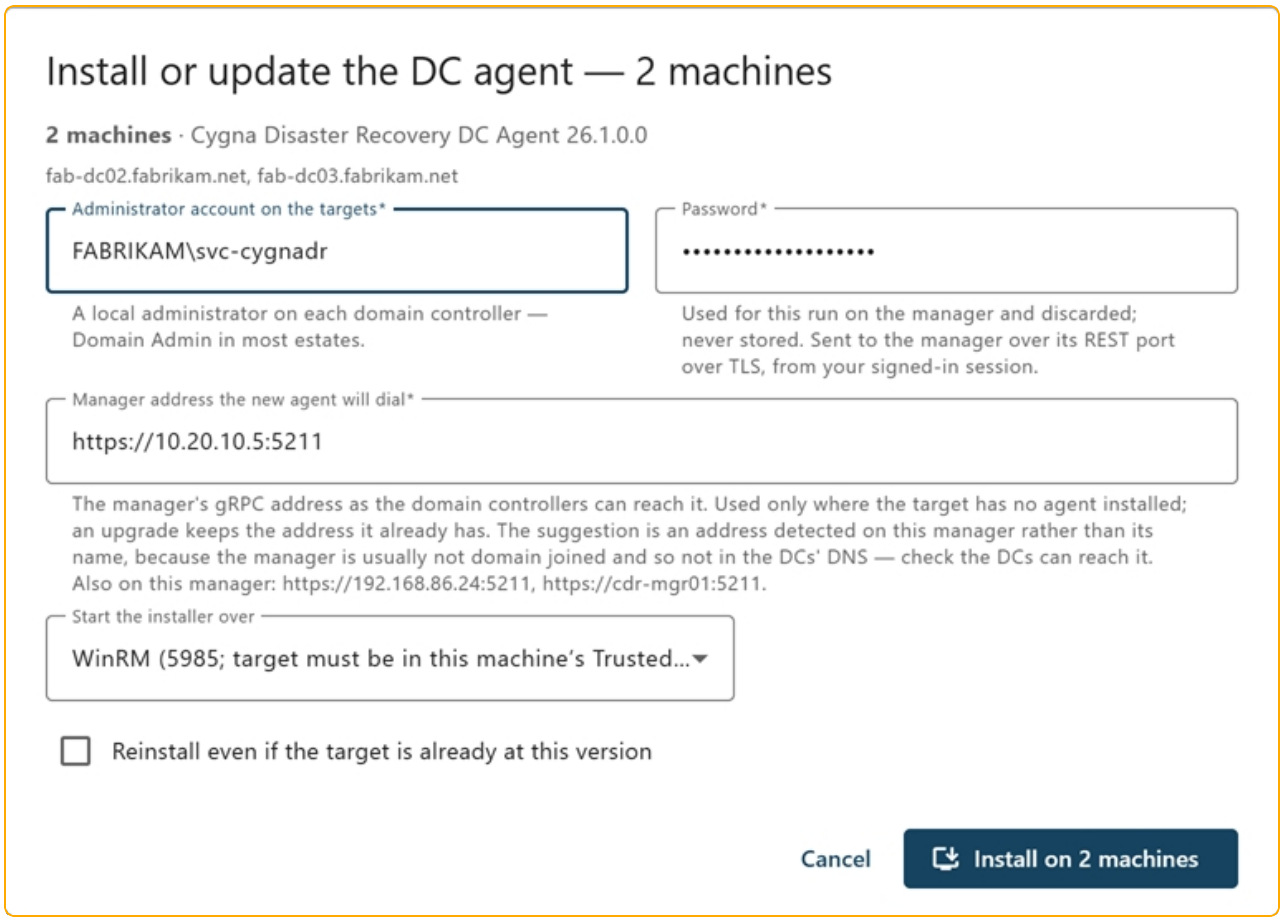
- The agent is on a domain controller that is neither retired nor excluded. Retire the domain controller if it has been demoted or removed, or exclude it with a reason if it is deliberately out of scope, and then uninstall. See Excluding and Retiring Objects. A DC agent on a candidate machine is not held back.
- The agent has work in flight, by the same rules as a restart.
- The agent itself refuses; the dialog gives its reason.

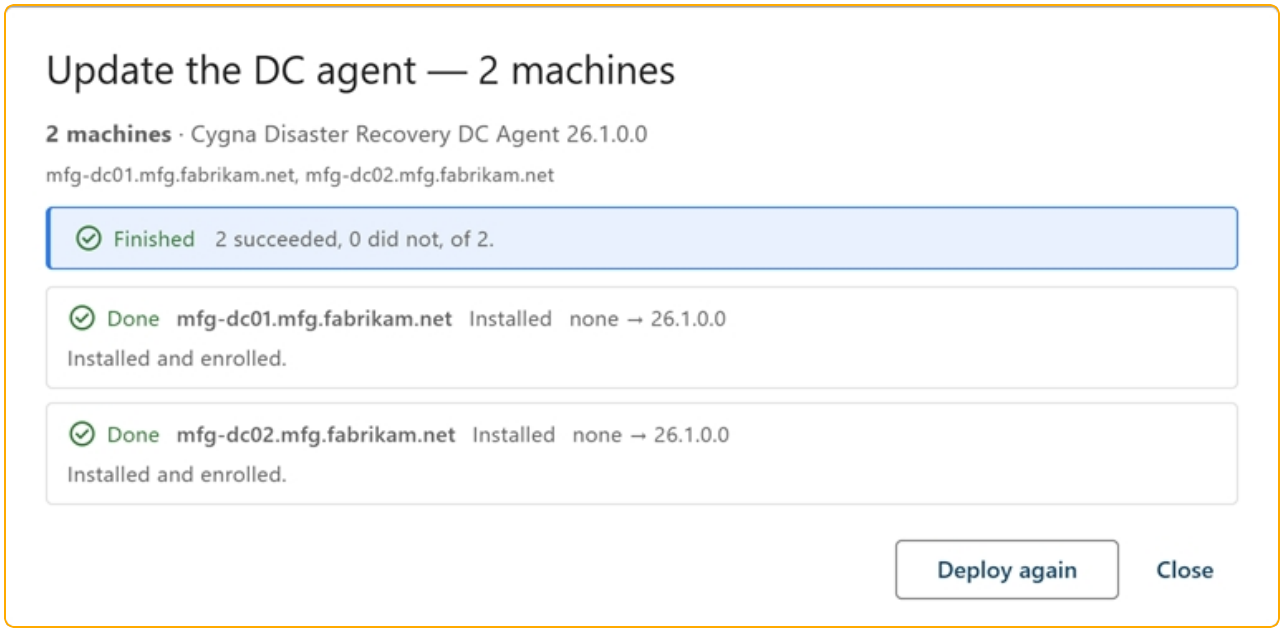
Once accepted, the agent removes itself in a few seconds and does not reconnect. Its record stays on this screen, shown as disconnected; once the agent has gone, delete it with **Forget this agent record…** in its row menu, or with `cdr agent forget`. Each uninstall request, accepted or refused, is recorded in the audit trail.

Deploy dialog

The deploy dialog updates one or more agents, or installs the DC agent on domain controllers. Its title names the targets, for example **Update the storage agent — <machine>** or **Update *n* agents**, and it shows the package version being deployed.

When every target updates itself, the dialog asks for nothing: it explains that each agent is sent its package over its own connection, and the button reads **Update on this machine** or **Update on *n* machines**. When any target needs a push install, the dialog also asks for the fields below.





For a push install, the dialog asks for an administrator account on the targets and its password, the Manager address a new agent will dial, and whether to start the installer over WinRM or DCOM. For each field, see Deploying DC Agents.

Select the button to start. The dialog then shows the run's progress: how many targets have finished and, for each target, its status, the agent's last line of output, the version before and after, and the transcript. For a push install, it also shows the path of the installer log left on the target. For a self-update, the transcript includes a summary of the installer log, and the full log stays in the `update` folder in the agent's data folder. You can select **Close — the run continues** to close the dialog while the deployment runs. When it finishes, the dialog reads **Finished** or **Finished with failures**.

Outcome	Meaning
Installed	The agent was installed on a machine that had none.
Upgraded	An older agent was updated.
Reinstalled	The same version was installed again.
Already current	The target already had this version; nothing was done.
Refused before starting	The Manager did not attempt the install on this target, or the agent refused the package. The message says why, for example that the agent is busy, cannot update itself and must be installed by hand, or already runs a newer build.
Unreachable	The Manager could not connect to the target.
Credential refused	The target did not accept the account.
Install failed	The installer ran and failed. Check the installer log on the target. A failed self-update rolls back, and the agent reconnects at its previous version.

If the request itself fails, the dialog says that whether anything started on the targets is unknown; check the audit trail before retrying. For the remote management prerequisites (ADMIN$, Remote Registry, WinRM or DCOM), see Accounts and Permissions and Ports and Protocols.

Related topics

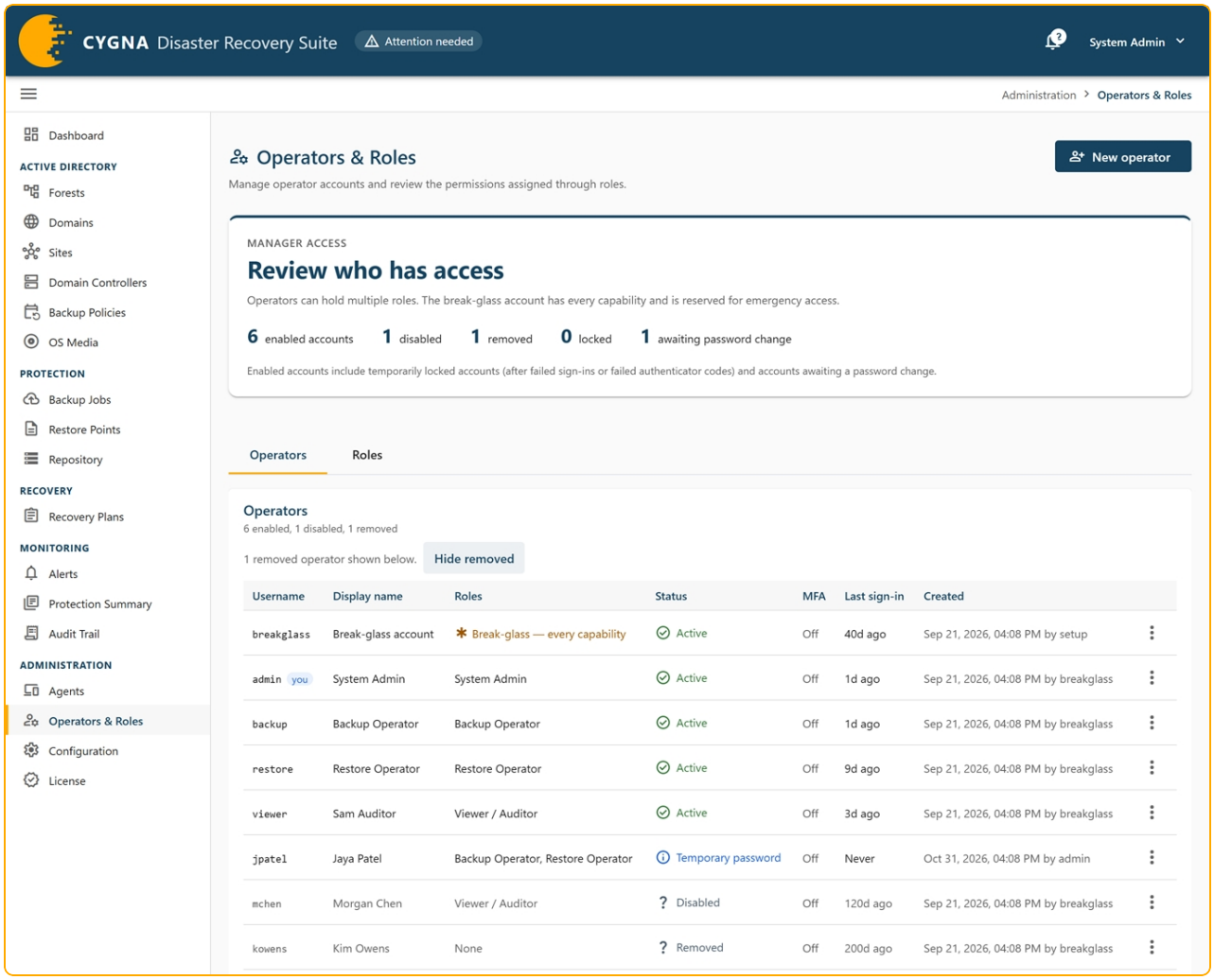
- Maintaining Agents
- Certificates and Trust
- Agent and Trust Commands

Operators and Roles Screen

The **Operators & Roles** screen lists who can sign in to this Manager and what each role may do. Operator accounts are Cygna Forest Recovery's own, held in the Manager's database rather than in Windows or Active Directory, so you can sign in when the directory is down, and the accounts come back with the Manager after a rebuild.

To open it, select **Administration** > **Operators & Roles**. The screen requires the `accounts.manage` capability, held by the System Admin role and the break-glass account.

For procedures and guidance on assigning roles, see Managing Operators and Roles.



Operators tab

The **Operators** card lists every account except removed ones, with a count of enabled, disabled, and removed accounts. When removed accounts exist, a line above the table says how many are not shown; select **Show removed** to list them at the end of the table, and **Hide removed** to hide them again. The **Manager access** panel above the tabs also

counts enabled, disabled, removed, and locked accounts, and accounts awaiting a password change.

Column	Description
Username	What the operator signs in with, and what the audit trail records. Your own account is tagged **you**.
Display name	The operator's name as shown in the console.
Roles	The roles the account holds. The break-glass account shows **Break-glass — every capability**; a removed account shows **None**.
Status	**Active**, **Disabled**, **Locked**, **Temporary password**, or **Removed** (see below).
MFA	Whether the operator uses multi-factor authentication: **Off**, **On**, **On, no recovery codes**, or **On, codes locked** after too many wrong authenticator codes. The tooltip gives the number of unused recovery codes, or when a code lock ends. Each operator turns MFA on for themselves.
Last sign-in	When the operator last signed in, or **Never**.
Created	When the account was created, and by whom.

Status	Meaning
Active	Can sign in.
Disabled	Cannot sign in. The account and its history are kept.
Locked	Locked after repeated failed sign-ins, until the time shown. Lockout lasts 15 minutes after 5 failures.
Temporary password	Has not yet signed in with the temporary password an administrator issued.
Removed	Removed for someone who has left. Holds no password, MFA, or role, and cannot sign in. The username stays reserved. The tooltip says when and by whom.

Row actions

Each row has a menu with the actions that apply to it:

- **Edit name and roles…**—Change the display name and roles. On your own row this reads **Edit display name…**, because you cannot change your own roles.
- **Reset password…**—Issue a temporary password (not available on your own row).
- **Reset MFA…**—Turn off multi-factor authentication for an operator who has lost their phone and their recovery codes. Shown only when the operator uses MFA, and not on your own row.
- **Unlock**—Shown when the account is locked, or its MFA codes are locked. Lets the operator try again immediately.
- **Disable…** or **Enable…**—Stop or restore the operator's ability to sign in (not available on your own row).
- **Remove…**—Remove the account of someone who has left (not available on your own row).

On a removed account's row, the menu says when and by whom the account was removed, and offers one action: **Restore…**.

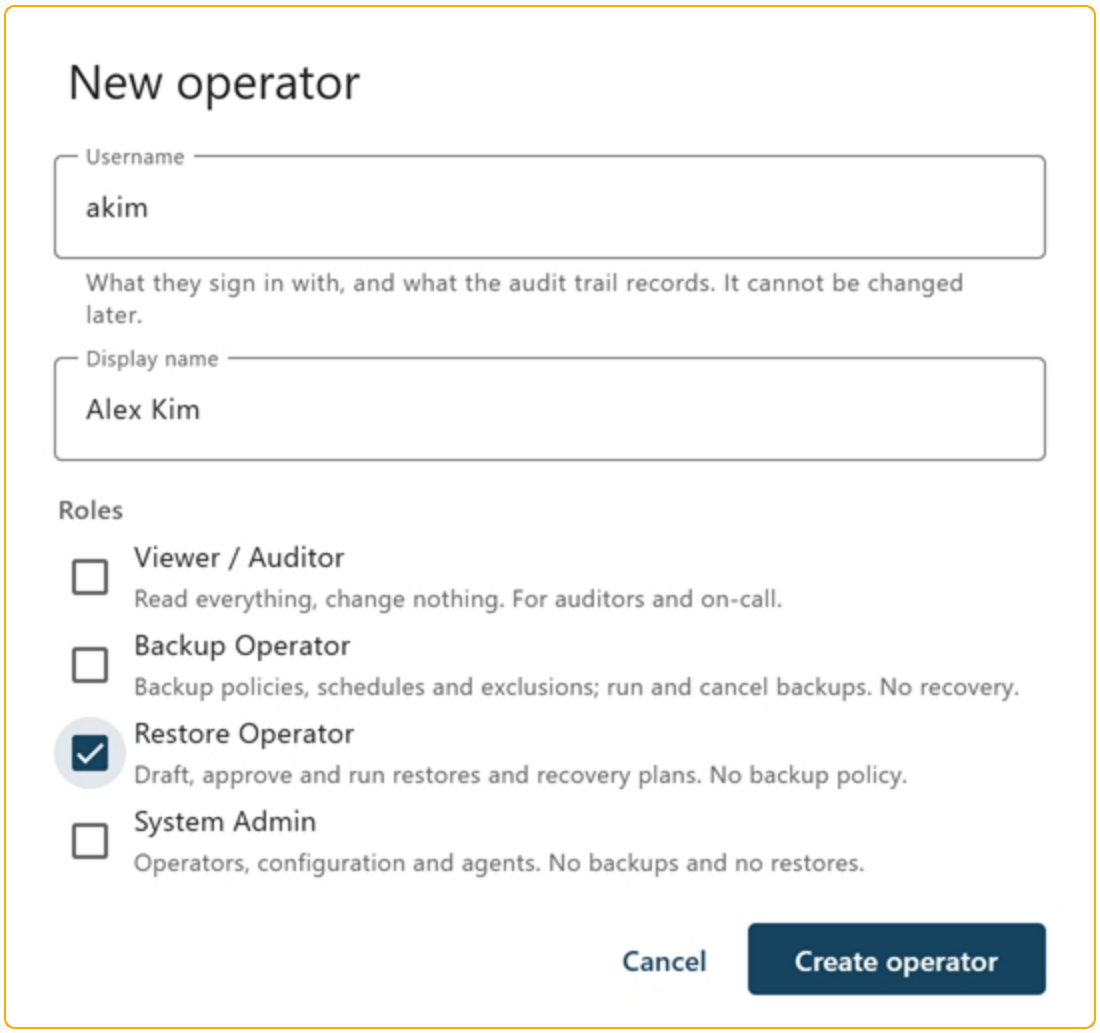
Accounts cannot be deleted. Disable is for a temporary absence and keeps the roles; remove is for someone who has left. See Managing Operators and Roles.

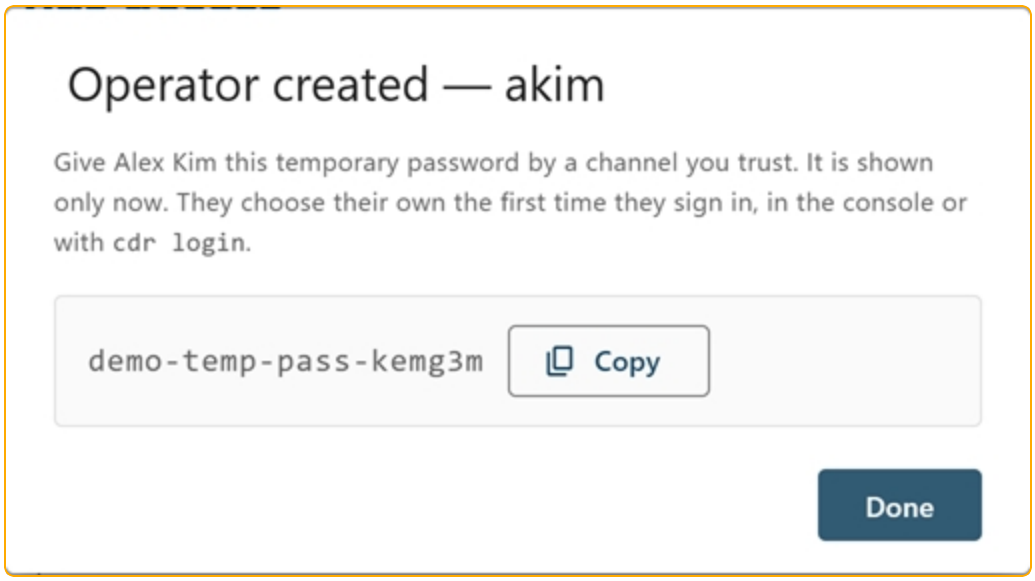
Creating an operator

Select **New operator** to open the operator dialog:

Field	Description
Username	2 to 64 characters: letters, digits, dot, hyphen, and underscore, starting with a letter or digit. It cannot be changed later.
Display name	Required, up to 128 characters.
Roles	One or more of the roles below. At least one is required.

Select **Create operator**. The Manager generates a temporary password and shows it once, in a dialog titled **Operator created — <username>**. Select **Copy**, give the password to the operator by a channel you trust, then select **Done**. The operator chooses their own password the first time they sign in, in the console or with `cdr login`. Administrators never set another operator's password.


New operator
Username
akim
What they sign in with, and what the audit trail records. It cannot be changed later.
Display name
Alex Kim
Roles
Viewer / Auditor
Read everything, change nothing. For auditors and on-call.
Backup Operator
Backup policies, schedules and exclusions; run and cancel backups. No recovery.
Restore Operator
Draft, approve and run restores and recovery plans. No backup policy.
System Admin
Operators, configuration and agents. No backups and no restores.
Cancel
Create operator


Operator created — akim
Give Alex Kim this temporary password by a channel you trust. It is shown only now. They choose their own the first time they sign in, in the console or with cdr login.
demo-temp-pass-kemg3m
Copy
Done

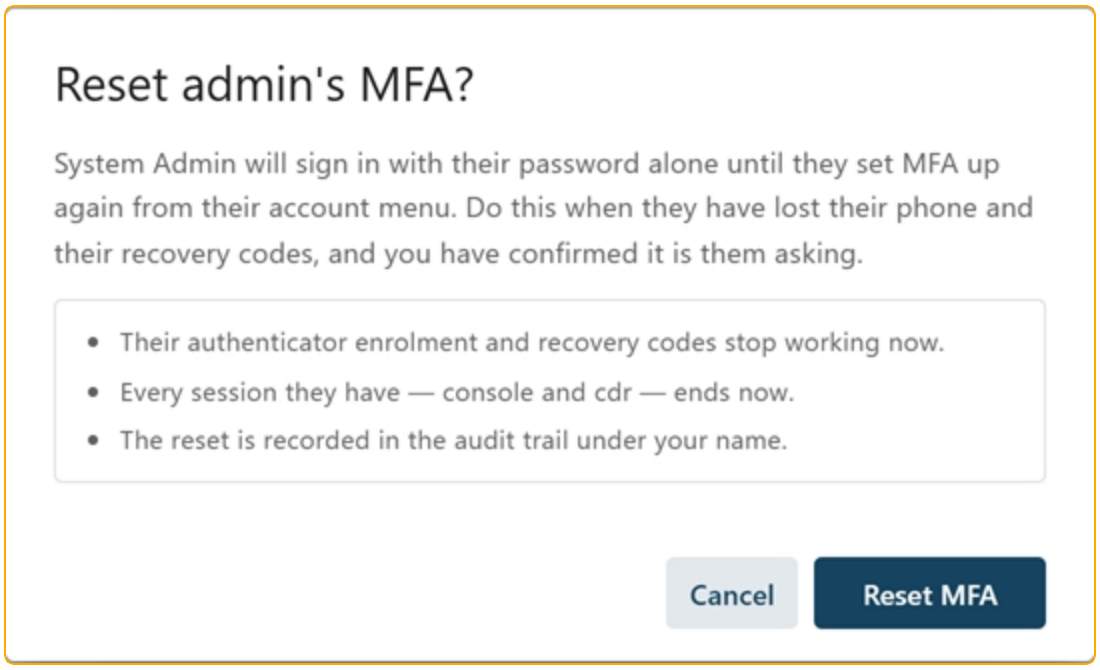
Resetting a password

Select **Reset password…** on an operator's row, then **Reset the password**. The operator's current password stops working, every session they have in the console and in `cdr` ends, and the reset is recorded in the audit trail under your name. The Manager shows a new temporary password once, in the same way as for a new operator.

To change your own password, use **Change password** in your account menu. Passwords must be at least 12 characters and must not contain the username.

Resetting MFA

Select **Reset MFA…** on an operator's row, then **Reset MFA**. Do this only when the operator has lost their phone and their recovery codes, and you have confirmed it is them asking. Their authenticator enrolment and recovery codes stop working, every session they have in the console and in `cdr` ends, and the reset is recorded in the audit trail under your name. They sign in with their password alone until they set MFA up again from their account menu. No administrator can turn MFA on for someone else. See Multi-Factor Authentication.

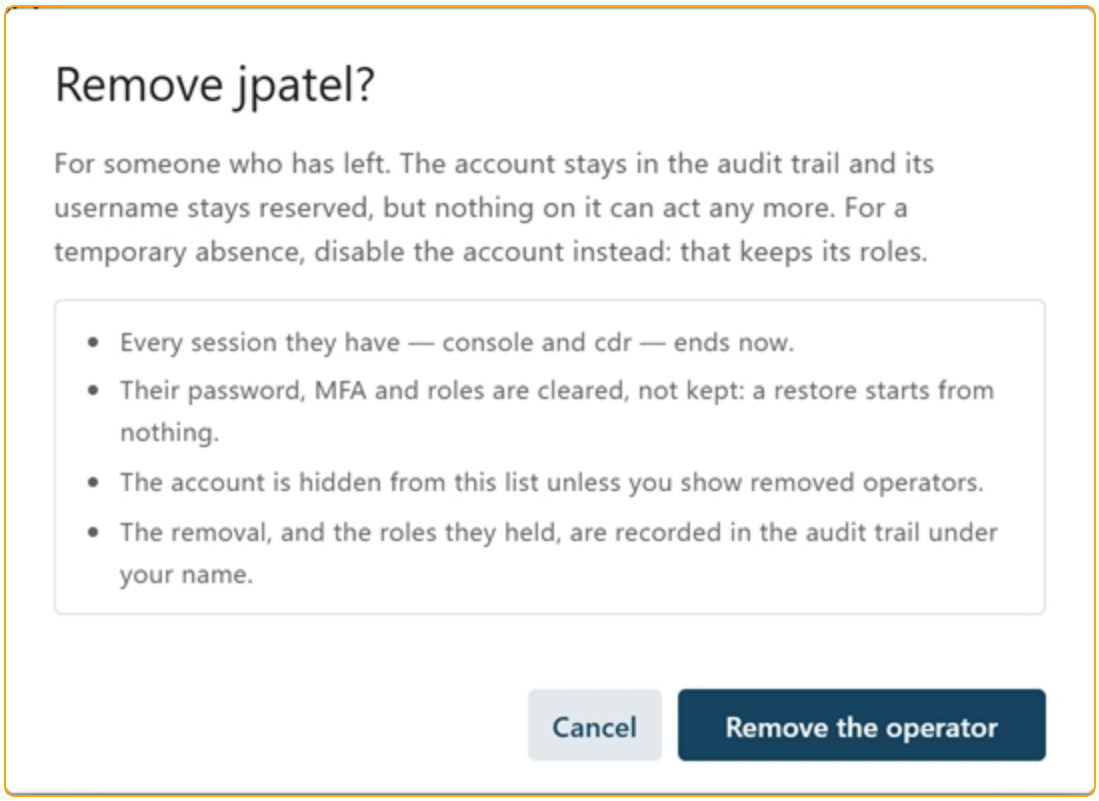


Disabling, enabling, and unlocking

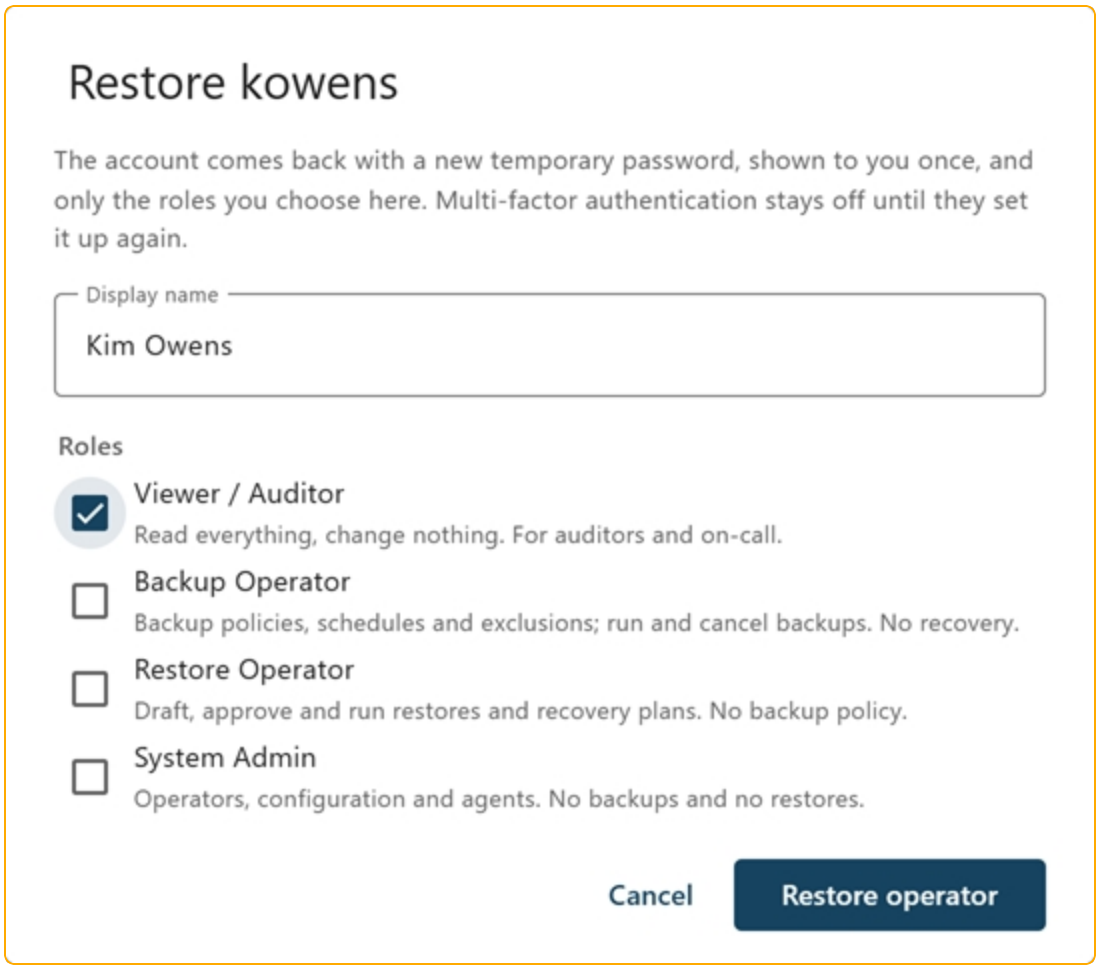
Select **Disable…**, then **Disable the account**. The operator cannot sign in until the account is enabled again, and every session they have ends on its next request. Select **Enable…**, then **Enable the account**, to let them sign in again with the roles the account already holds. Select **Unlock** to clear a lockout, or a lock on the operator's MFA codes. You cannot disable your own account.

Removing and restoring an operator

Select **Remove…** on an operator's row, then **Remove the operator**. Every session they have in the console and in `cdr` ends at once, their password, MFA, and roles are cleared, and the removal and the roles they held are recorded in the audit trail under your name. The account is then hidden unless you select **Show removed**. You cannot remove your own account or the break-glass account. Creating an operator with a removed operator's username is refused.



To bring a removed account back for the same person, select **Show removed**, then **Restore…** on its row. The dialog, titled **Restore <username>**, has the **Display name** and **Roles** fields of the operator dialog, with no role selected. Choose at least one role and select **Restore operator**. The Manager shows a new temporary password once, in a dialog titled **Operator restored — <username>**, in the same way as for a new operator. MFA stays off until the operator sets it up again.



The same actions are available as `cdr operator list`, `remove`, and `restore`. See Connecting and Signing In.

Roles tab

The **Roles** tab shows the **What each role may do** table, read from the Manager, which enforces it on every request. Each row is a capability, with its description and code, and each column a role. An operator can hold several roles.

Role	What it may do
Viewer / Auditor	Read everything, change nothing. For auditors and on-call staff.
Backup Operator	Backup policies, schedules, and exclusions; run and cancel backups. No recovery.
Restore Operator	Draft, approve, and run restores and recovery plans, including unattended plans, their notification addresses, and the emailed

Role	What it may do
	recovery report. No backup policy.
System Admin	Operators, configuration (including the mail server), and agents, including deleting an agent's record. No backups and no restores.
Break-glass	Every capability. Not an assignable role: a property of one account.

The roles separate duties on purpose. System Admin holds neither backup nor restore capability, and only Restore Operator (and break-glass) may run a restore, so account administration never brings destructive capability with it. For the full capability table, see Managing Operators and Roles.

What each role may do

Read from the manager, which enforces this table on every request

Capability	Viewer / Auditor	Backup Operator	Restore Operator	System Admin	Break-glass
See the dashboard and alerts `dashboard.view`	✓	✓	✓	✓	✓
See forests, domains, sites, DCs and agents `environment.view`	✓	✓	✓	✓	✓
See backups, policies, restore points and media `backup.view`	✓	✓	✓	✓	✓
See restore runs and recovery plans `plan.view`	✓	✓	✓	✓	✓
See the protection summary `reports.view`	✓	✓	✓	✓	✓
See the audit trail `audit.view`	✓	✓	✓	✓	✓
Change backup policies, schedules and exclusions `backup.manage`	—	✓	—	—	✓
Start and cancel backups `backup.run`	—	✓	—	—	✓
Manage OS and recovery media `media.manage`	—	✓	—	—	✓
Draft and approve restores and recovery plans `plan.approve`	—	—	✓	—	✓
Run a restore or recovery plan, and acknowledge its points of no return `plan.execute`	—	—	✓	—	✓
Install and update agents `agents.manage`	—	—	—	✓	✓
Create operators and assign their roles `accounts.manage`	—	—	—	✓	✓
Change manager and agent configuration `config.manage`	—	—	—	✓	✓
Set the recovery passphrase and export the configuration bundle `config.export`	—	—	—	✓	✓

The break-glass account

The break-glass account is created at the first console sign-in and holds every capability. It exists for when nothing else works. Its row cannot be changed from this screen: no administrator can change its roles, password, or status. To change its password, sign in as

the break-glass account and use **Change password**, then update the offline record of the credential. The one action on its row is **Unlock MFA codes**, shown when too many wrong authenticator codes have locked its codes.

Note: Nobody can change their own roles, and nobody but the break-glass account itself can change the break-glass account. Either would let one person go from account administration to running a restore without a second person.

For how the break-glass account is created, see Signing In for the First Time.

Related topics

- Audit Trail
- Multi-Factor Authentication
- Manager Day-One Checklist

Configuration

The **Configuration** screen holds the settings for the Manager and every agent. The settings are stored in the Manager's database and pushed to the agents from there, and each agent reports back which settings it is actually running. This topic describes how the screen works; for every setting, its default, and its range, see Settings Reference.

To open it, select **Administration** > **Configuration**. The screen requires the `config.manage` capability, held by the System Admin role and the break-glass account. The tabs run long: once you scroll down, a **Back to top** button in the lower right corner takes you back to the save bar and the tabs.

Layout

The **Manager and fleet settings** panel at the top reads **Review your pending changes** while you have unsaved edits, and **Review settings and where they apply** otherwise. It shows how many agents are waiting for a restart to apply saved settings and how many have not reported their configuration, then when the configuration last changed (**Last changed** and how long ago), or **Nothing has been changed from its default**. Hover over it to see the current configuration revision. With `agents.manage`, the panel also offers **Restart them…**, or **Restart *n* that can be restarted from here…** when only some of the waiting agents can restart themselves. Either button restarts every connected agent waiting for a restart. See Agents.

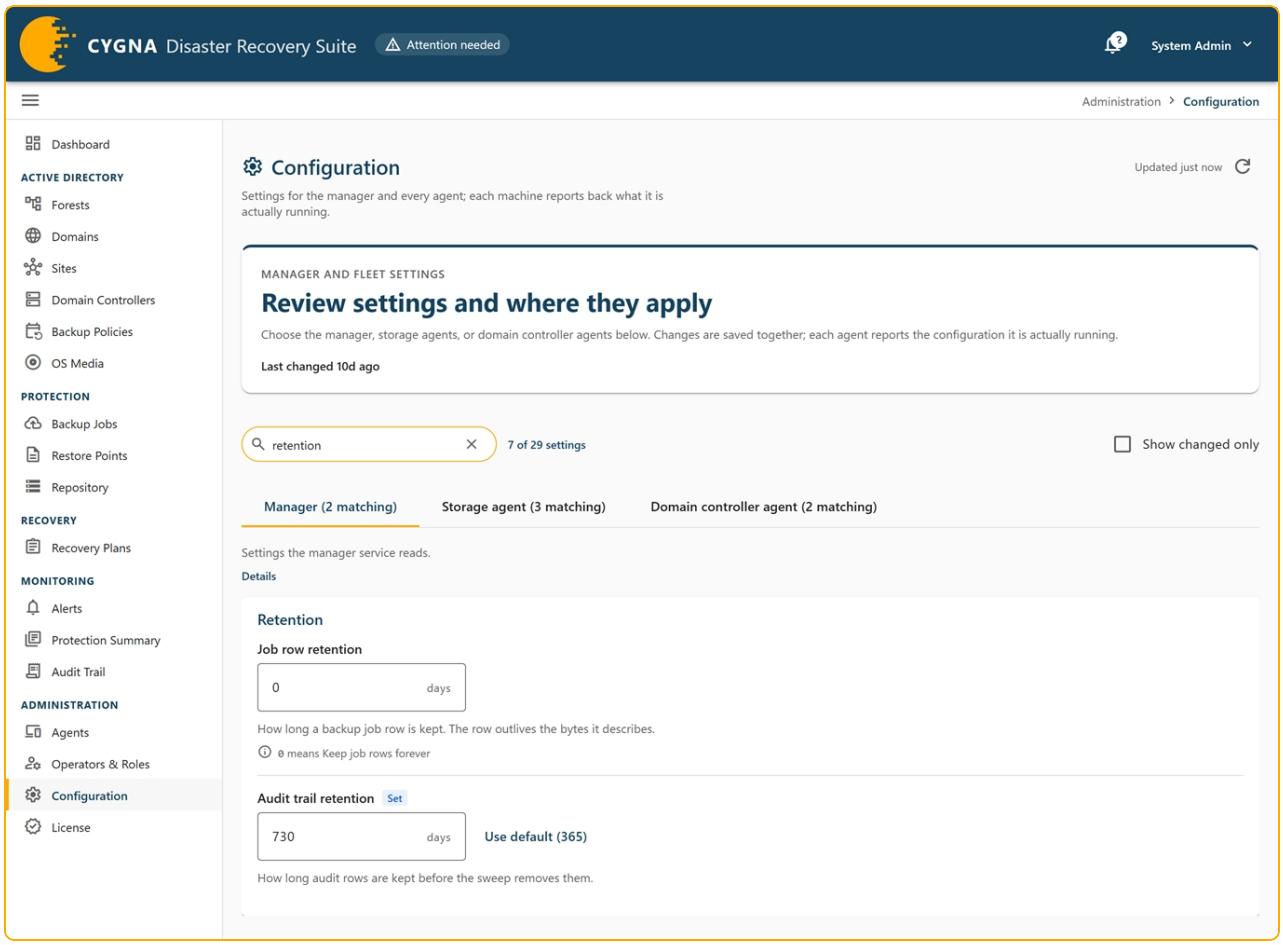
The settings are on four tabs, one per scope. Within each tab, settings are grouped in cards, such as **Backup execution** or **Vault retention**.

Tab	Applies to
Manager	Settings this Manager reads, such as scheduled test restores, backup timeouts, concurrency, database retention, agent deployment, and the mail server it sends email through. Changes take effect on the next sweep or dispatch, except where a field says a restart is needed.
Storage agents	Settings sent to the storage agents: staging, vault, retention, verification, the OS media library, and recovery images. Storage agents cache these and keep running on the last values they received if the Manager is unreachable.
Domain controller agents	Settings sent to every DC agent as a fleet default: heartbeat, topology reporting, job logs, and Windows Server Backup installation. Agents apply a change when they receive it, and catch up on their next connection if they were offline.
Isolated Recovery Host agents	Settings sent to the agents on Isolated Recovery Hosts, the Hyper-V hosts that test restores create their VMs on: readiness checks, Windows features, and the VMs and recovery network for test restores. An installation without an Isolated Recovery Host has nothing here to send to.

Finding a setting

Above the tabs, a search box and **Show changed only** narrow the settings on every tab at once:

- **Search**—The box (**Search name, key, description, group…**) keeps the settings whose label, key, summary, or group contains the text you enter. To find a field, you can paste a setting key as `cdr config show` prints it. The longer explanation under **Details** is not searched. The count beside the box shows how many settings match across all tabs, as ***n* of *m* settings**.
- **Show changed only**—Hides every setting that is at its product default.



While you search, each tab's label shows how many settings matched on it, for example **Storage agents (3 matching)**. A tab where nothing matched says so and names the tabs where the search did match, for example **No settings here match “retention”. It matches on: Manager 2, Storage agents 5.** The browser address keeps the search as `?q=`, so it survives a refresh and you can bookmark or share it. Neither filter hides an unsaved change: the save bar still counts and names every one.

A tab's label also shows how many unsaved changes it holds, for example **Manager (2 matching, 1 pending)**. The agent tabs also show whether their agents are running the current configuration, for example **All 12 agents are on the current configuration**, or **10 of 12 agents on the current configuration · 1 catching up · 1 not reported**. An agent catching up has reported an older configuration; an agent not reported has never reported one. Hover over the chip to see the current revision.

Settings

Each setting shows its label, its current value as an On/Off switch, a list, or a number or text box with its unit, and a summary of what it does. Where a value of 0 has a special meaning, such as "keep forever", a line under the field says so. A **Details** section explains the setting further. Below each field:

- **Undo**–Shown while the field has an unsaved change.
- **Use default (<default>)**–Removes your value so the setting follows the product default, now and in future releases.

A password setting, such as **SMTP password** in the **Email** group, is write-only. The Manager never sends the stored password back, so the field never holds it:

- The field starts empty. Its placeholder reads **Set — type to replace it** when a password is stored, and **Not set** when none is.
- Typing in the field queues the new password as a change. Emptying the field again takes the change back; it does not clear the stored password.
- To remove the stored password, select **Clear** below the field, where other settings show **Use default**. The field then says **Will be cleared when saved.**
- Saving without typing leaves the stored password as it is.
- A save that changes **SMTP server** or **SMTP security** must also enter the password again or clear it, or it is refused, so the stored password is never sent to a new server.

The audit trail records only that the password was set, changed, or cleared. For how the password is stored, see Settings Reference.

Badges mark settings that behave differently:

Badge	Meaning
Set on the machine	The value is set in the machine's own configuration file, usually at install time, and shown here as the agent reports it. It is read-only on this screen. These settings are in each tab's **Local to each machine** card, and include the storage root, Manager address, CA fingerprint, and log folders.
Restart required	Saved immediately, but not in force until the service restarts. The agent confirms which settings are waiting for a restart.
Can delete data	Changing this can cause backups or records to be deleted. Saving it needs a typed confirmation.
Cannot weaken the machine's floor	The storage machine uses the more protective of its own value and this one.
Set	Changed from the product default.

The retention floor and the minimum versions per DC are set on the storage machine and shown on the **Storage agents** tab read-only. A value sent from this screen can tighten retention but never weaken it. To lower a floor, change it on the storage machine itself.

Saving changes

Edits are not sent until you save. When you have unsaved changes, a save bar shows how many there are on each tab, with **Discard** and **Save**. If you try to leave the screen with unsaved changes, the console asks you to confirm; leaving discards them.

The Manager checks the whole batch before saving it. If any value is refused, nothing is saved and the screen lists each error. The Manager also checks settings against each other. For example, the vault reservation timeout must exceed the backup job timeout by at least 30 minutes; a conflict like this is shown as **Inconsistent** and blocks the save. Combinations that are allowed but have no effect, such as a version ceiling below the storage machine's minimum, are shown as **Worth knowing**.

When the changes are saved, the screen reports the new revision and how many connected agents were told. Agents that were offline pick up the change on their next connection.

Changes that can delete data

When any pending change has the **Can delete data** badge, the save button reads **Review and save…**. It opens a confirmation titled **This change can delete data** that lists each such change, its scope, and its old and new value. Type `CONFIRM` and select **Save and reduce what is kept** to save.

Machines in this scope

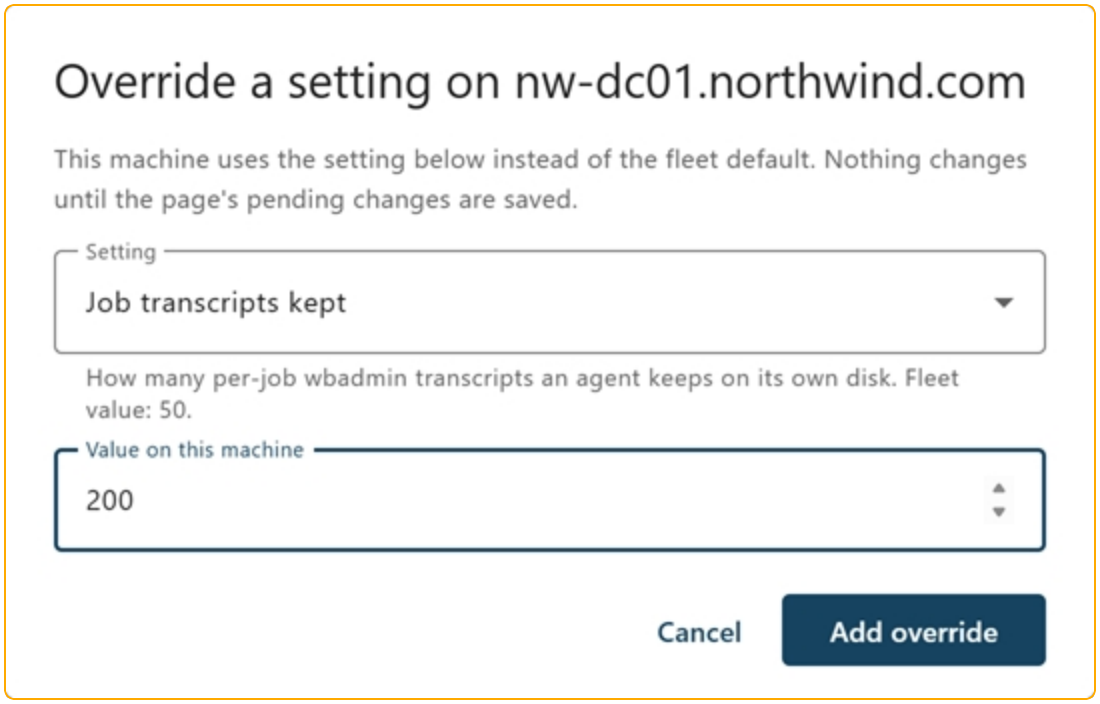
On the agent tabs, the **Machines in this scope** card shows what each agent reports it is running:

- Whether it is running the current configuration (**Current configuration**, **Not yet on the latest change**, or **Has not reported**), and whether it is connected. Hover over the chip to see the revision it is running and the current revision.
- Settings saved but not in force until the agent restarts.
- Values the machine declined because they were less protective than its own floor, with the value it is using instead.
- Settings its local configuration file still sets that are now managed from this screen and are being ignored. Remove them from the file.
- Settings overridden for this machine (see below).

Per-machine overrides

An agent can use a different value from the fleet default for any setting that is not set on the machine, for example to try a change on one domain controller before applying it to all. Select **Override a setting on this machine…**, choose the **Setting**, enter the **Value on this**

machine, and select **Add override**. The override is queued with your other changes and saved from the save bar. To remove an override, select **Follow the fleet default** beside it.



Manager recovery, configuration backups, and support packages

The **Manager recovery**, **Configuration backups**, and **Support package** cards are on the **License** screen. See Manager recovery and configuration backups and Support package in Licensing.

Related topics

- Settings Reference
- Retention
- Configuration and Exclusion Commands
- Logs and Diagnostics

Backup Tasks

This chapter covers the day-to-day work of keeping every domain controller recoverable: scheduling backups, taking a backup outside the schedule, following and stopping backup jobs, and deciding which objects Cygna Forest Recovery counts when it reports on coverage.

The Manager starts every backup. Agents never start one on their own, and no domain controller is backed up until it has a backup policy. A policy belongs to one machine and sets how often it is backed up, optionally from what time of day, and which backup type is taken. Each backup runs as a job that you can follow on the **Backup Jobs** screen until it is committed to the vault and becomes a restore point.

Who can do what

The tasks in this chapter need the Backup Operator role, or the break-glass account:

- **Changing policies and exclusions**—creating, editing, disabling, and removing backup policies, excluding objects, and recording objects as removed from the directory.
- **Starting and stopping backups**—**Back up now** and **Stop backup** on the **Domain Controllers** screen, **Back up now** on the **Backup Policies** screen, **Retry** on a failed job on the **Backup Jobs** screen, and the equivalent `cdr` commands.

Any operator who can view backups sees the **Backup Policies** and **Backup Jobs** screens. Changing a schedule on **Backup Policies** needs the Backup Operator role. For more information, see Managing Operators and Roles.

Before you start

Backups need a DC agent on each domain controller and a connected storage agent. If you are setting up a new installation, start with Creating Your First Backup Policies. For how a backup moves from the domain controller into the vault, see How Backup Works.

In this chapter

- Managing Backup Policies
- Running a Backup Now
- Monitoring and Cancelling Backup Jobs
- Running a Test Restore
- Excluding and Retiring Objects

Managing Backup Policies

A backup policy tells the Manager how often to back up one domain controller, optionally from what time of day, and which backup type to take. Create a policy for every domain controller you want to protect, change the interval or type as your recovery needs change, and disable a policy when you need to pause backups without losing its settings. You manage policies on the **Active Directory** > **Backup Policies** screen, one row per domain controller.

Before you begin

- You need the Backup Operator role. Other operators can open the screen, but the Manager refuses their changes.
- The domain controller must be known to the Manager. A machine appears on the screen once its DC agent enrolls or discovery reports it. You can create a policy for a machine that has no agent yet, but it does not run until an agent is enrolled. The row says "No agent is enrolled, so this will not run." The schedule is held, with **Next due** showing **Waiting for agent**, and no failed backup is recorded. The first backup runs as soon as the machine's first DC agent connects. Once a machine has had a DC agent, a due backup that cannot run because the agent is stopped or the machine is down is recorded as a failed job, so a protected machine that is not being backed up shows up.
- Decide on the backup type. Full system recovery needs full system backups. See Backup Types.

Choosing the interval, start time, and type

The interval is how long the Manager waits after one scheduled backup before it starts the next. Without a start time (**Any time**), the Manager spreads policies slightly around their interval so that a fleet does not start every backup at the same moment (the **Schedule jitter** setting).

With a start time (**At a set time**), backups run at fixed times of day, for example outside business hours. Keep these rules in mind:

- The start time is 24-hour, in the Manager's local time, not the time zone of your browser or the domain controller.
- It is the earliest a backup is due, never the exact time. Each backup is due at a random moment within the **Start time spread** after it (at most half the interval), so domain controllers on the same start time do not all ask for a backup slot at once.
- Backups then wait for a free backup slot. With more domain controllers on one start time than **Maximum concurrent backups**, the last of them start well after it. The editor estimates how late, and warns you when the slots cannot keep up. To finish sooner,

raise **Maximum concurrent backups** or give some domain controllers another start time.

- Only an interval that runs at the same times every day can have a start time: a whole number of days, or one that divides a day evenly (every 1, 2, 3, 4, 6, 8, or 12 hours, or a divisor of an hour). For example, every 6 hours from 22:00 runs at 22:00, 04:00, 10:00, and 16:00. Every 7 hours is refused.
- Runs stay on their times of day. A backup that started late, for example after a long wait for a slot, is followed by the next start time that is at least half an interval later.
- Daylight saving time: a start time in the hour that is skipped in spring runs at the first time that exists after it (02:30 becomes 03:00). A start time in the hour that repeats in autumn runs once, in the first of the two.

For the spread settings and **Maximum concurrent backups**, see Settings Reference.

Option	Description
Hourly, **Every 4 hours**, **Every 12 hours**, **Daily**	Interval presets. A new policy starts on **Daily**.
or every *n* days / **hours** / **minutes**	A custom interval. It must be a whole number of minutes, at least 5 minutes and at most 90 days.
Full system	Captures the critical volumes, including Windows, applications, and DC state. The backup supports both DC state recovery and full system recovery. This is the default for a new policy.
DC state	Smaller and faster to take, but supports DC state recovery only. A machine whose Windows installation is lost cannot be recovered from it and has to be rebuilt instead.

The **Recovers** column shows the result of your choice for each machine: **DC state + Full system** or **DC state only**. When any enabled policy takes DC state only, the screen shows an advisory with a **Show these** button that lists those machines.

For guidance on how often to back up and how much history the vault keeps, see Retention and Coverage and Readiness.

Create a policy

To create a backup policy for one domain controller:

1. Go to **Active Directory** > **Backup Policies**.

   Machines that need a policy are listed first. To show only those with backups and no policy, select the **unscheduled** chip. To show the domain controllers that could make up a domain's shortfall, select the **need a backup policy** chip.

2. On the row for the domain controller, select **Schedule**.

   The **Schedule** dialog opens with the defaults: enabled, daily, any time, Full system.

3. Under **State**, leave **Enabled** selected.

4. Under **Interval**, select a preset, or enter a number and select a unit beside **or every**.

5. Under **Start time**, leave **Any time** selected, or select **At a set time** and enter the time.

   With a start time, the dialog names the Manager's time zone and estimates when the domain controllers on that time should start.

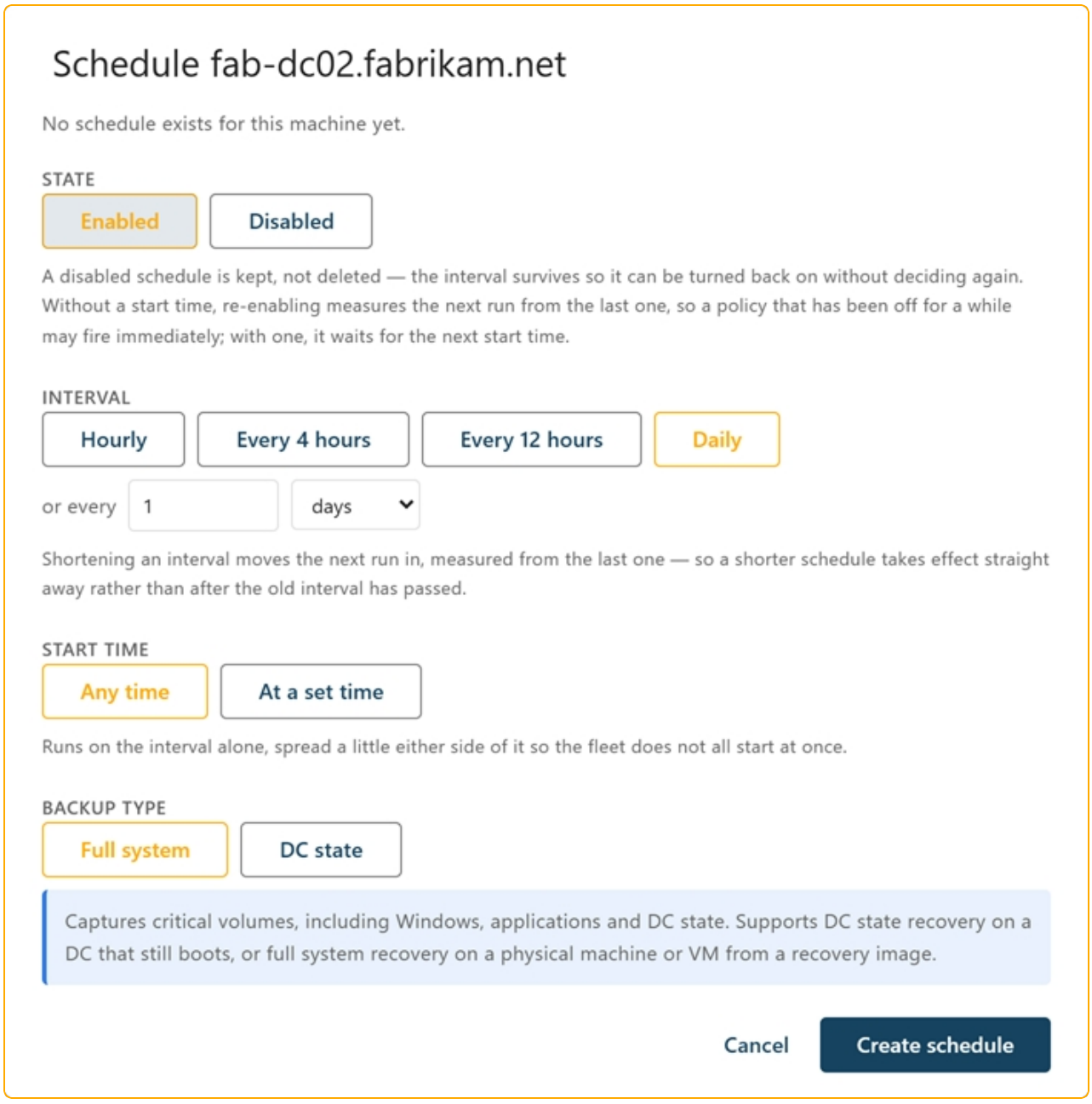
6. Under **Backup type**, select **Full system** or **DC state**.

   The dialog states what the chosen type can recover.

7. Select **Create schedule**.

   A message confirms the schedule and when the first backup is due: immediately, or at the first start time. For a machine whose DC agent has not come online yet, the first backup waits until the agent connects.

To try a new schedule straight away, select **Back up now** (play icon) on its row. See Running a Backup Now.



Edit a policy

To change the state, interval, start time, or type of one policy:

1. On the **Backup Policies** screen, select **Edit** on the row for the domain controller.
2. Change the **State**, **Interval**, **Start time**, or **Backup type**.
3. Select **Save changes**.

   Changes take effect as soon as they are saved. The confirmation says how the next run moved, for example from one time to another, or that the timing is unchanged. If

the agent was not reachable, the message says it picks up the change when it reconnects.

How an edit moves the next run:

- **Without a start time**–the next run is measured from the last one. Shortening an interval therefore brings the next backup forward at once, and a policy that has been disabled for a while may run as soon as you enable it again. Removing a start time also measures the next run from the last one.
- **With a start time**–setting or changing the start time, changing the interval, or enabling the policy moves the next run to the first start time that is at least half an interval after the last backup and not in the past. For example, a daily policy that last ran yesterday at 23:10 and is set to 22:00 runs tonight at 22:00. Changing only the backup type keeps the next run.

Edit several policies at once

Use bulk edit to apply the same change to many domain controllers, for example to move a domain to full system backups or to give every unscheduled machine a policy.

To edit several policies:

1. On the **Backup Policies** screen, select the checkbox on each row you want to change.

   To select every row that matches the current filter and search, select the checkbox in the header, then select **Select all *n* matching**.

2. In the bar at the bottom of the screen, select **Edit *n* schedules...**.

   The dialog lists the selected machines.

3. For each of **State**, **Interval**, **Start time**, and **Backup type**, either leave **Keep current** selected or select the new value.

   If a start time does not suit the interval some machines keep, the dialog names them; choose an interval for all of them, or **Any time**. With a start time, the dialog warns you when more machines share it than there are backup slots.

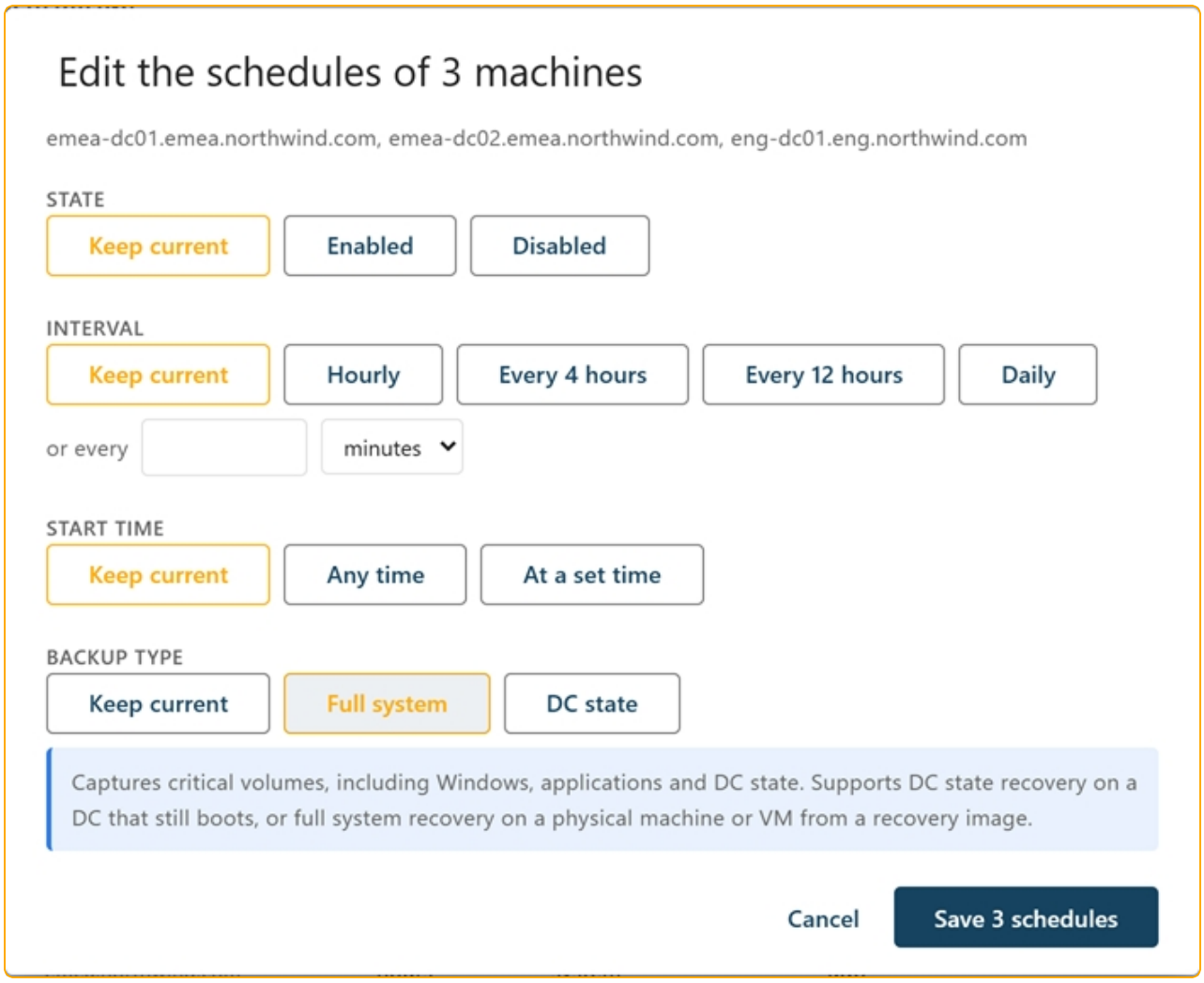
4. Select **Save *n* schedules**.

   If some machines could not be saved, the dialog lists them with the Manager's reason. Correct the problem and select **Retry *n***.

How **Keep current** works:

- A field left on **Keep current** keeps each machine's own value. Only the fields you change are applied to every selected machine.

- A selected machine that has no policy yet gets one. Any field left on **Keep current** uses the defaults for a new policy: enabled, daily, any time, Full system. The dialog says how many of the selected machines have no policy.
- If none of the selected machines has a policy, the **Keep current** choices are not shown. The fields start on the defaults, and every machine gets exactly what is selected.
- You must change at least one field, unless you are creating policies for machines that have none.



Disable or remove a policy

Disabling and removing both stop scheduled backups, but they are not the same:

- **Disable**–the policy is kept with its interval, start time, and type, so you can turn it back on without deciding again. The **Schedule** column shows the interval followed by **· off**, and **Next due** shows **Not scheduled**.

- **Remove**–the policy is deleted, along with its settings. The machine counts as unscheduled. Removing a policy deletes no backups.

To disable a policy:

1. Select **Edit** on the row for the domain controller.
2. Under **State**, select **Disabled**, and then select **Save changes**.

To disable several policies, use bulk edit and set only **State** to **Disabled**.

To remove a policy:

1. On the row for the domain controller, select the **Remove schedule** icon. To remove several, select their rows and then select **Remove *n* schedules...** in the bar at the bottom of the screen.
2. Read the consequences, and then select **Remove schedule**.



Note: A domain controller that is recorded as removed from the directory stays on this screen while it still has a policy, with the warning "Backups are still being dispatched to it; remove the schedule." Remove its policy so the Manager stops backing up a machine that is no longer part of the estate. See Excluding and Retiring Objects.

Using the command line

The `cdr policy` commands manage the same policies. `cdr policy set` reads the current policy first, so an edit changes only the options you give.

```
cdr policy list --due
cdr policy set dc01.corp.example.com --every 1440 --type bmr
cdr policy set dc01.corp.example.com --every 1440 --at 22:00
cdr policy set dc01.corp.example.com --at none
cdr policy set dc01.corp.example.com --disabled
cdr policy set dc01.corp.example.com --enabled
cdr policy remove dc01.corp.example.com
```

`--every` takes minutes and is required when you create a policy. `--at` sets the start time in the Manager's local time, and `--at none` removes it; an edit without `--at` keeps the start time. Setting `--type systemstate` asks you to confirm that the machine loses full system recovery; `--yes` skips the confirmation. For every option, see Backup Commands.

Next steps

- Check that each new policy produces a restore point. See Monitoring and Cancelling Backup Jobs.

- To take a backup without waiting for the schedule, see Running a Backup Now.
- For a description of every column and filter on the screen, see Backup Policies.

Running a Backup Now

Take a backup outside the schedule when you need a fresh restore point straight away: before a risky change such as a schema update or an operating system patch, after a domain controller was patched (a DC state recovery cannot restore DC state from before an update), after a recovery, or to prove that a new domain controller can be backed up. When a recovery plan completes, it requests a backup of every domain controller it restored or rebuilt that has an enabled backup policy, so you do not need to start those yourself. An ad hoc backup is an ordinary backup job. It does not change the domain controller's policy or its next scheduled run. You can start one from the **Domain Controllers** screen, of either type, or from the **Backup Policies** screen, of the type the machine's schedule takes.

Before you begin

- You need the Backup Operator role.
- The domain controller must have an enrolled DC agent, and a storage agent must be connected.
- No backup can already be running on the machine. A running backup shows **Stop backup…** in place of **Back up now**.
- No recovery can be working on the machine: a recovery plan running over it (or halted with a step that moved in the last 24 hours), a restore run in progress on it, or a plan that recovered it in the last 2 hours while its DC agent has not connected yet. The backup is then refused with **A recovery is working on this DC**, and no job is recorded. A scheduled backup of the machine waits and starts by itself when the recovery is done. A test restore does not hold backups.

Back up one domain controller

To back up a domain controller now:

1. Go to **Active Directory** > **Domain Controllers**.
2. Open the actions menu on the row for the domain controller.
3. Select **Back up now – Full system** or **Back up now – DC state**.

   Both types are offered whatever the machine's policy says. Choose **Full system** unless you need only a restore point for DC state recovery. A message confirms that the backup has been queued, with a **View backup jobs** link.

If the menu says "No agent enrolled, so nothing can back this machine up.", install the DC agent first. See Deploying DC Agents.

Back up several domain controllers

To back up several domain controllers now:

1. On the **Domain Controllers** screen, select the checkbox on each row.
2. In the bar at the bottom of the screen, select **Back up *n* now**, and then select the backup type.

   The Manager queues a job for every selected machine that can take one. If any could not be started, the screen lists them with the reason.



Back up from the Backup Policies screen

To try a schedule you have just created, or to back up a machine of the type its schedule takes, start the backup from its policy.

To back up a scheduled domain controller now:

1. Go to **Active Directory** > **Backup Policies**.
2. On the row for the domain controller, select **Back up now** (play icon).

   The button is shown only on rows that have a schedule, and its tooltip names the type, for example **Back up now (Full system)**. It is unavailable while a schedule is being saved. A message confirms that the backup has been queued, with a **View its jobs** link to the machine's jobs on the **Backup Jobs** screen.

If the Manager refuses the backup, the message **The backup of** *machine* **was not started.** gives the Manager's reason. It stays until you dismiss it.

Retry a failed backup

A job that ended without a restore point–**Failed**, **Interrupted**, or **Cancelled**–can be started again from the **Backup Jobs** screen. Open the job's row menu (⋮) and select **Retry – *type***, or select the same button on the job's detail sheet. The retry is a new backup of the same

machine and of the same type, sent as **Back up now** sends it, so the same conditions apply. A message confirms that it has been queued, or says why the Manager did not start it.

What happens next

A queued backup waits until the Manager has room to run it. The Manager limits how many backups run at once (see **Maximum concurrent backups** in Settings Reference), so a large bulk request runs in turns. A backup appears on the **Backup Jobs** screen as soon as it is queued, and then moves through these statuses:

1. **Queued**–the backup is waiting for a backup slot. It is kept across a Manager restart.
2. **Created**–the backup has a slot and the Manager has started the job.
3. **Staging**–the storage agent prepares the staging folder and a credential for this job only.
4. **Running**–the backup runs on the domain controller.
5. **Committing**–the storage agent verifies the content and moves it into the vault.
6. **Completed**–the backup is in the vault and is a restore point.

A machine has only one backup queued or running at a time. Asking for another while one is queued or running is refused, and the message names the backup that is already under way.

A backup that is running blocks a restore of the same machine until it finishes or is stopped. A queued backup does not: if a recovery starts working on the machine while the backup waits, the backup stays queued until the recovery is done. To stop a running or queued backup, see Monitoring and Cancelling Backup Jobs.

Using the command line

`cdr backup start` queues a backup of one domain controller. The default type is `bmr` (Full system). With `--wait`, the command follows the job and exits with its outcome, which is useful in a script.

```
cdr backup start dc01.corp.example.com
cdr backup start dc01.corp.example.com --type systemstate
cdr backup start dc01.corp.example.com --wait
```

If a recovery is working on the machine, `cdr backup start` refuses the backup and exits with code 3.

For every option and exit code, see Backup Commands.

Next steps

- Confirm that the job completed and produced a restore point. See Monitoring and Cancelling Backup Jobs.
- Review what the backup captured on the Restore Points Screen.

Monitoring and Cancelling Backup Jobs

Every backup, scheduled or started by an operator, runs as a job. Check jobs to confirm that new policies produce restore points, to find out why a backup failed, and to follow a backup you started by hand. Stop a running backup when it is in the way, for example when it would block a restore of the same machine or is loading a domain controller at a bad time.

Before you begin

- Any operator who can view backups can monitor jobs.
- Stopping a backup needs the Backup Operator role. The **Backup Jobs** screen has no stop action; you stop a backup from the **Domain Controllers** screen or with `cdr jobs cancel`.

Follow backup jobs

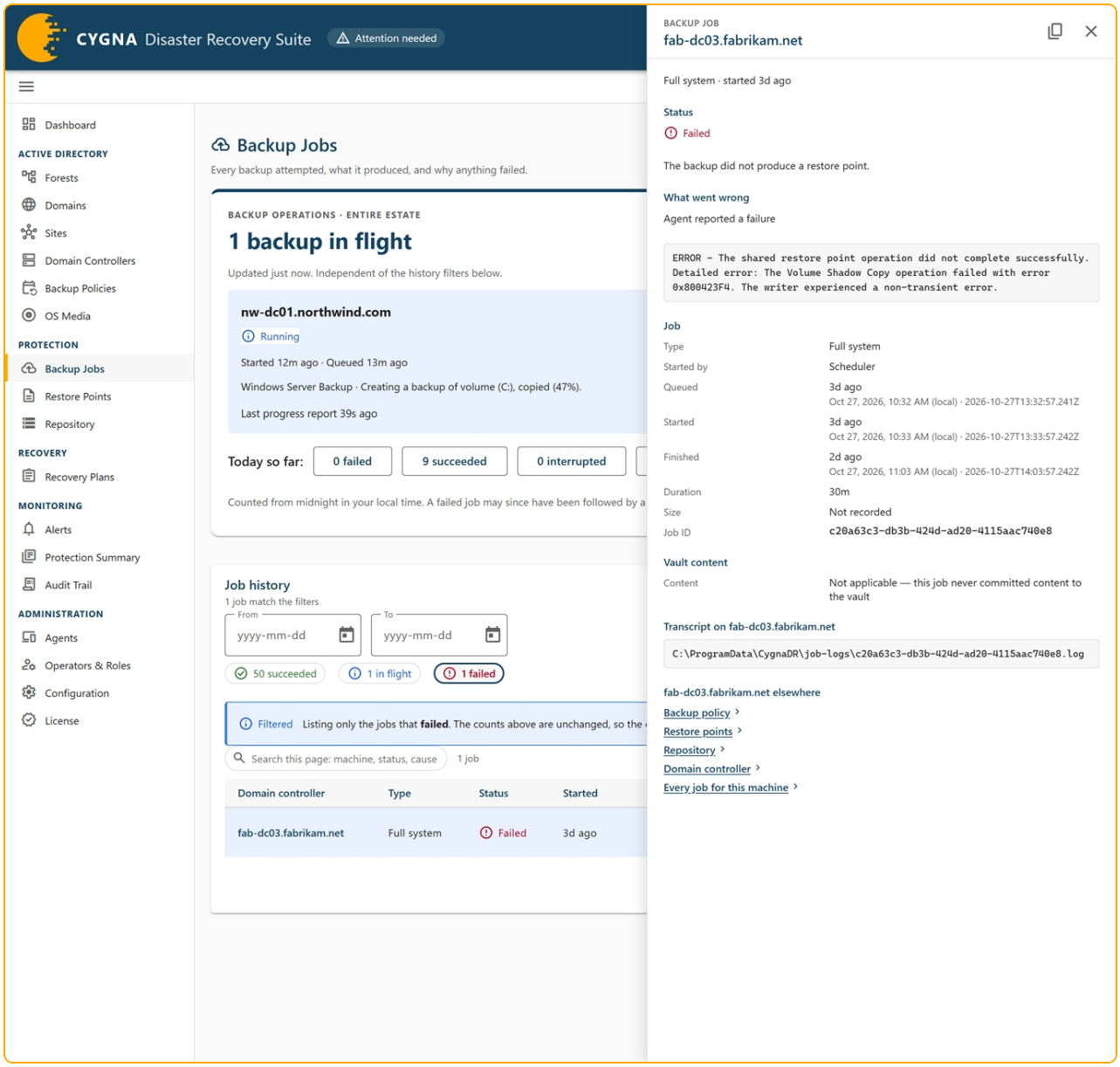
To check recent backup jobs:

1. Go to **Protection** > **Backup Jobs**.

   The **Job history** card lists jobs, newest first, with counts across the top. The screen refreshes itself every few seconds.

2. Narrow the list as needed:
   - Select a count chip, such as **in flight** or **failed**, to list only those jobs. Select it again to list all jobs.
   - Enter dates in **From** and **To** to limit the list to jobs created in that range.
   - Enter text in the search box to search the jobs on the current page by machine, status, cause, or error text.
3. While a job is in flight, read the line under its status. It shows the last phase the agent reported and how long ago.
4. Select the domain controller name on a row to open the job's details.

   The details show what the status means, what went wrong, the last progress report, the job's timings and size, whether its content is in the vault or has been pruned by retention, and the path of the transcript on the domain controller.



What to watch for

Status	What to do
Completed	Nothing. The backup is in the vault and is a restore point unless retention has since removed it.
Failed or **Interrupted**	No restore point was produced. Read the cause in the details and look it up in Backup Job Errors. After you fix the cause, select **Retry – *type*** in the row's actions menu (Backup Operator role), start a new backup, or wait for the next scheduled one. A **Cancelled** job can be retried the same way.

Status	What to do
Completed, not committed	The domain controller took the backup, but the Manager did not see it reach the vault. Usually nothing. The Manager checks the storage machine every hour and whenever the storage agent connects. It commits content still in staging and records content already in the vault, so the job changes to **Completed**. If the content is in neither place, the job stays as a record and is not a restore point. Take a new backup. To check at once, select **Check now** under **Backups not in the vault** on the **Repository** screen.
Completed (unreconciled)	The Manager could not confirm the backup against the vault. Treat it as uncertain, not as a restore point, until this is resolved.
A job that stays in flight	A **Queued** job is waiting for a backup slot; its **Started** column reads **Waiting for a slot**. If many jobs stay queued, backups are taking longer than their schedules allow; see **Maximum concurrent backups** in Settings Reference. For a job that has started, check the time of the last report under its status. If nothing has been reported for a long time, check the agent's connection on the **Agents** screen. The Manager fails a job that does not report within the **Backup job timeout**.

For all statuses and columns, see Backup Jobs.

Stop a running backup

Stopping a backup is a clean stop: the agent ends the job and releases its snapshot. Nothing already in the vault is touched, and the machine keeps its last good restore point. The stopped job produces no restore point, and the next scheduled backup runs as normal.

To stop a running backup:

1. Go to **Active Directory** > **Domain Controllers**.
2. Open the actions menu on the row for the domain controller and select **Stop backup…**.

   To stop several, select their rows and then select **Stop *n* running backups…** in the bar at the bottom of the screen.
3. Select **Stop the backup** (or **Stop *n* backups**).

   The message says the machine has been asked to stop. The job stays in flight until the backup winds down, and then its status changes to **Cancelled**. A backup that was still **Queued** never started, so it changes to **Cancelled** at once.

Using the command line

```
cdr jobs list --running
cdr jobs list --failed --detail
cdr jobs list --machine dc01.corp.example.com
cdr jobs cancel 3f2a9c1d
```

`cdr jobs cancel` takes the job ID, or the short form that `cdr jobs list` prints, and asks you to confirm; `--yes` skips the confirmation. For every option, see Backup Commands.

Next steps

- For job errors and their remedies, see Backup Job Errors.
- To see the overall effect on protection, check the Dashboard and Alerts.

Running a Test Restore

A test restore restores one backup version onto a new virtual machine on an Isolated Recovery Host, checks it as a working domain controller in isolation, and deletes the virtual machine. Nothing in production changes. Scheduled test restores run on their own once a host is ready. Start one yourself when you want an answer about a particular backup now, for example before you rely on it in a recovery. For how a test restore works, see Test Restores.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute.
- An Isolated Recovery Host must be ready. Check its panel on the **Administration** > **Agents** screen. See Setting Up an Isolated Recovery Host.
- The backup must be a **Full system** restore point.
- A recovery image must exist for the domain controller's operating system. Check the **OS Media** screen.

Start a test restore from the console

To run a test restore:

1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Single-DC recovery**.
2. In **Recovery mode**, select **Full system**.

3. In **Restore destination**, select **Test restore on *host* – isolated, nothing in production changes**.

   Until a host is connected and ready, the option is unavailable and reads **no Isolated Recovery Host is ready**.

4. In **Source domain controller**, select the domain controller, and in **Restore point**, select the full system restore point to test.

   Each restore point shows the result of its latest test restore, if any: **test restore verified**, **test restore failed**, **test restore running**, **test restore waiting to start**, or **test restore inconclusive**. The **Test restore** column of the **Restore Points** screen shows the same results, so you can find untested or failed versions before you start.

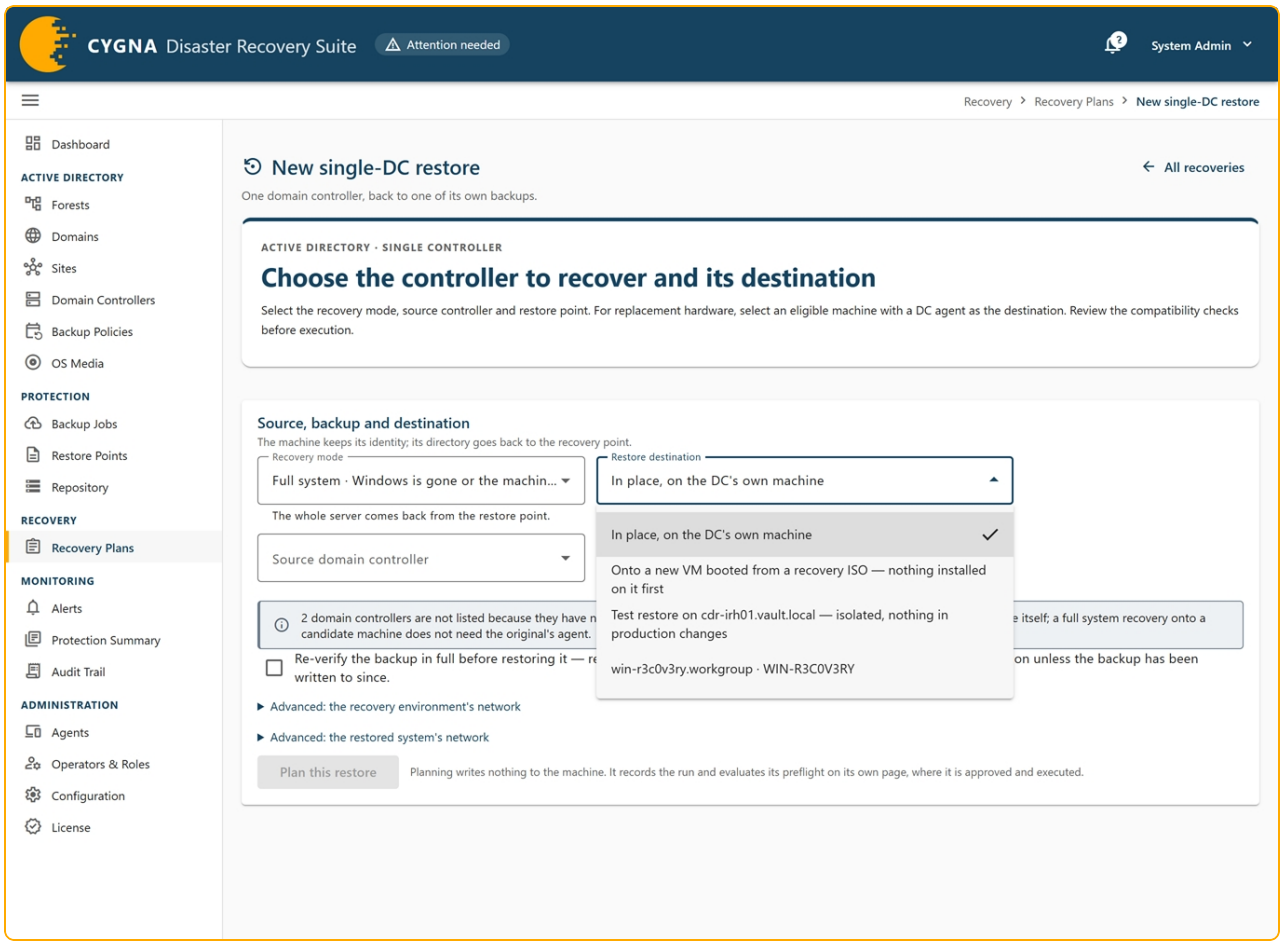
5. Select **Plan this test restore**.

   The run's page opens, titled **Test restore of *domain controller***.

6. Resolve any hard check on the **Preflight** card, and select **Approve this restore**.

7. Select **Execute…**, and in the **Test restore *machine*** dialog, select **Test *NAME* now**.

   The dialog says that the host restores the backup into a new virtual machine on its private switch, checks it, and deletes it, and that the domain controller itself is not touched. You do not type the domain controller's name, as you do for a restore that overwrites one.



Follow the test restore

While the test runs, the recovery banner on other screens reads **Test restore in progress — nothing in production changes**, unless a real recovery is also running. Select the domain controller's name in the banner to open the run's page.

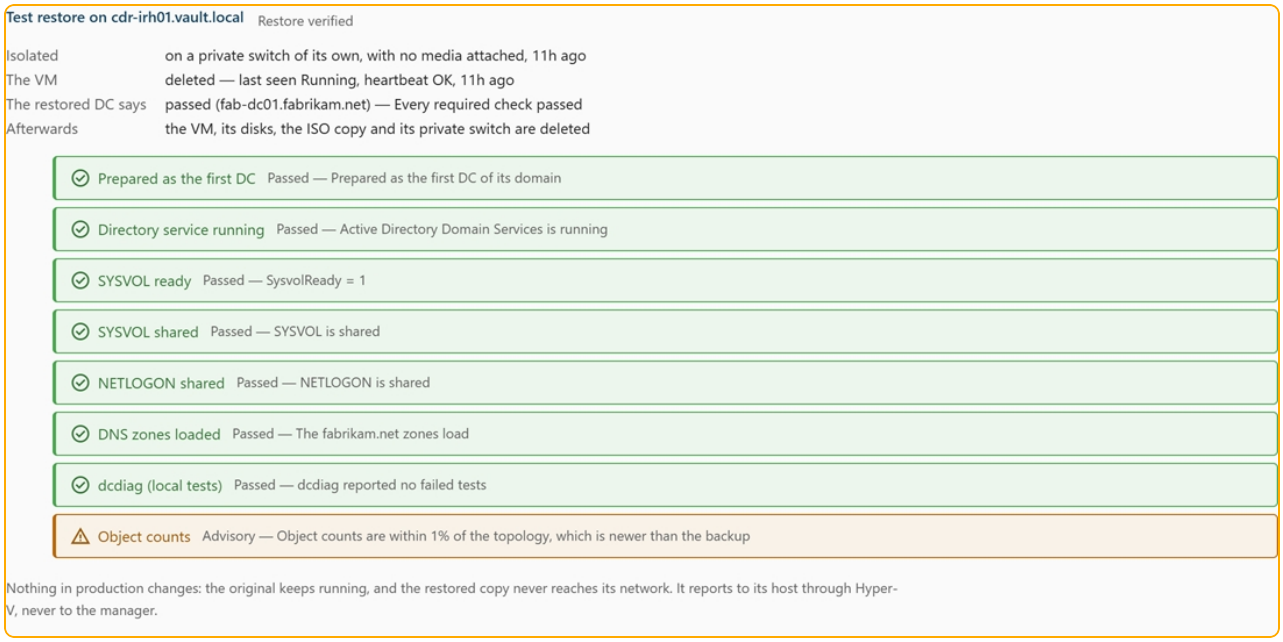
The run's page shows a **Test restore on *host*** card while the test runs:

- **Isolated**—Whether the virtual machine has been moved to its private switch with no media attached. Until then it reads that the restore is still running on the host's recovery network.
- **The VM**—The virtual machine's state and whether Hyper-V receives its heartbeat. Once the virtual machine has been deleted, it reads **deleted**, followed by what the host last saw and when.
- **The restored DC says**—The latest step the restored domain controller reported. While its checks are still running, it says how many required checks pass, which are still failing, and until when it keeps trying.

- **As of**–Shown only when the host cannot read the virtual machine or has stopped reporting on it. It gives the Manager's account of why, and that the restored domain controller may be further along than the card shows.
- **Afterwards**–Whether the virtual machine, its disks, the ISO copy, and the private switch were deleted.

Below them, each check is listed with its result, colored by that result: **Passed**, **Failed**, **Advisory**, or **Could not be checked**. When the test ends, the card shows the verdict, **Restore verified** or **Restore failed**, and for a failure, the step it stopped at and why. **No verdict yet** means the test has not finished, or ended without one; the card then gives the reason.

The **What the machine reported** card records each change as it happens: when the host is asked to isolate the virtual machine, each check's result as it changes (**Restored DC check**), changes in the virtual machine's heartbeat between reports (**VM heartbeat changed**), and the host's own problems, such as **The host cannot read the VM**, **The host stopped reporting on the VM**, and **The host is reading the VM again**.



To stop a test restore, select **Ask the machine to stop…** on the run's page, or run `cdr restore cancel`. A test restore can be stopped at any point. The virtual machine is deleted and the backup stays unverified.

From the command line

Plan the test restore, approve it, run it, and follow it:

```
cdr restore verify dc01.corp.example.com --version 09/08/2026-02:31
cdr restore approve <run-id>
```

```
cdr restore run <run-id>
cdr restore show <run-id> --wait
```

`cdr restore verify` is the same as `cdr restore plan --verify` and takes the same options. Use `--host` to choose the Isolated Recovery Host by its agent ID, when you have more than one, and `--image` to start from a particular recovery image. `cdr recovery-host list` shows each host and what stops it running a test restore. For details, see Restore Commands.

If the test restore fails

- **Restore failed**—The verdict names the step and the reason. Test a newer backup of the same domain controller. If a check failed, investigate the domain controller the backup came from; the same problem may be present in production.
- **Restore failed at the first-DC preparation**—The check's detail names the cause, for example a Group Policy setting that stops the domain controller serving time. Fix it on the source domain controller: the same cause would stop a real recovery.
- **No verdict**—The host could not do its part, for example it could not build or isolate the virtual machine, or it stopped reporting on it (step `host`). Check the host's panel on the **Recovery hosts** tab of the Agents screen, fix what it reports, then run the test again.
- **The plan is refused**—No Isolated Recovery Host can run it now. The message says why, for example that the host is not ready or is already running a test restore. Only one test restore runs on a host at a time.

For the checks that can stop a test restore before it starts, see Preflight Check Reference.

Excluding and Retiring Objects

Cygna Forest Recovery counts every forest, domain, site, and domain controller it knows about when it decides whether the estate is covered, and raises an alert when one is not protected. Two tools keep those counts honest. Use an **exclusion** when an object still exists but is deliberately out of scope, such as a test domain or a domain controller that is about to be decommissioned. A domain controller you simply do not back up needs no exclusion: it is rebuilt on recovery and raises no alert. Use a **removal record** (retirement) when an object no longer exists in the directory, so that it stops counting.

Neither changes anything in Active Directory, and neither hides the object. An excluded or removed object is still listed, with a chip that says why.

Before you begin

- You need the Backup Operator role. Without it, the exclusion and removal actions are not shown.

- Have a reason ready. Every exclusion and every manual removal record needs one. It is recorded against your name, shown wherever the object appears, and kept in the audit trail.

Exclude an object

An excluded object leaves the counts, verdicts, and alerts, but stays on every screen. Excluding a forest, domain, or site also excludes the objects in it; those show **Excluded via its domain**, **Excluded via its site**, or **Excluded via its forest**.

To exclude an object:

1. Go to the screen for the object: **Active Directory** > **Forests**, **Domains**, **Sites**, or **Domain Controllers**.
2. Open the actions menu on the object's row and select **Exclude from alarms…**.

   On **Domain Controllers**, you can also select several rows and then select **Exclude *n* from alarms…** in the bar at the bottom of the screen.
3. In **Why is *object* out of scope?**, enter the reason.

   Write it for whoever reads it next, for example "Orphaned partition — no DC has existed since the 2019 migration".
4. Optionally, in **Review by (optional)**, enter or pick a date.

   After that date, the object counts again and its exclusion is marked as due for review. Leave the field empty to exclude the object permanently. A review date is recommended: it stops a temporary decision from becoming permanent by accident.
5. Select **Exclude from alarms**.

   The object shows the **Excluded** chip. Hover over the chip to see the reason, who set it, and the review date.



Review, change, or clear an exclusion

An exclusion stops working in two cases, and the object starts counting and raising alerts again:

- **Due for review**—its review date has passed. The chip reads **Exclusion due for review**.
- **Void**—the object it was written about has changed identity since the exclusion was recorded. The chip reads **Exclusion void — the object changed**.

Both raise the alert **Exclusions due for review, or voided**. An exclusion on a domain controller or domain that has since been recorded as removed is not due for review, whatever its date: nothing can alert for an object that is gone. Its chip reads **Exclusion spent — object removed**, as a notice, and the alert leaves it out. Clear it when you no longer need the record.

To change or clear an exclusion:

1. Open the actions menu on the object's row and select **Edit exclusion…**.
2. Do one of the following:
   - To extend the exclusion, update the reason and the review date, and then select **Update exclusion**.
   - To bring the object back into scope, select **Clear exclusion**. The object counts again from the next read.

An inherited exclusion can be cleared only on the object that holds it. The dialog names that object. On **Domain Controllers**, **Clear *n* exclusions** clears several at once.

Objects removed from the directory

When every agent with a current, complete view of the directory stops seeing a domain controller, domain, or trust, discovery records it as removed. It leaves the counts and alerts, and the **Domain Controllers** screen lists it only under the **removed** chip. Each new removal raises the alert **Objects removed from the directory** for seven days. Confirm that each removal was intended: an unexpected one can be a sign of an attack.

Discovery does not record removals on its own when more than a quarter of a forest's domain controllers or domains, or of a domain's trusts, disappear at once. It holds them and raises **Mass removals from the directory held for review**. Confirm in the directory that each object really is gone, and then record each one with `cdr cleanup retire`. If they are not gone, find out why the agents stopped seeing them.

Note: A removal record changes only what Cygna Forest Recovery counts. To remove a dead domain controller's objects and metadata from Active Directory itself, use `cdr cleanup dc`. See Directory Cleanup Commands.

Mark a domain controller as removed

Use this when you know a domain controller is gone and do not want to wait for discovery.

To mark a domain controller as removed:

1. On the **Domain Controllers** screen, open the actions menu on its row and select **Mark as removed from the directory…**.
2. In **Why it is gone**, enter the reason.
3. Select **Mark as removed**.

   If a later enumeration includes the machine again, the record is withdrawn automatically.

If the machine still has a backup policy, remove the policy as well. Otherwise its backups are still dispatched. See Managing Backup Policies.

Withdraw a removal record

If a domain controller was recorded as removed by mistake, select **Withdraw removal record…** on its row (shown while the **removed** chip is selected), and then select **Withdraw the record**. The machine is listed and counted again.

Forget a domain controller

Forgetting deletes the Manager's record of a removed domain controller. Use it to clear out machines that never had anything worth keeping. It is offered only when the machine is

recorded as removed, no agent is enrolled for it, no backup jobs exist for it, and no backup policy names it. Otherwise the menu explains which of these keeps it.

To forget a domain controller:

1. On the row of the removed domain controller, select **Forget this machine…**.
2. Enter the machine's FQDN to confirm, and then select **Forget it**.

   The audit trail keeps a copy of what was deleted. If the directory ever defines the name again, it is discovered as a new domain controller.

Using the command line

```
cdr exclusion add dc dc09.corp.example.com --reason "Decommissioning, see change 4411" --review-by 2026-12-31
cdr exclusion add domain test.corp.example.com --reason "Test domain, not protected" --permanent
cdr exclusion list --spent
cdr exclusion remove dc dc09.corp.example.com
cdr cleanup retire dc09.corp.example.com --reason "Decommissioned"
cdr cleanup unretire dc dc09.corp.example.com
cdr cleanup forget dc09.corp.example.com --confirm dc09.corp.example.com
```

`cdr exclusion add` needs exactly one of `--review-by` or `--permanent`. `cdr cleanup retire` also records domains and trusts with `--kind domain` or `--kind trust`, and the Manager refuses it while a current, complete read of the directory still lists the object. For every option, see Configuration and Exclusion Commands and Directory Cleanup Commands.

Next steps

- Check the effect on readiness on the Dashboard. Excluded forests are noted there but not shown.
- For how exclusions affect verdicts, see Coverage and Readiness.

Recovery Procedures

This chapter gives the step-by-step procedures for recovering Active Directory with Cygna Forest Recovery, from one domain controller to a whole forest. Each procedure lists what you need before you start, the exact console steps, what happens while the recovery runs, how to stop it, and the equivalent `cdr` commands. To decide which procedure fits, read Choosing a Recovery Approach first.

Pick the procedure

Situation	Procedure
Changes to accounts, groups, or other objects have replicated across a healthy domain and must be undone, for example a mass password reset. A domain controller of the domain still starts Windows, and you have its backup from before the changes.	Restoring a DC with DC State Recovery
One domain controller's directory or Windows installation is damaged or untrustworthy, but the machine can still start Windows once.	Full System Recovery in Place
One domain controller will not start, or its hardware or virtual machine is gone.	Full System Recovery onto a Candidate, or, on Hyper-V, Full System Recovery onto a New VM
Every domain controller of one domain is lost or compromised, and the forest root and the other domains are still running.	Recovering a Domain
The forest root domain is lost, the whole forest is compromised, or the forest has only one domain.	Recovering a Forest

A single-DC full system recovery is non-authoritative: the machine returns to its backup and then catches up from its healthy peers. A single-DC DC state recovery is authoritative for the subtrees and objects you name: they go back to the backup on every domain controller of the domain, and everything else catches up from the peers. Do not use either when the whole domain is suspect. If a domain controller cannot be restored, for example a read-only domain controller or one whose operations master roles were seized by another domain controller after its backup, rebuild it instead: remove it from the directory and promote a new domain controller. For more on the scopes, see Recovery Scopes.

How every recovery runs

All procedures follow the same stages, shown as a stepper at the top of the recovery's page:

1. **Plan**—you choose what to recover, from which restore point, and onto which machine. Planning writes nothing to any machine.
2. **Preflight**—the Manager checks the plan against the restore point and the live estate. Hard checks block approval; advisory checks tell you what the recovery costs; a restore can also raise checks the approver must accept by name. A domain or forest plan's checks of candidate machines run again at approval. See Preflight Check Reference.
3. **Approve**—an operator with approval rights records that the plan has been reviewed.
4. **Execute**—an operator with execution rights starts the recovery and confirms it by typing a name.
5. **Monitor** (or **Run**)—you follow progress and respond when the recovery waits for you. A domain or forest plan can instead run unattended, and can email its halts, waits, and final recovery report.

Approval and execution are separate capabilities. The Restore Operator role holds both; no other role, including System Admin, can run a recovery. You can split them between two people so that every recovery is seen by two operators. Every action is recorded in the audit trail. For the full lifecycle, see Recovery Lifecycle.

While a recovery runs, a banner on every other console screen links to it and says when a recovery is waiting for you.

Before any recovery

- Read Recovery Requirements and confirm the requirements for your procedure. For a domain or forest recovery, also prepare the isolation described in Isolating Recovery Targets.
- Know what the recovery does not bring back, such as certificate services or DHCP on the same machine. See What Recovery Does Not Cover.
- If the incident is an attack, choose restore points from before the compromise.

In this chapter

- Restoring a DC with DC State Recovery
- Full System Recovery in Place
- Full System Recovery onto a Candidate

- Full System Recovery onto a New VM
- Recovering a Domain
- Recovering a Forest
- After a Recovery

Restoring a DC with DC State Recovery

A DC state recovery undoes the changes made to chosen parts of Active Directory since a restore point, on every domain controller of the domain. You name the subtrees and objects to roll back. The DC agent on one domain controller copies the backup to the machine's own disk, restarts it into Directory Services Restore Mode (DSRM), and restores DC state with `wbadmin`. Still in DSRM, it marks each subtree and object you named authoritative with `ntdsutil`, and then restarts the machine normally. The marked objects replicate out to every other domain controller as they were at the restore point. Everything else on this domain controller converges from its replication partners, as after any restore.

Use it when a change has already replicated across the domain and must be undone: for example, a mass password reset, or accounts and group memberships changed in bulk. Every change made to the named objects since the backup is undone, including attributes, password resets, and group memberships. It is a supported Windows system state restore with wbadmin, not a virtual machine snapshot revert, so it does not cause USN rollback. It works from both full system and DC state backups.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute.
- The domain controller starts Windows and its DC agent is connected. The console lists only domain controllers with a connected agent. The DC agent must be as current as the Manager and run as its Windows service, so that it can hold back the restart wbadmin starts; otherwise preflight blocks approval.
- You know the distinguished names of the subtrees and objects to roll back, as they were when the backup was taken.
- The domain controller is not a read-only domain controller, and its operations master roles were not seized by another domain controller after the backup. Either case means it must be rebuilt, not restored.
- You have a restore point that:
  - was taken on the same Windows build and revision the machine runs now. If the machine was patched since the backup, DC state cannot go back onto it.

- is younger than the domain's tombstone lifetime.
- has a wbadmin version identifier.
- The domain replicates SYSVOL with DFS Replication.
- The machine has free space for a local copy of the backup, about 1.15 times its size. The agent copies the backup to the machine before it restarts into Directory Services Restore Mode, and the restore runs from that copy. See **Restore staging volume** in Settings Reference.
- No backup is running on the machine. Let it finish or stop it first.

For the full list, see Recovery Requirements.

Note: Anything written to this domain controller after the restore point that has not replicated to another domain controller is lost.

Choose what to roll back

Name only what should go back. Each entry is one of:

- **Subtree**–the object and everything beneath it, for example `OU=Sales,DC=corp,DC=example,DC=com`.
- **One object**–a single object, for example `CN=Bob,OU=Sales,DC=corp,DC=example,DC=com`.

Objects created since the backup are not in it, and are left as they are. A rollback names at most 50 entries; plan a larger one as several runs. The Manager checks every entry when you plan, and the DC agent checks them again before it marks anything. The **Not accepted, and why** list on the planning page shows what is refused:

Not accepted	Why
A partition's head (`DC=…`, `CN=Configuration,…`)	It would roll back every password and trust in the domain. Name the subtrees or objects instead.
The schema	It cannot be restored authoritatively.
A `krbtgt` account on its own	It holds nothing but the old Kerberos keys. To undo a password reset, roll back `CN=Users` instead, as described below.
`OU=Domain Controllers`, and anything in it	Old machine passwords break every domain controller's secure channel. A domain controller is restored or rebuilt, not rolled back by object.
`CN=System` and the objects	Trusts and AdminSDHolder live there. Group Policy

Not accepted	Why
directly in it	objects under `CN=Policies` can be rolled back.
Anything outside this domain controller's domain or the forest's configuration partition	This domain controller cannot mark it.

The Manager also refuses an entry that is not a distinguished name, one that contains a double quote (mark its parent instead), one named twice, and one that lies inside a subtree already named. Planning is refused if the Manager has not yet recorded which domain the domain controller belongs to; wait until the domain controller is listed under its domain on the **Domains** screen, and plan again.

Group Policy objects roll back in the directory but not in SYSVOL. SYSVOL is restored non-authoritatively, so the policies' files come back from the other domain controllers as they are now. Preflight warns you; once the rollback has replicated, restore the policies from a Group Policy backup, for example with `Restore-GPO`.

Rolling back CN=Users and krbtgt

To undo a mass password reset, roll back the `CN=Users` container, or any subtree that holds the domain's `krbtgt` account. That puts `krbtgt` back to the backup too. The domain's Kerberos keys go back on every domain controller, including the read-only domain controllers' `krbtgt` accounts, so tickets issued since then stop working and users and services sign in again. Every user in the subtree gets the password they had at the backup, including anyone who changed it since.

An old key may be one an attacker holds. When a subtree holds `krbtgt`, the planning page offers **Reset krbtgt twice once the rollback validates**, selected by default. With it selected, Cygna Forest Recovery resets `krbtgt` twice once the rollback validates, so the backup's keys stop validating tickets. The second reset waits until the first has reached every writable domain controller. Approving then asks for a domain administrator. If you clear it, approving asks for no credentials, and you reset `krbtgt` twice yourself once the rollback has replicated, the second time after the first has reached every writable domain controller.

Plan the restore

To plan a DC state recovery:

1. Go to **Recovery** > **Recovery Plans**.
2. Under **Active Directory recovery**, select **Single-DC recovery**.

   The **New single-DC restore** page opens.
3. In **Recovery mode**, select **DC state**.

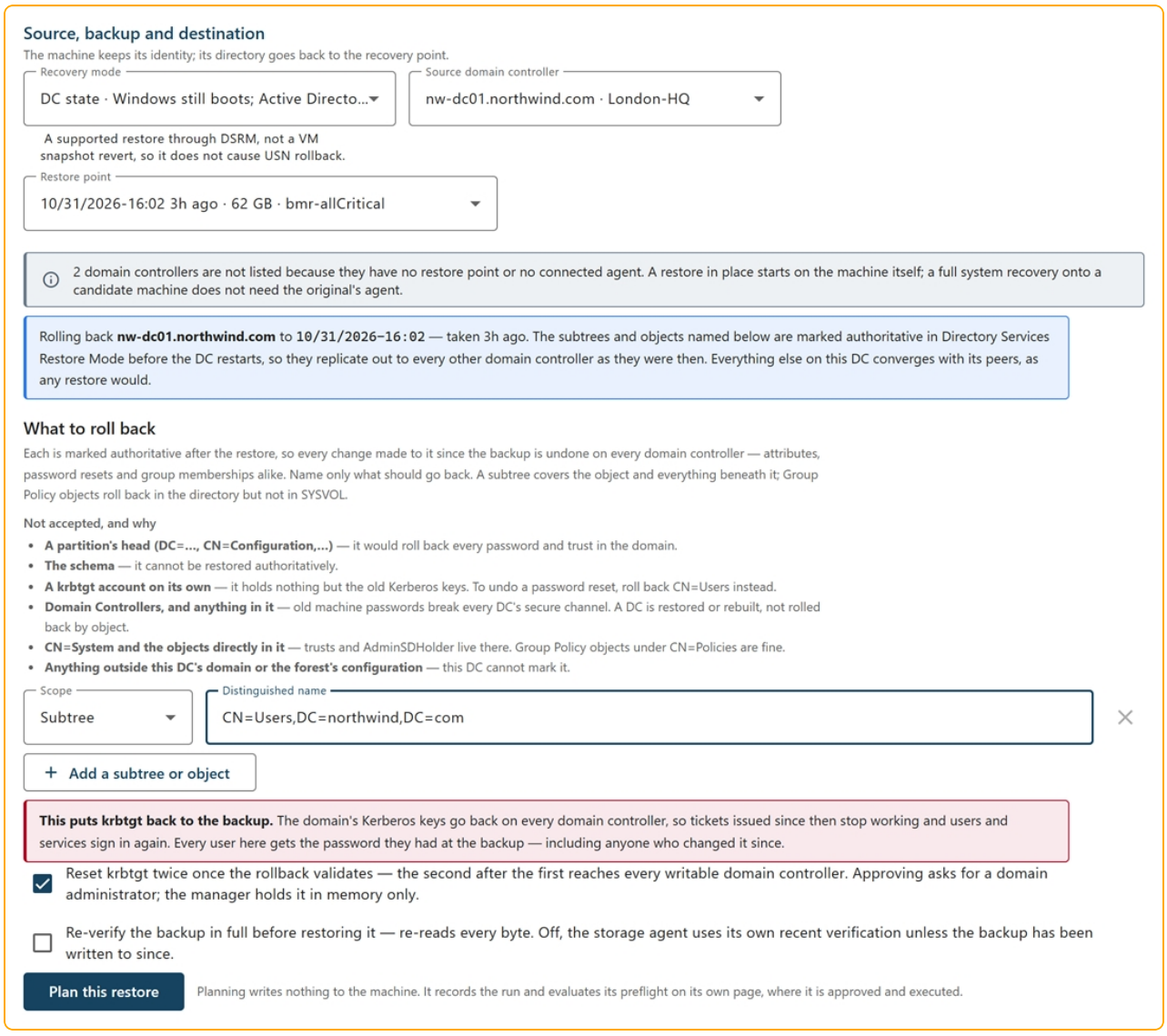
4. In **Source domain controller**, select the machine.
5. In **Restore point**, select the backup to restore.

   The newest usable restore point is preselected. Restore points that cannot be used are listed but unavailable, with the reason. Below the fields, the page states what the rollback will do and from when.
6. Under **What to roll back**, select **Subtree** or **One object** in **Scope**, and enter the **Distinguished name**.

   To name another, select **Add a subtree or object**. To remove one, select its **Remove** button.
7. If a subtree holds `krbtgt`, read the warning that appears, and leave **Reset krbtgt twice once the rollback validates** selected, or clear it to reset `krbtgt` yourself.
8. Optionally, select **Re-verify the backup in full before restoring it**.

   This re-reads every byte of the backup against its recorded hash. When it is cleared, the storage agent relies on its own recent verification unless the backup has changed since.
9. Select **Plan this restore**.

   The button is available once at least one distinguished name is entered. If the Manager refuses an entry, the page says why. The restore's own page opens with the status **Planned**. Its **Rolled back — marked authoritative** card lists each subtree and object. Nothing has been sent to the machine.



Approve and execute

To approve and execute the restore:

1. On the restore's page, review the **Preflight** card.

   Resolve every hard check. Each one shows a remedy. Read the advisory checks: they do not block approval, but they say what the restore costs.

2. Accept the rollback's own checks, after reading them:

   - **The rollback overwrites these on every domain controller** (`authoritative.overwrites`) lists what is rolled back. Select **I have read this and want these rolled back on every domain controller**.

- When a subtree holds `krbtgt`, **The rollback puts krbtgt back to the backup, then resets it twice**, or **…, and nothing resets it** (`authoritative.krbtgt`). Select **I have read this and accept that krbtgt and these users' passwords go back to the backup**.

3. If the rollback resets `krbtgt`, enter a **Domain administrator**, as `DOMAIN\user` or `user@domain`, and its **Password**.

   The Manager holds the credentials in its memory only, for the reset. They are not stored on the run or written to the audit trail, and the Manager forgets them when the reset is done or fails, 8 hours after they were given, or when the Manager restarts. If the Manager already holds credentials for the run, the page says so, and credentials you enter replace them.

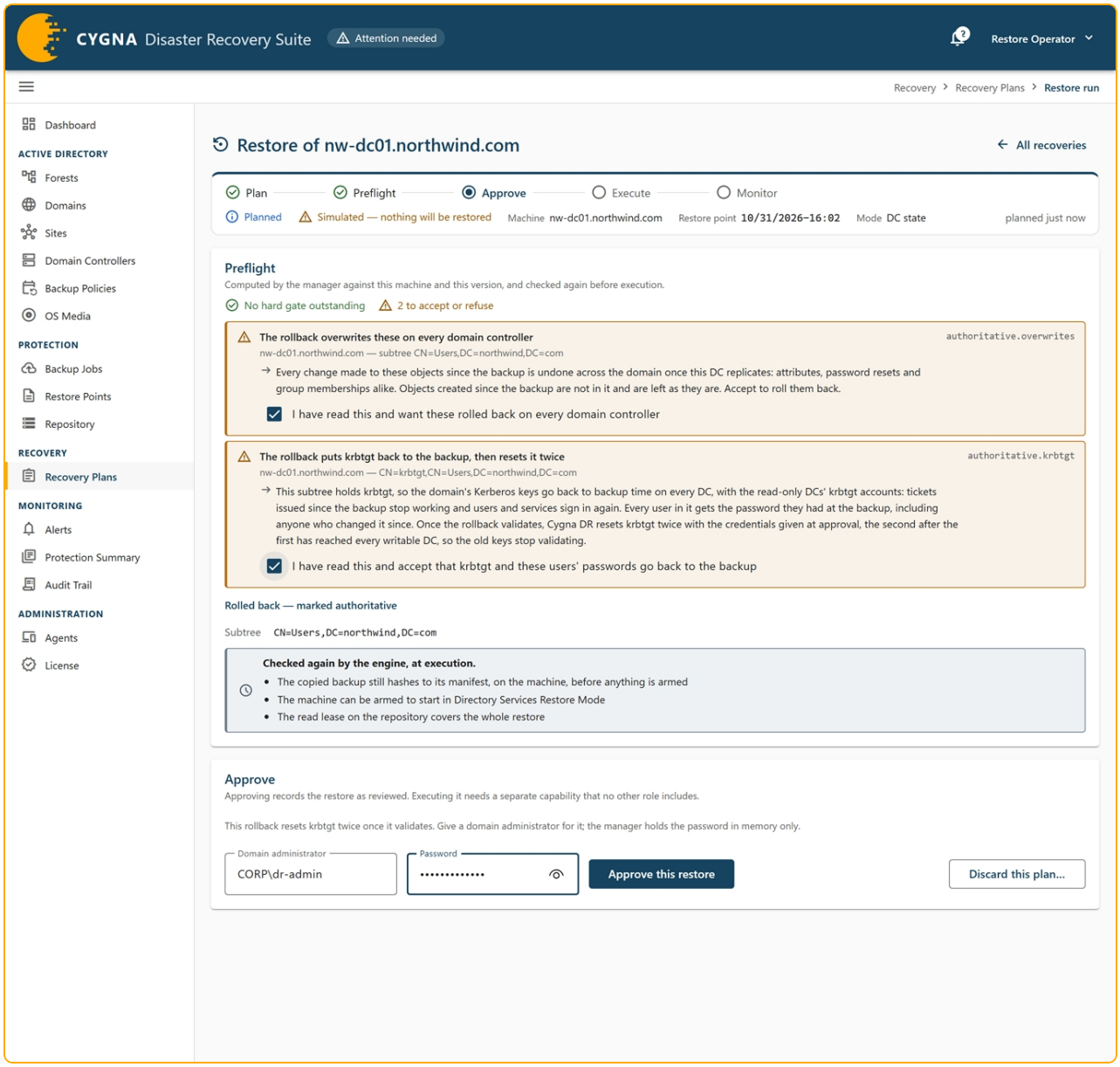
4. Select **Approve this restore**.

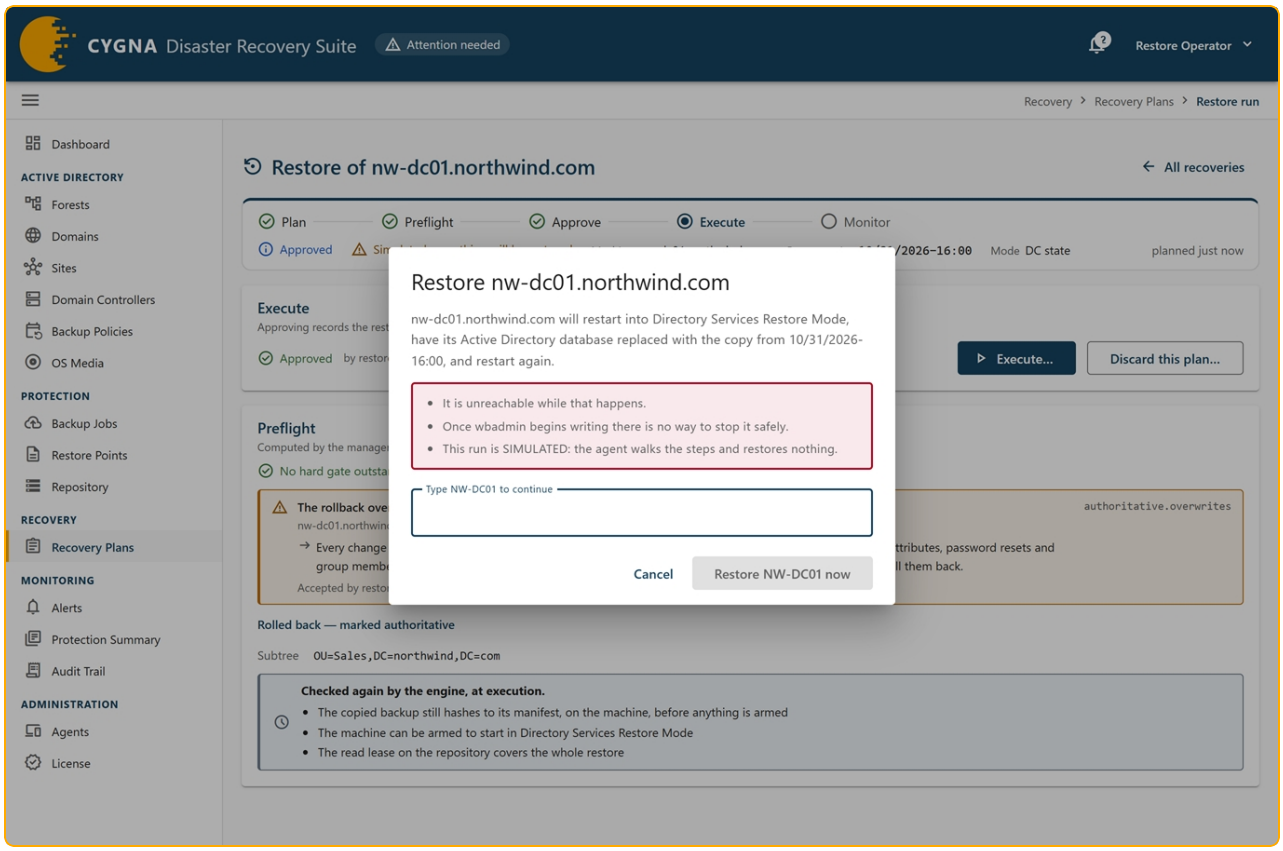
   The button is available only when no hard check is outstanding, every check that needs your acceptance is accepted, and, for a `krbtgt` reset, the credentials are entered. The card shows who approved it. To take the approval back, select **Withdraw approval**.

5. Select **Execute…**.

   The confirmation says the machine will restart into Directory Services Restore Mode, have its Active Directory database replaced, and restart again.

6. Enter the machine's short name in capitals, for example `DC01`, and select **Restore DC01 now**.


CYGNA Disaster Recovery Suite
Attention needed
Restore Operator
Recovery > Recovery Plans > Restore run
Dashboard
ACTIVE DIRECTORY
Forests
Domains
Sites
Domain Controllers
Backup Policies
OS Media
PROTECTION
Backup Jobs
Restore Points
Repository
RECOVERY
Recovery Plans
MONITORING
Alerts
Protection Summary
Audit Trail
ADMINISTRATION
Agents
License
Restore of nw-dc01.northwind.com
All recoveries
Plan
Preflight
Approve
Execute
Monitor
Planned
Simulated — nothing will be restored
Machine nw-dc01.northwind.com
Restore point 10/31/2026-16:02
Mode DC state
planned just now
Preflight
Computed by the manager against this machine and this version, and checked again before execution.
No hard gate outstanding
2 to accept or refuse
The rollback overwrites these on every domain controller
authoritative.overwrites
nw-dc01.northwind.com — subtree CN=Users,DC=northwind,DC=com
Every change made to these objects since the backup is undone across the domain once this DC replicates: attributes, password resets and group memberships alike. Objects created since the backup are not in it and are left as they are. Accept to roll them back.
I have read this and want these rolled back on every domain controller
The rollback puts krbtgt back to the backup, then resets it twice
authoritative.krbtgt
nw-dc01.northwind.com — CN=krbtgt,CN=Users,DC=northwind,DC=com
This subtree holds krbtgt, so the domain's Kerberos keys go back to backup time on every DC, with the read-only DCs' krbtgt accounts: tickets issued since the backup stop working and users and services sign in again. Every user in it gets the password they had at the backup, including anyone who changed it since. Once the rollback validates, Cygna DR resets krbtgt twice with the credentials given at approval, the second after the first has reached every writable DC, so the old keys stop validating.
I have read this and accept that krbtgt and these users' passwords go back to the backup
Rolled back — marked authoritative
Subtree CN=Users,DC=northwind,DC=com
Checked again by the engine, at execution.
The copied backup still hashes to its manifest, on the machine, before anything is armed
The machine can be armed to start in Directory Services Restore Mode
The read lease on the repository covers the whole restore
Approve
Approving records the restore as reviewed. Executing it needs a separate capability that no other role includes.
This rollback resets krbtgt twice once it validates. Give a domain administrator for it; the manager holds the password in memory only.
Domain administrator
CORP\dr-admin
Password
Approve this restore
Discard this plan...



Note: **Important:** Once wbadmin begins writing, the restore cannot be stopped safely. The machine is unreachable while it restores.

What happens during the restore

Before anything changes, the Manager checks again that the backup still matches its recorded hash, that the version is present on the machine, and that the machine's boot configuration can be set and cleared. For a DC state recovery, the agent copies the backup to the machine's own disk and verifies it there, so the restore does not depend on the network in DSRM. The copy shows its progress on one line, by bytes copied. The agent also keeps a copy of its own installer package, and raises Windows' service start limit to 180 seconds so that it starts in time in DSRM; the limit is put back when the run ends. It then arms a one-time start into DSRM and restarts the machine; if anything goes wrong, the next start is a normal one. wbadmin restores DC state, and from that point the restore cannot be stopped.

wbadmin starts a restart of the machine as it exits. The agent holds that restart back until it has recorded the restore's outcome and marked every entry, for at most 2 hours per entry still to mark plus 15 minutes, and then lets it go ahead. Nothing is stopped while the restart waits. An agent that cannot hold back a restart refuses the rollback before anything is changed.

Still in DSRM, the agent marks each subtree and object authoritative with `ntdsutil`, which raises their version numbers so that they win over the copies on the other domain controllers. It tries every entry, even after one fails, and takes the object list `ntdsutil` writes as the evidence of each mark. `ntdsutil` also writes back-link files: the group memberships of the marked objects that are held on objects that were not marked. The agent records each file's hash before the machine leaves DSRM.

After a normal restart, the agent imports the back-link files with `ldifde` against the restored domain controller, using only the recorded files and only while each still matches its hash, so group memberships come back too. It then reads the marked objects' replication metadata to confirm the raised versions. The Manager checks that AD DS starts, that the marks and back-links are in place, that replication with inbound and outbound partners converges, and that dcdiag is clean. The run is shown as restored only when the marks have been read back.

A check the agent could not make shows **Not checked** on the run's **Steps** card. To make it again, select **Validate again** or run `cdr restore validate`. For a DC state recovery, this also imports any back-link file that failed and reads the marks again. For each status the run shows along the way, see Recovery Lifecycle.

If an entry could not be marked, the machine still restarts normally, and the run fails and names the entries that were not marked. Those come back from the other domain controllers as they are now; entries that were marked replicate out. To try again, plan a new DC state recovery.

In DSRM, and on the normal start until its checks are done, the agent reports live when the machine can reach the Manager: the run shows **Restoring**, with wbadmin's progress on one line. Without a network path, the page says how long it expects the machine to be unreachable, and silence during that time is expected. Either way, the agent reports the whole timeline when it returns. The **What the machine reported** card shows each phase and the end of the transcript, which also holds wbadmin's recovery summary.

If an agent that reports live has not been heard 10 minutes after the restart into DSRM, the run's timeline shows **No word from the agent in restore mode**. Check the machine's console, and start the `CygnaDRDcAgent` service if it is not running. The run itself is not changed.

On the first normal start after the restore, the agent checks that Windows Installer can still update it. If the agent was updated after the backup was taken, it reinstalls the version that was running before the restore, and its service restarts once. See Installer Messages.

If the run stops with **Stopped — needs an operator** and the page shows **This machine needs you**, do not start another restore of the machine until somebody has looked at it. Its directory may be inconsistent. Read the note and transcript the agent left on the machine, and then either restore again from a different restore point or rebuild the domain controller.

The krbtgt reset

For a rollback planned with the `krbtgt` reset, the **Steps** card has a step **R-04** that starts once the rollback validates. The agent on the restored domain controller resets `krbtgt` as the domain administrator given at approval, then waits until the first reset has reached every writable domain controller of the domain before it makes the second. A domain controller that cannot be read counts as not reached. If the first reset has not reached every writable domain controller within 30 minutes, the second is not made, and the step names the domain controllers that are behind. Resetting again before a domain controller has the first key would leave it rejecting tickets.

The run stays completed whatever happens to the reset: the rollback itself is restored. If the reset fails, or the Manager no longer holds the credentials because 8 hours passed or the Manager restarted, the step fails and the run shows that it needs an operator.

To start the krbtgt reset again:

1. On the run's page, find **krbtgt has not been reset twice yet**.
2. Enter a **Domain administrator** and its **Password**.
3. Select **Reset krbtgt**.

   This needs execution rights. It does only what is left: if the first reset was applied, only the second is made.

From the command line, run `cdr restore reset-krbtgt <run> --admin DOMAIN\user`.

Stopping or discarding a restore

Discarding and withdrawing approval need approval rights; stopping and abandoning need execution rights.

- **Discard a plan you have not executed**—select **Discard this plan…**, and then **Discard plan**. The run is closed as cancelled and the machine can be planned again at once. Use this for a plan whose preflight failed, for example.
- **Stop an executing restore**—select **Ask the machine to stop…**, and then **Ask it to stop**. The machine abandons the restore only if wbadmin has not started writing. After that, the Manager refuses. Once the machine has restarted into DSRM and its agent is away, the Manager also refuses, because the restore may be running there.
- **Close a restore whose machine is gone**—if the restore is past the point where it can be stopped and the machine is gone or will be rebuilt, select **Close as abandoned…**, enter a reason, and select **Close as abandoned**. Nothing is sent to the machine. Do not use it as a domain controller until it is rebuilt or its transcript has been read.

Using the command line

```
cdr backup list --machine dc01.corp.example.com
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --authoritative-subtree
"OU=Sales,DC=corp,DC=example,DC=com"
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --authoritative-subtree
"CN=Users,DC=corp,DC=example,DC=com" --reset-krbtgt
cdr restore show <run>
cdr restore approve <run> --accept authoritative.overwrites
cdr restore approve <run> --accept authoritative.overwrites --accept authoritative.krbtgt --admin
CORP\admin
cdr restore run <run> --wait
cdr restore reset-krbtgt <run> --admin CORP\admin
cdr restore cancel <run>
cdr restore discard <run>
cdr restore abandon <run> --reason "Machine replaced"
```

`cdr restore plan` plans a DC state recovery by default, and needs at least one `--authoritative-subtree` or `--authoritative-object`; each can be repeated. Copy the version identifier exactly as `cdr backup list` prints it. With `--admin`, `cdr` prompts for the password, or reads it from the `CYGNADR_RESTORE_ADMIN_PW` environment variable. `cdr restore run` asks you to type the machine's name. For every option, see Restore Commands.

Result and next steps

The domain controller runs the same Windows installation. The subtrees and objects you named are as they were at the restore point on every domain controller of the domain, and everything else on this domain controller has caught up from its partners. Next:

- If the rollback reached Group Policy objects, restore their files from a Group Policy backup.
- If the rollback put `krbtgt` back and you did not choose the reset, reset `krbtgt` twice, the second time after the first has reached every writable domain controller.
- Take a fresh backup of the domain controller and complete the follow-ups in After a Recovery.
- For the screen itself, see Recovery Plans.

Full System Recovery in Place

A full system recovery in place restores a whole domain controller—the operating system, its patches, and its directory—from a full system backup onto the same machine. The DC agent downloads the recovery image for the machine's operating system, restarts the machine into it once, and the recovery agent in the image restores every critical volume

from the vault onto the existing disks. Use it when one domain controller's Windows installation is damaged or can no longer be trusted, but the machine can still start Windows one more time so its agent can stage the image. If the machine will not start at all, recover onto a candidate or onto a new VM instead (see Full System Recovery onto a Candidate and Full System Recovery onto a New VM). The restore is non-authoritative: the domain controller catches up from its healthy partners afterward.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute.
- The domain controller starts Windows and its DC agent is connected.
- A recovery image exists for the machine's operating system. Check the **Recovery image** column on the **OS Media** screen. If no image exists, see Setting Up OS Media and Recovery Images.
- You have a **Full system** restore point younger than the tombstone lifetime. A DC state backup cannot do a full system recovery.
- The machine's disks still match the backup. An in-place recovery restores onto the existing layout and cannot repartition, so a disk that was added, resized, replaced, or converted since the backup blocks it. Choose a backup taken since the change.
- The machine has enough memory to load the recovery image into a RAM disk (about the image size plus 1 GB), and free space on the system volume for the image (about 1.1 times its size).
- The machine can reach the storage machine over SMB at the storage agent's advertised address, which is its IP address by default. If the storage machine is advertised by name, the recovery image must be able to resolve it through DNS, which is the only name resolution it has.
- The machine's clock is within five minutes of the Manager's.
- BitLocker is suspended, or you have the recovery keys at hand.
- A DHCP server serves the machine's network for the duration, or you plan a fixed address for the recovery environment (see Network settings).

For the full list, see Recovery Requirements.

Note: Everything on the critical volumes goes back to the restore point. Anything written to this domain controller since then that has not replicated to another domain controller is lost, and so are patches and software installed since the backup.

Plan the recovery

To plan a full system recovery in place:

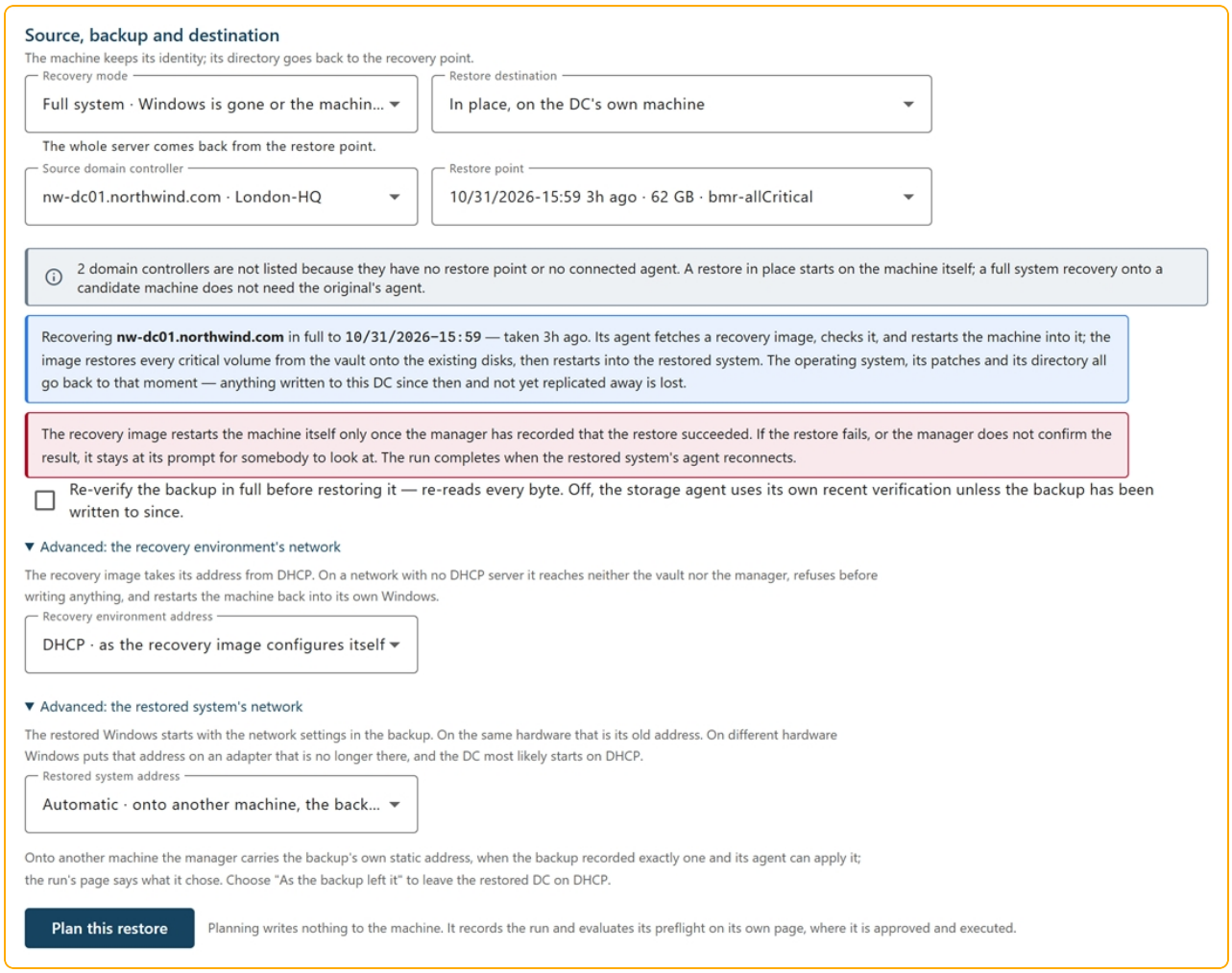
1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Single-DC recovery**.
2. In **Recovery mode**, select **Full system**.
3. In **Restore destination**, leave **In place, on the DC's own machine** selected.

   The other destinations are **Onto a new VM booted from a recovery ISO — nothing installed on it first**, a test restore on an Isolated Recovery Host, which changes nothing in production (see Running a Test Restore), and any candidate machines.

   In place, only domain controllers with a restore point and a connected agent are listed.
4. In **Source domain controller**, select the machine.
5. In **Restore point**, select a full system restore point.

   DC state restore points are listed but unavailable. The page states what the recovery will do.
6. Optionally, select **Re-verify the backup in full before restoring it**.
7. If the machine needs specific network settings, expand the advanced sections and set them. See Network settings.
8. Select **Plan this restore**.

   The restore's own page opens with the status **Planned**. Nothing has been sent to the machine.



Network settings

Full system recovery has two network choices: the address the recovery environment uses while it restores, and the address the restored system takes when it first starts. Both are under advanced sections on the planning page. The recovery environment uses DHCP by default, and the restored system's address is chosen automatically.

The recovery environment's network

By default the recovery image takes its address from DHCP. On a network with no DHCP server it reaches neither the vault nor the Manager, refuses before writing anything, and restarts the machine back into its own Windows.

Recovery environment address	When to use it
DHCP	The default. A DHCP scope serves the machine's network.
This machine's own address	The machine has a static address and there is no DHCP. The recovery environment uses the address the machine last reported.
Another address	The restore runs on a different subnet or an isolated segment. Enter **Address, with prefix** (for example `10.99.0.10/24`), and optionally **Gateway (optional)** and **DNS servers, comma-separated**. DNS servers are needed only if the storage machine is advertised by name; then give servers that resolve it.

If the machine has more than one network adapter, enter the one to use in **Adapter MAC (when it has several)**. A planned address is applied before the recovery environment contacts anything. If it cannot be applied, the run refuses before writing and the machine restarts into its own Windows. It never falls back to DHCP.

The restored system's network

The restored Windows starts with the network settings in the backup. Windows keeps a static address against the adapter it was set on, so if the machine's adapter is not the backed-up one—for example, a rebuilt virtual machine or a replaced card—the restored domain controller would come back on DHCP. **Automatic**, the default, prevents that where it can.

Restored system address	When to use it
Automatic	The default. The Manager puts the backup's own static address back—on the backup's adapter if the machine still has it, otherwise on its only adapter—when the backup recorded exactly one static address and the machine's latest description says which adapter to use. Otherwise nothing is changed. The run's page says what it chose.
As the backup left it	Nothing is changed. Choose it to leave a restored domain controller on DHCP.

Restored system address	When to use it
The backup's own address	The adapter has changed. The backup's static address is put back on the machine's adapter.
Another address	The domain controller moves to a new subnet, for example after a compromise. Enter **Address, with prefix**, and optionally a gateway and DNS servers—usually the domain controller itself or another domain controller on the recovery network.

Use **Target adapter MAC (when it has several)** to name the adapter. The restored domain controller's own agent applies the address at its first start, before it contacts the Manager. When the address changes, it re-registers the domain controller in DNS and restarts Netlogon. A failure is written to `POST-RECOVERY-README.txt` on the machine and reported when the agent connects.

Note: Moving a domain controller to a new subnet leaves work for you: a subnet object mapped to its site, old address records, delegation records in the parent zone, a reverse zone, and anything else that reaches the domain controller by its old address, such as forwarders, firewall rules, and DHCP authorization.

Approve and execute

To approve and execute the recovery:

1. On the restore's page, review the **Preflight** card and resolve every hard check.

   The card also shows the planned network settings and **The machine, as it described itself**: firmware, memory, disks, free space, and whether it can reach the storage machine. If you have changed the machine since planning, select **Ask the machine again**.

2. If the card shows checks to accept or refuse—for example, the machine cannot resolve the storage machine's name from where it is now—read each one's remediation. If you know the restore will get past it from where it runs, safely, select **I have read this and accept it: the restore will get past it, safely, from where it runs** under it. Otherwise, do not approve: discard the plan, or cancel it from its recovery plan.

   Approval stays unavailable until you accept each such check. A storage machine name is also a redirection point: whatever answers for it receives the read credential and supplies what is restored, so accept a name only if its DNS is trustworthy now. The lasting fix is to advertise the storage machine by IP address.

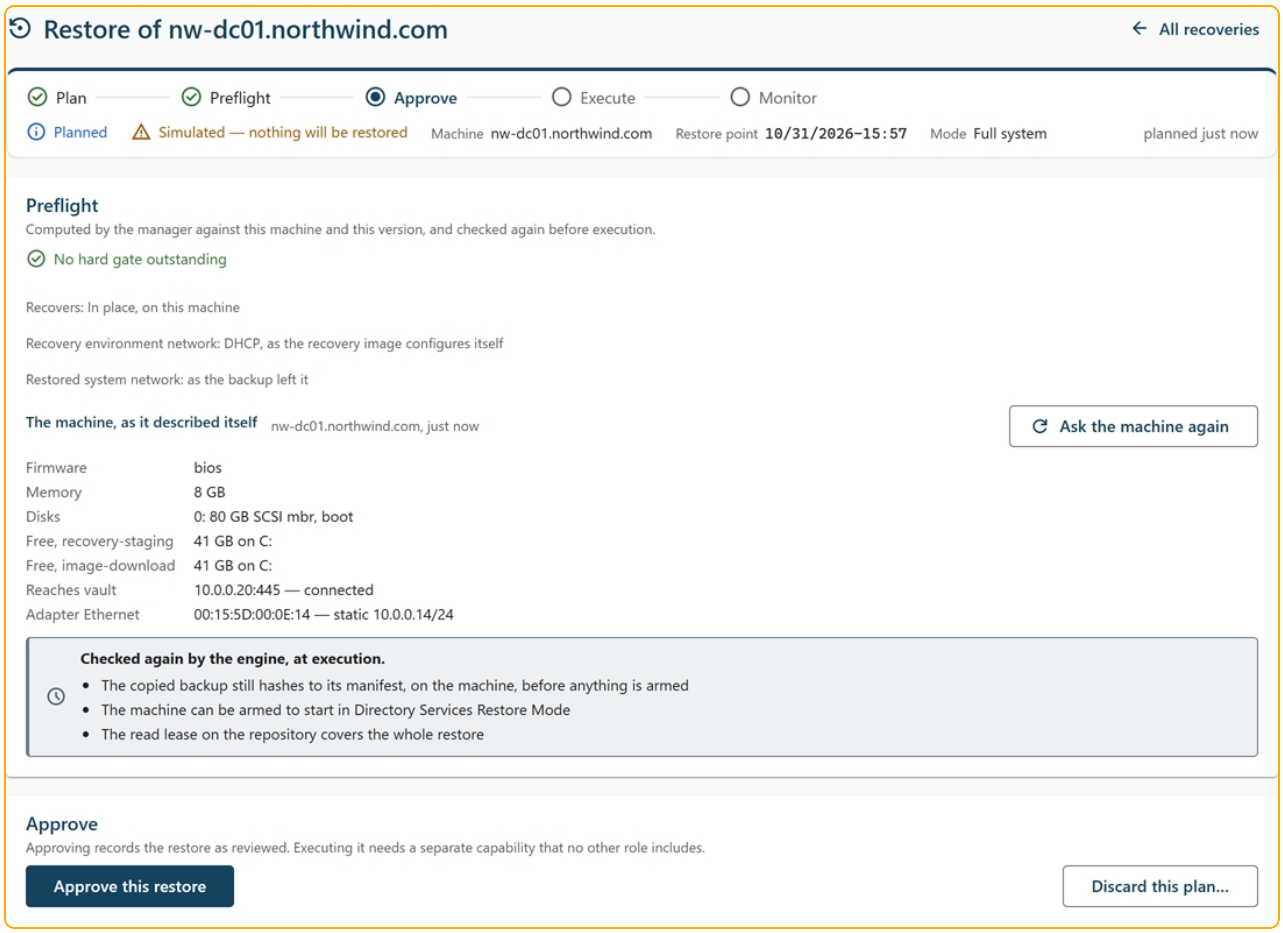
3. Select **Approve this restore**.

Each check you accepted is recorded on the run under your name.

4. Select **Execute…**.

   The confirmation says the machine will restart into a recovery image and have every critical volume overwritten.

5. Enter the machine's short name in capitals, for example `DC01`, and select **Restore DC01 now**.



Note: **Important:** Once wbadmin begins writing, the recovery cannot be stopped safely, and the machine's previous operating system is gone.

What happens during the recovery

The run's status moves from **Verifying the backup** through **Preparing the machine**, **Rebooting into restore mode**, **Restoring**, and **Validating** to **Restored**. The **What the machine reported** card shows the detailed phases, in this order:

1. **Fetching the recovery image** and **Recovery image staged on this machine**–the DC agent downloads the image, checks it, and copies it to the system volume.

2. **Armed to boot the recovery image** and **Restarting into the recovery image**—the agent arms a one-time start from the image and restarts the machine.
3. **Recovery environment started** and **Backup verified from the recovery environment**—the recovery agent applies the planned network settings, checks the firmware and disks it can see against the backup, and connects to the vault.
4. **Restoring volumes**—wbadmin restores every critical volume. From here the recovery cannot be stopped. The recovery agent reports each volume's progress. After a good restore, it replaces the restored DC agent with the build that staged the restore, so the domain controller does not come back on an older agent from its backup.
5. **Waiting for the restored system to start** and **Restored system started and reconnected**—once the Manager has recorded that the restore succeeded, the recovery environment restarts into the restored Windows.
6. **Validating**—the restored domain controller's own agent checks that AD DS is running and SYSVOL is ready, then waits up to 15 minutes for replication to converge and runs dcdiag. The run completes or fails on its answer.

The **Steps** card shows the result of each check. To check again, for example after fixing what made a check fail, select **Validate again** on the run's page, or run `cdr restore validate`. Validating again changes nothing on the machine; it needs `plan.approve`.

The machine's own agent is offline for several minutes before the recovery image reports in; the page says how long to expect. If the restore fails, or the Manager does not confirm the result, the recovery environment stays at its prompt for somebody to look at rather than restarting.

If the recovery image does not start, the machine returns to its own Windows at the next start, because the start from the image is one-time only. The run reports **The machine restarted into its own Windows, not the recovery image**, and nothing has been written.

Stopping or discarding a recovery

- **Discard a plan you have not executed**—select **Discard this plan…**, and then **Discard plan**.
- **Stop an executing recovery**—select **Ask the machine to stop…**, and then **Ask it to stop**. This works only before wbadmin starts writing.
- **Close a recovery whose machine is gone**—select **Close as abandoned…**, enter a reason, and confirm. Do not use the machine as a domain controller until it is rebuilt or its transcript has been read.

If the page shows **This machine needs you**, do not start another restore of the machine until somebody has looked at it. The ways forward are a fresh recovery from a different restore point or rebuilding the domain controller.

Using the command line

```
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --mode server
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --mode server --recovery-network original --restored-network unchanged
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --mode server --recovery-address 10.99.0.10/24 --recovery-dns 10.99.0.11
cdr restore survey <run>
cdr restore approve <run>
cdr restore approve <run> --accept target.vault-unresolved
cdr restore run <run> --wait
cdr restore validate <run>
```

The network options mirror the console: `--recovery-network` and `--restored-network` take `unchanged`, `original`, or `override`, with the address, gateway, DNS, and adapter options for an override. Omit `--restored-network` for the automatic choice. `cdr restore approve --accept` accepts one check by its code; repeat it for each. `cdr restore survey` asks the machine to describe itself again. Stopping, discarding, and abandoning use `cdr restore cancel`, `discard`, and `abandon`. For every option, see Restore Commands.

Result and next steps

The domain controller runs the operating system and directory from the restore point, on the same machine, and catches up from its partners. Next:

- Install any patches released since the backup, take a fresh backup, and complete the follow-ups in After a Recovery.
- For how recovery images work, see Recovery Images.

Full System Recovery onto a Candidate

A full system recovery onto a candidate restores a whole domain controller from a full system backup onto a different machine: a new virtual machine or a replacement server that runs Windows and has the DC agent installed, but is not a domain controller. The candidate's agent stages the recovery image and restarts the candidate into it, and the recovery agent restores the domain controller's critical volumes onto the candidate's disks. When it finishes, the candidate is the domain controller, with the original's name, identity, and directory as of the restore point. Use it when a domain controller will not start, or when its hardware or virtual machine is gone or must not be trusted. The restore is non-authoritative: the domain controller catches up from its healthy partners afterward. The storage machine and Isolated Recovery Hosts are never offered as candidates, even with a DC agent installed. To check a backup without touching a machine, run a test restore instead.

To recover onto an empty virtual machine instead, with no Windows or agent installed on it first, see Full System Recovery onto a New VM.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute.
- A recovery image exists for the original domain controller's operating system. Check the **Recovery image** column on the **OS Media** screen.
- You have a **Full system** restore point of the domain controller, younger than the tombstone lifetime.
- You have a candidate machine prepared as described below.
- You can power off or isolate the original domain controller. Two machines with one identity damage a domain rather than recover it, so the Manager does not approve the recovery while the original still answers.
- If volumes in the backup were protected by BitLocker, have the recovery keys at hand. A TPM protector belongs to the original machine.

For the full list, see Recovery Requirements.

Prepare the candidate

The candidate must be shaped like the backup:

- **Firmware**—the same firmware type as the original, UEFI or BIOS. A restored system on the other firmware does not start. On Hyper-V, a generation 2 virtual machine is UEFI and generation 1 is BIOS. Recovery onto either firmware type is supported.
- **Disks**—at least as many disks as the backup, each the same size or larger. wbadmin recreates each critical disk at its recorded size and refuses a smaller one, after the machine has already restarted into the recovery image.
- **Memory**—enough to load the recovery image into a RAM disk, about the image size plus 1 GB. Less memory than the original works but may run slowly.
- **Storage controller**—preferably one the restored Windows already has a driver for. Otherwise it stops at `INACCESSIBLE_BOOT_DEVICE`. If the candidate's boot controller differs from the original's, install its driver on the original domain controller before the backup. Built-in drivers cover common virtual controllers.
- **Network**—the candidate can reach the Manager and the storage machine. The storage machine is reached at its advertised address, its IP address by default; if it is advertised by name, the candidate must resolve it through DNS alone. The recovery environment needs a driver for the candidate's network adapter. Add any driver it lacks to the storage agent's **Driver folder**, so that it is built into the recovery image.

Create a virtual machine with cdr dc rebuild

`cdr dc rebuild` creates a virtual machine shaped from the domain controller's newest restore point—firmware generation, memory, processors, and disk sizes—and attaches matching installation media from the OS media library. On Hyper-V, it creates the machine on the Hyper-V host where you run the command, which must run elevated. For ESXi, it writes a PowerCLI script that you run in your own PowerCLI session; `cdr` never runs an ESXi build itself.

```
cdr dc rebuild dc01.corp.example.com --plan
cdr dc rebuild dc01.corp.example.com --switch "Recovery" --start
cdr dc rebuild dc01.corp.example.com --hypervisor esxi --datastore ds01 --vmhost esx01.-
corp.example.com --script C:\Temp\dc01-rebuild.ps1
```

`--plan` shows what would be built without creating anything. For the other options, including `--memory`, `--iso`, and `--manifest`, see Inventory Commands. With `--restore-run`, the same command builds a VM for a recovery onto a new VM, which needs no Windows installation; see Full System Recovery onto a New VM.

Install Windows and the DC agent

To make a machine a candidate:

1. Create the virtual machine with `cdr dc rebuild`, or prepare a virtual machine or server by hand that meets the shape above.
2. Install Windows Server on it. Do not promote it to a domain controller.
3. Install the DC agent on it with the hand-install command shown on the **Administration** > **Agents** screen, for example:

   ```
   msiexec /i CygnaDR.DcAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
   MANAGERCAFINGERPRINT=<fingerprint> ENROLLMENTTOKEN=<token>
   ```

   The agent enrolls and describes the machine's hardware to the Manager.
4. Confirm that the candidate is listed in **Restore destination** on the **New single-DC restore** page, or with `cdr recovery candidates`.

For installation details, see Deploying DC Agents.

Power off the original

Power off the original domain controller, or isolate it so that the candidate cannot reach it. The candidate checks every address the original used; while the original answers, preflight shows a hard check that blocks approval. If the original's hardware or virtual machine still exists, make sure it cannot be started again on the production network after the recovery.

Plan the recovery

Once you choose a candidate in **Restore destination**, the console lists every domain controller with a restore point, whether or not its own agent is connected; one whose agent is down is marked **agent not connected**. The original's agent is not needed for a recovery onto a candidate.

To plan a full system recovery onto a candidate:

1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Single-DC recovery**.
2. In **Recovery mode**, select **Full system**.
3. In **Restore destination**, select the candidate.

   The list also offers the domain controller's own machine, a new VM booted from a recovery ISO, and a test restore on an Isolated Recovery Host; the candidates follow them.
4. In **Source domain controller**, select the original domain controller, and in **Restore point**, select a full system restore point.
5. Optionally, expand **Advanced: the restored system's network** and check **Restored system address**.

   Windows binds a static address to the adapter it was set on, and the candidate's adapter is a different device. The default, **Automatic**, carries the backup's own static address onto the candidate's adapter when the backup recorded exactly one; the run's page says what it chose. Select **The backup's own address** to require it, **As the backup left it** to leave the restored domain controller on DHCP, or **Another address** if it moves to a new subnet. For all options, see Full System Recovery in Place.
6. If the candidate's network has no DHCP, expand **Advanced: the recovery environment's network** and set **Another address** for the recovery environment.
7. Select **Plan this restore**.

Survey, approve, and execute

To approve and execute the recovery:

1. On the restore's page, review **The machine, as it described itself**.

   This is the candidate's own survey: firmware, memory, disks, free space, adapters, whether it reaches the storage machine, and whether the original still answers.
2. If you have powered off the original or changed the candidate since it described itself, select **Ask the machine again**.

3. Resolve every hard check on the **Preflight** card, accept or refuse each check that asks for your acceptance (see Full System Recovery in Place), and then select **Approve this restore**.
4. Select **Execute…**, enter the domain controller's short name in capitals, and select **Restore *NAME* now**.

Note: **Important:** Once wbadmin begins writing, the recovery cannot be stopped safely. The candidate's own Windows installation is overwritten and the candidate becomes the domain controller.

What happens during the recovery

The recovery runs as described in Full System Recovery in Place, on the candidate: its agent fetches and stages the recovery image, restarts into it, and the recovery agent checks the firmware and disks it can see against the backup before wbadmin restores every critical volume. Once the Manager has recorded success, the candidate restarts into the restored domain controller, whose agent applies the planned address and reconnects. When it first reconnects, the Manager removes the candidate's own agent record, because the operating system it described no longer exists. The run completes when validation passes.

After the restore, the candidate starts the DC agent build that staged the restore rather than the older one in the backup, and that agent checks that the domain controller works before the run completes. If the checks show **Not checked**, update the agent and select **Validate again**, or run `cdr restore validate`.

To stop, discard, or abandon the recovery, use the same actions as for an in-place recovery. Stopping works only before wbadmin starts writing.

Using the command line

```
cdr recovery candidates
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --mode server --onto <can-
didate-agent-id>
cdr restore show <run>
cdr restore survey <run>
cdr restore approve <run>
cdr restore run <run> --wait
cdr restore validate <run>
```

`cdr recovery candidates` lists each candidate's agent ID and the shape it reported. For every option, see Restore Commands and Recovery Plan Commands.

Result and next steps

The candidate now runs as the domain controller from the restore point and catches up from its partners. Next:

- Check the domain controller's address, DNS records, and site membership, especially if the restored network was left as the backup had it.
- Make sure the original machine is wiped or cannot return to the network.
- Take a fresh backup and complete the follow-ups in After a Recovery.

Full System Recovery onto a New VM

A full system recovery onto a new VM restores a whole domain controller from a full system backup onto an empty virtual machine. Nothing is installed on the VM first: no Windows and no agent. When you execute the recovery, the storage agent builds a recovery ISO for this run only, and the VM boots from it. The recovery agent on the ISO enrolls with the Manager, checks the VM's firmware and disks against the backup, and restores the domain controller's critical volumes onto the VM's disks. When it finishes, the VM is the domain controller, with the original's name, identity, and directory as of the restore point.

Use it when a domain controller will not start, or when its hardware or virtual machine is gone or must not be trusted, and you do not want to prepare a candidate machine with Windows and the DC agent first. On Hyper-V, `cdr dc rebuild` builds the VM from the restore point and boots it from the ISO in one step. The restore is non-authoritative: the domain controller catches up from its healthy partners afterward.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute. Fetching the recovery ISO to build the VM needs the `plan.execute` capability.
- You have a **Full system** restore point of the domain controller, younger than the tombstone lifetime.
- The installation ISO for the domain controller's operating system is in the OS media library. The storage agent builds the recovery ISO from its boot files. See OS Media.
- A recovery image exists for the domain controller's operating system. Check the **Recovery image** column on the **OS Media** screen.
- For the automated VM build, a Hyper-V host where you can run `cdr` elevated. On another hypervisor, build the VM by hand. Keep the Hyper-V host on current Windows updates. A host that is behind can give new virtual machines firmware that will not start an up-to-date restored system: the restore completes, and the restored system does not start.

- The VM's network reaches the Manager and the storage machine.
- You can power off the original domain controller. Two machines with one identity damage a domain rather than recover it.
- If volumes in the backup were protected by BitLocker, have the recovery keys at hand. A TPM protector belongs to the original machine.

For the full list, see Recovery Requirements. To compare this method with the others, see Restore Methods.

How the recovery media works

The recovery ISO holds the installation media's boot files, the recovery image, and the job for this run. It carries only three things that identify the run to the Manager: a single-use enrollment token, the Manager's address, and the fingerprint of the Manager's CA. The credential that reads the vault stays in the Manager's memory, and is handed to the recovery environment over its session after the recovery agent on the ISO enrolls.

- The media lives for the run's read lease, 8 hours. If no VM has booted it and enrolled by then, the run fails; nothing has been written, and the VM can be deleted.
- If the Manager restarts after the media is built and before the VM enrolls, it no longer holds the vault credential. Cancel the run and plan it again.
- A copy of the ISO holds the run's enrollment token until the run ends. Delete every local copy afterward.

Power off the original

Power off the original domain controller, or disconnect it. The new VM comes up as the same machine, with the same name and the same SID. While the original's agent is connected to the Manager, preflight shows the hard check `target.original-running`, which blocks approval, and execution is refused. If the original's hardware or virtual machine still exists, make sure it cannot be started again on the production network after the recovery.

Plan the recovery

Once you choose a new VM in **Restore destination**, the console lists every domain controller with a restore point, whether or not its own agent is connected. The original's agent is not needed.

To plan a full system recovery onto a new VM:

1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Single-DC recovery**.

2. In **Recovery mode**, select **Full system**.
3. In **Restore destination**, select **Onto a new VM booted from a recovery ISO — nothing installed on it first**.
4. In **Source domain controller**, select the original domain controller, and in **Restore point**, select a full system restore point.
5. Optionally, expand **Advanced: the restored system's network** and check **Restored system address**. The VM's adapter is a different device from the original's, so the options are the same as for a candidate; see Full System Recovery onto a Candidate.
6. If the VM's network has no DHCP, expand **Advanced: the recovery environment's network** and set **Another address** for the recovery environment.
7. Select **Plan this restore**.

Approve and execute

Nothing runs on the new VM before the recovery ISO boots, so there is no survey of it, and **Ask the machine again** is not offered. Preflight shows the advisory check `target.new-vm` instead: the recovery agent checks the VM when it boots.

To approve and execute the recovery:

1. On the restore's page, resolve every hard check on the **Preflight** card, accept or refuse each check that asks for your acceptance (see Full System Recovery in Place), and then select **Approve this restore**.
2. Select **Execute…**, enter the domain controller's short name in capitals, and select **Restore *NAME* now**.

   The storage agent builds the run's recovery ISO. When it is ready, the **Recovery media** card on the run's page shows the ISO, its SHA-256 hash, where it is exposed, and the command that builds the VM.

Build and boot the VM

On Hyper-V, build the VM with `cdr dc rebuild --restore-run`. Run it in an elevated PowerShell on the Hyper-V host that will run the VM, signed in to the Manager. Copy the command with the copy button on the **Recovery media** card, or from `cdr restore show`, and replace `<SWITCH>` with the host's virtual switch. The command names the Manager with `--manager`, so `cdr` on the host does not look for it on localhost. `cdr` refuses the command while the placeholder is unchanged:

```
cdr dc rebuild dc01.corp.example.com --restore-run <run> --switch "Recovery" --start
```

The command:

- Shapes the VM from the run's restore point: firmware generation, memory, processors, and disk sizes. The disks are blank.
- Copies the run's recovery ISO to the Hyper-V host, to `%ProgramData%\CygnaDR\RecoveryMedia` unless you give `--media-dir`, and checks its SHA-256 hash against the media's.
- Attaches the ISO with the disk first in the boot order, so the blank VM boots the ISO, and the restored system boots from its disk afterward.
- With `--start`, starts the VM. The restore then runs unattended.

`--plan` shows what would be built, and copies nothing. For the other options, see Inventory Commands.

On another hypervisor, build a VM by hand with the same firmware type as the original, at least as many disks as the backup, each the same size or larger, and enough memory; see Full System Recovery onto a Candidate for the shape. Attach the ISO shown in **Exposed at** on the **Recovery media** card, and start the VM.

Note: **Important:** The local copy of the ISO holds the run's enrollment token until the run ends. Delete it after the recovery.

What happens during the recovery

The VM boots the recovery ISO, and the recovery agent enrolls with the Manager, which then hands it the vault credential. The **Recovery media** card's **Vault credential** fact changes to **handed to the recovery environment over its session**. Before wbadmin writes anything, the recovery agent checks the VM's firmware and disks against the backup; if they do not fit, it stops with nothing written. wbadmin then restores every critical volume. Once the Manager has recorded success, the VM restarts into the restored domain controller, whose agent applies the planned address and reconnects. The run completes when validation passes, as described in Full System Recovery in Place.

The run's timeline shows the steps **Recovery ISO built**, **Waiting for a VM booted from the recovery ISO**, and **Vault credential handed to the recovery environment**. If the recovery image has never been proven by a restore and the VM does not boot it, nothing enrolls, and the run fails when its media expires; nothing will have been written, and the VM can be deleted.

Note: **Important:** Once wbadmin begins writing, the recovery cannot be stopped safely. To discard or abandon the recovery, use the same actions as for an in-place recovery. To stop it before then, power the VM off from its hypervisor: once the VM's recovery environment has connected, the Manager cannot tell it to stop, and **Ask the machine to stop…** is refused with **The recovery environment on the new VM cannot be told to stop**. When the recovery environment has been silent past its allowance, cancelling closes the run.

Using the command line

```
cdr restore plan dc01.corp.example.com --version "09/02/2026-23:00" --mode server --onto-new-vm
cdr restore approve <run>
cdr restore run <run>
cdr dc rebuild dc01.corp.example.com --restore-run <run> --switch "Recovery" --start
cdr restore show <run> --wait
```

`cdr restore run` builds the recovery ISO. `cdr restore show` then gives the media's file, size, SHA-256 hash, and location, whether the vault credential has been handed over, and the `cdr dc rebuild` command to run. Run `cdr dc rebuild` from another terminal on the Hyper-V host. For every option, see Restore Commands.

Result and next steps

The VM now runs as the domain controller from the restore point and catches up from its partners. Next:

- Delete the local copy of the recovery ISO.
- Check the domain controller's address, DNS records, and site membership, especially if the restored network was left as the backup had it.
- Make sure the original machine is wiped or cannot return to the network.
- Take a fresh backup and complete the follow-ups in After a Recovery.

To prove that backups restore this way without touching production, run a test restore on an Isolated Recovery Host. It uses the same recovery ISO and new-VM path. See Test Restores.

Recovering a Domain

A domain recovery brings one domain back from backup inside a forest that is still running. One domain controller of the domain is restored on an isolated target. The domain's other domain controllers are removed from the directory and promoted again afterward. The restored domain is fenced off from its old state: its operations master roles are seized, its RID pool is raised, and its machine account and krbtgt passwords are reset. The domain then rejoins the live forest, and the global catalogs in the other domains are reconciled with it. Use it when every domain controller of one child or tree domain is lost or compromised and the forest root is healthy. A domain recovery writes to machines outside the domain—global catalogs and trust partners—and the plan lists them before you approve.

You cannot recover the forest root domain, or the only domain of a single-domain forest, this way. Domain recovery never seizes the forest-wide roles, which live in the root; recover the forest instead (see Recovering a Forest).

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute.
- Someone with the Backup Operator role is available for the backup checkpoint near the end of the plan, unless the plan runs unattended. The Restore Operator role cannot start a backup.
- No other recovery plan is active for the same forest. A forest can have only one plan at a time.
- At least one writable domain controller of the domain has a usable full system restore point, a connected agent or a candidate machine to recover onto, and a recovery image for its operating system. If the incident is an attack, choose a restore point from before the compromise.
- Every global catalog outside the domain has a DC agent. The domain's partition is reconciled on each of them through its agent, and preflight refuses a global catalog without one.
- The topology is current. A global catalog the Manager does not know about keeps its lingering objects. Refresh it if the **Forests** screen shows the topology as stale.
- The restore target is isolated: cut off from production, with firewalled paths only to the Manager's agent port and the storage machine's SMB. Do not simply unplug the machine. See Isolating Recovery Targets.
- You have, for the domain, a Domain Admin account and password that are valid as of the restore point—usually the built-in Administrator. The plan acts as this account; the Manager holds it in memory only.
- You have chosen a new Directory Services Restore Mode (DSRM) administrator password for the recovered domain controllers, and somewhere to keep it under your password policy.
- The built-in Administrator account of every domain in the forest is enabled and its password is known. Logons that need a global catalog can fail while the catalogs are reconciled.
- For each trust with another domain of the forest that has a live domain controller with a DC agent, an administrator account and password for that partner domain. The plan resets the partner's side through that agent, as that account.
- For each trust whose partner has no DC agent, and each external trust, an administrator of the partner domain is ready to reset the partner's side.

For the full list, see Recovery Requirements. For what is not recovered, such as certificate services and DHCP on the same machines, see What Recovery Does Not Cover.

Build the plan

To build a domain recovery plan:

1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Domain recovery**.

   The **New domain recovery** page opens at step 1, **Choose**.

2. Select the **Forest**, and then the **Domain to recover**.

   The page lists the domain's domain controllers with a **Recovery action** for each. By default, the connected writable PDC emulator (or the first eligible domain controller) is set to **Restore** and every other domain controller to **Rebuild**. A domain controller under an active exclusion is marked "excluded from protection" and starts as **Exclude**; if only excluded domain controllers can be restored, one of them is still set to **Restore**. Retired domain controllers are not listed. Domain controllers of the forest with no known domain—usually a demotion that did not finish—are not in the plan, and a notice counts them.

3. Set the recovery action of each domain controller:

   - **Restore**—brought back from backup by a full system recovery. Restore exactly one domain controller; a plan that restores two cannot be approved. The others are rebuilt and replicate the whole directory from it.
   - **Rebuild**—removed from the directory now and promoted fresh once the domain is back. Its Windows roles and features are reinstalled from its last backup's record before it is promoted; their configuration and its installed programs are not. Read-only domain controllers can only be rebuilt.
   - **Exclude**—neither restored nor rebuilt. Its metadata is removed from the directory along with the domain's other domain controllers, and it is not brought back. Keep the machine off the network.

   To set several at once, select their checkboxes and then **Restore**, **Rebuild**, or **Exclude** in the bar at the bottom of the screen.

4. For the domain controller you restore, select the **Restore point**, and in **Restore destination**, select **Existing machine** or a candidate machine (**Replacement**).

   Use a candidate when the original machine will not start. See Full System Recovery onto a Candidate for how to prepare one.

5. For each domain controller you rebuild, choose in **Rebuild destination** either **Existing machine, reinstalled** or a candidate machine (**Replacement**).

   A bound candidate is renamed to the domain controller's name, given its roles and features back, and promoted. Each machine can be bound to only one domain controller.

6. Review the **Outside the domain** card.

   It lists what the plan does to the rest of the forest: **Global catalog reconciliation** of the global catalogs in other domains, one site at a time; **Trust reset** of the domain's trusts on both sides; and the **Administrator sign-in check**, which asks you to confirm that the default administrator of every domain can sign in while the global catalogs are being reconciled. A global catalog marked **No agent** blocks the plan until you install the agent on it or remove its global catalog role.

7. Under **Options**, set:

   - **Global catalog reconciliation**–**Remove lingering objects – GCs stay online** (the default, which runs in advisory mode first on every global catalog) or **Unhost, then rehost – for small forests** (every global catalog drops the domain and takes it back site by site, and cannot answer for it in between).
   - **RID pool increase**–how far the domain's available RID pool is raised. The default is 100,000.
   - **Security breach**–select this if the domain was compromised. It adds breach resets on the restored domain controller and re-creation of the domain's group managed service accounts.
   - **Re-verify each backup in full before restoring it**–optional; adds a full read of the backup.
   - **Run unattended**–optional. See Recovering a Forest. A domain recovery always resets the domain's trusts. A trust reset carried out by hand–an external trust, or a partner with no DC agent–stops for a person even in an unattended plan; the trust resets the agents carry out do not.
   - **Notifications** > **Email addresses, comma-separated**–optional. These addresses receive the plan's halts, aborts, and waits for a person, and its recovery report when it completes.
   - **New DSRM password** > **Password** and **Password again**–optional. The password the plan sets on every domain controller it recovers. Leave it empty to supply it later; the DSRM step waits for it.

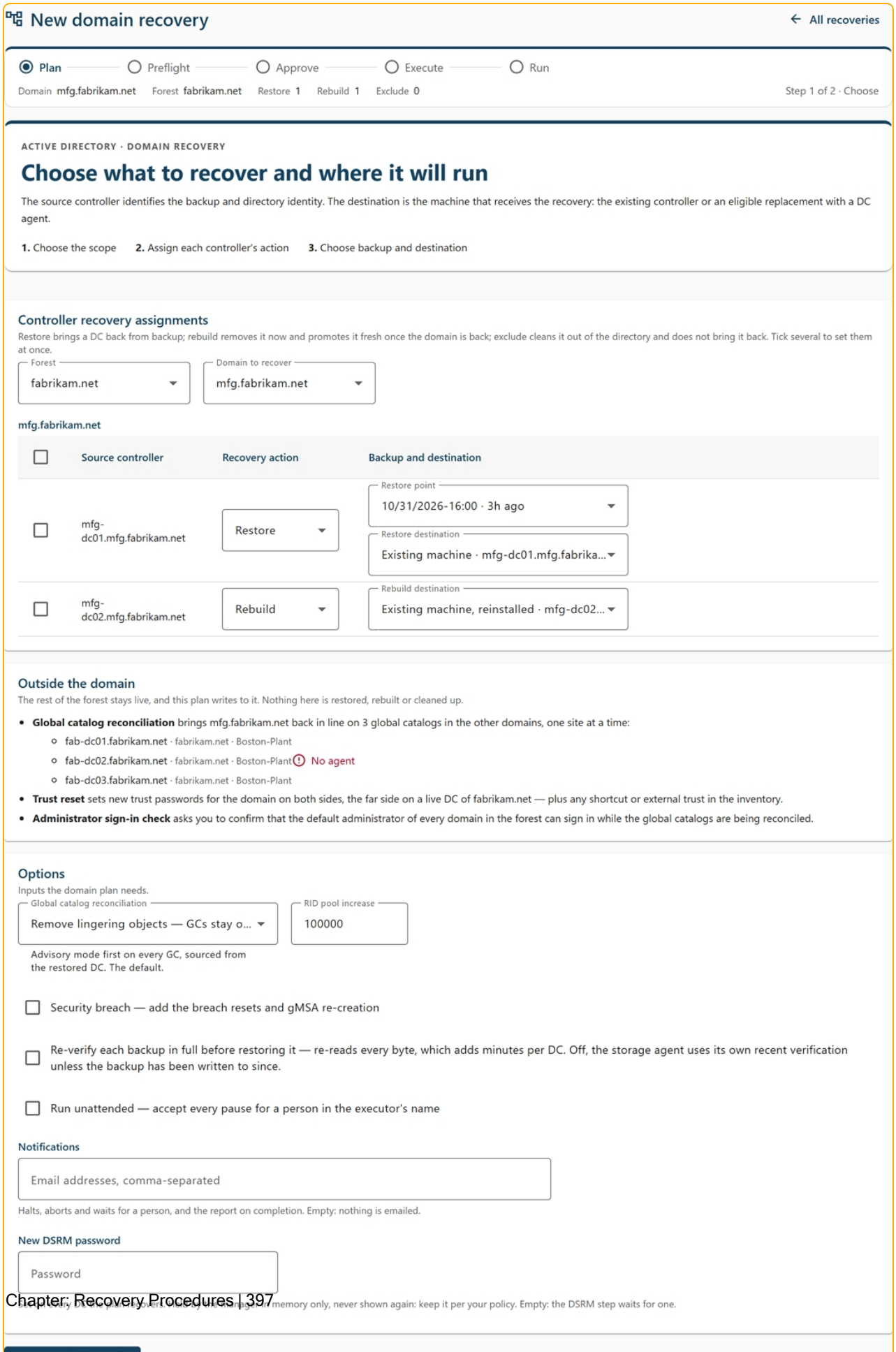
   The notification and DSRM password fields are shown only to an operator with the `plan.execute` capability.

8. Select **Review the plan**.

   If the button is unavailable, the reason is shown beside it. Step 2, **Review**, states what the plan will do in the domain and in the rest of the forest.

9. Select **Build the plan**.

   The plan's page opens with the status **Draft**. Building writes nothing to any machine.


New domain recovery
All recoveries
Plan
Preflight
Approve
Execute
Run
Domain mfg.fabrikam.net
Forest fabrikam.net
Restore 1
Rebuild 1
Exclude 0
Step 1 of 2 · Choose
ACTIVE DIRECTORY · DOMAIN RECOVERY
Choose what to recover and where it will run
The source controller identifies the backup and directory identity. The destination is the machine that receives the recovery: the existing controller or an eligible replacement with a DC agent.
1. Choose the scope
2. Assign each controller's action
3. Choose backup and destination
Controller recovery assignments
Restore brings a DC back from backup; rebuild removes it now and promotes it fresh once the domain is back; exclude cleans it out of the directory and does not bring it back. Tick several to set them at once.
Forest
fabrikam.net
Domain to recover
mfg.fabrikam.net
mfg.fabrikam.net
Source controller
Recovery action
Backup and destination
mfg-dc01.mfg.fabrikam.net
Restore
Restore point
10/31/2026-16:00 · 3h ago
Restore destination
Existing machine · mfg-dc01.mfg.fabrika...
mfg-dc02.mfg.fabrikam.net
Rebuild
Rebuild destination
Existing machine, reinstalled · mfg-dc02...
Outside the domain
The rest of the forest stays live, and this plan writes to it. Nothing here is restored, rebuilt or cleaned up.
Global catalog reconciliation brings mfg.fabrikam.net back in line on 3 global catalogs in the other domains, one site at a time:
fab-dc01.fabrikam.net · fabrikam.net · Boston-Plant
fab-dc02.fabrikam.net · fabrikam.net · Boston-Plant No agent
fab-dc03.fabrikam.net · fabrikam.net · Boston-Plant
Trust reset sets new trust passwords for the domain on both sides, the far side on a live DC of fabrikam.net — plus any shortcut or external trust in the inventory.
Administrator sign-in check asks you to confirm that the default administrator of every domain in the forest can sign in while the global catalogs are being reconciled.
Options
Inputs the domain plan needs.
Global catalog reconciliation
Remove lingering objects — GCs stay o...
RID pool increase
100000
Advisory mode first on every GC, sourced from the restored DC. The default.
Security breach — add the breach resets and gMSA re-creation
Re-verify each backup in full before restoring it — re-reads every byte, which adds minutes per DC. Off, the storage agent uses its own recent verification unless the backup has been written to since.
Run unattended — accept every pause for a person in the executor's name
Notifications
Email addresses, comma-separated
Halts, aborts and waits for a person, and the report on completion. Empty: nothing is emailed.
New DSRM password
Password
Empty: only the plan recovers. Ready: the agent in memory only, never shown again: keep it per your policy. Empty: the DSRM step waits for one.

Approve, supply credentials, and execute

To start the recovery:

1. On the plan's page, review the **Preflight findings** card and resolve every hard check. To change your choices, select **Edit this draft**.

   One advisory check is always present: the plan changes machines outside the domain being recovered. Read it. If the domain controller is restored onto a candidate machine, the checks also compare the candidate's firmware and disks with the chosen restore point. They run again when you approve, and approval is refused if one of them blocks.

2. Select **Approve this plan**.

3. On the **Domain credentials** card, select **Supply…** for each domain listed, enter the **Administrator account** (for example `CHILD\Administrator`) and **Password**, and select **Hand to the plan**.

   The card lists the recovered domain and each partner domain whose live domain controller resets its side of a trust. Each row shows **Held by the manager** or **Not held**, read from the plan itself, so the card is accurate when you open the page and after a Manager restart. A step that cannot run without a credential waits in **Waiting for a credential** until one is supplied. The Manager forgets every credential if it restarts; supply them again if the card shows **Not held**.

4. If the **New DSRM password** row shows **Not held**, select **Supply…**, enter the **Password** twice, and select **Hand to the plan**. You can also do this later; the DSRM step waits for it.

5. Select **Execute…**, enter the domain's FQDN, and select **Execute recovery**.

   For an unattended plan, the dialog is **Execute the UNATTENDED recovery of <domain>**: type the domain's name, attest the isolation of the restore target, and select **Execute unattended**.

Note: Aborting later stops the plan but does not undo steps that have already run.

Note: **Important:** An unattended domain plan attests that the restored data predates the damage, attests the administrator sign-in check, and attests the reconnect, without anyone checking. Cygna Forest Recovery reconnects nothing itself. Before you execute one, make sure every original domain controller the recovery replaces is already powered off, wiped, or isolated, and that the network is ready for the domain to rejoin the forest.

What happens during the recovery

The **Steps** card lists every step in the order the Manager runs it, with its code, title, machine, and state. The run stops for you at each decision; the **Decision required** section at the top of the page says what is needed, and the banner on other screens reads "A recovery is waiting for you". The steps, in order:

Step	What happens	Your part
DR-02	Isolate every restore target.	Attest that nothing of production can reach the targets and that their only links are the firewalled paths to the Manager's agent port and the storage machine's SMB. See Isolating Recovery Targets.
DR-03	Restore the domain controller (non-authoritative AD DS, authoritative SYSVOL). Select **Follow the restore** to watch the restore run.	Watch. If its preflight raises a check only you can accept, select **Decide on the restore** and decide on the restore run.
DR-04	Verify the restored data is from before the damage.	Sign in to the restored domain controller and spot-check the directory. Attest, or reject and re-plan with an older restore point.
DR-05	Breach resets (breach response only).	Acknowledge, then carry out by hand and attest.
DR-06 to DR-10	Seize the domain-wide operations master roles, clean up the other domain controllers' metadata (a domain controller already absent from the restored directory counts as removed), repair	Acknowledge each point of no return.

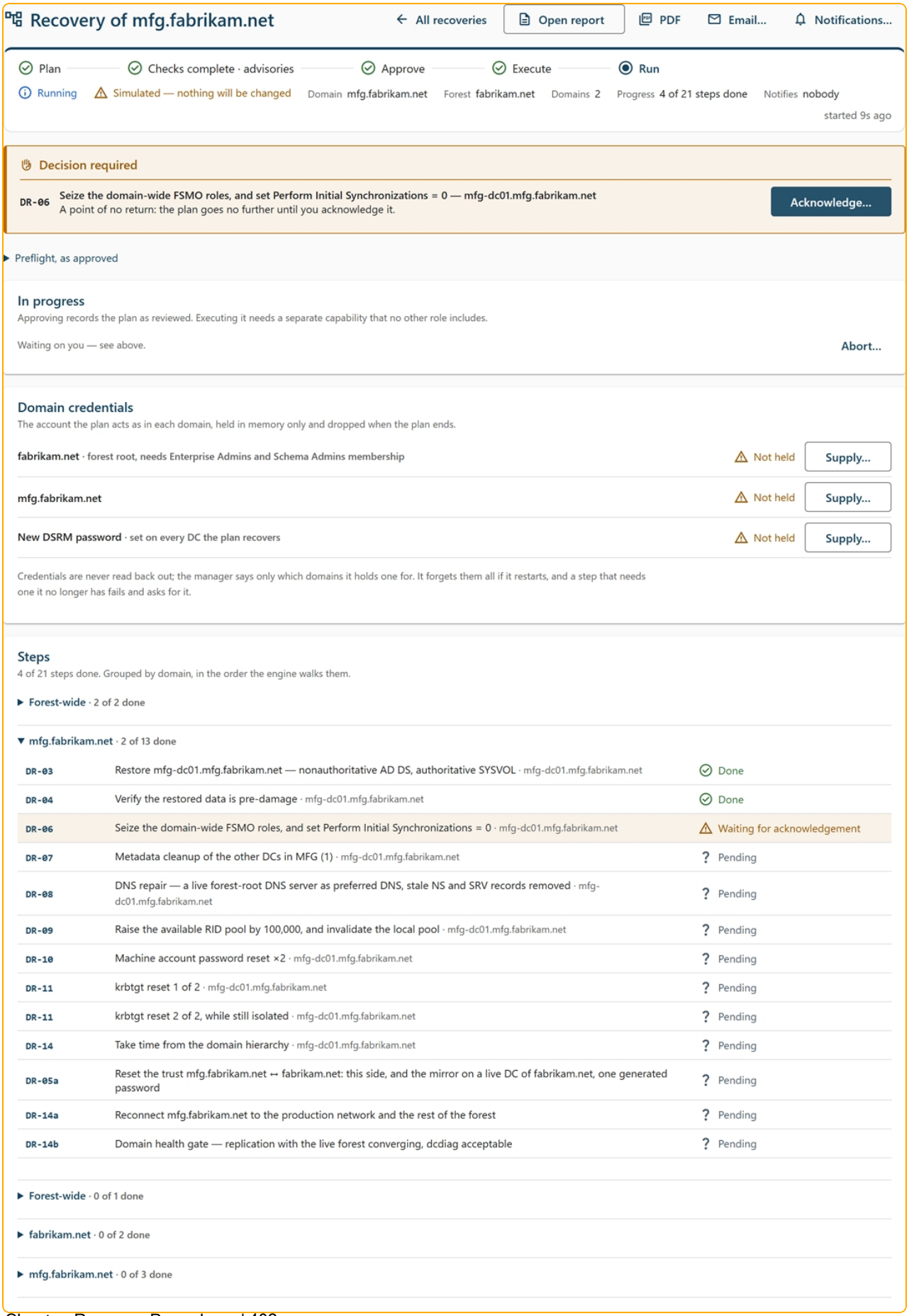
Step	What happens	Your part
	DNS, raise the RID pool, and reset the machine account password twice.	
DR-05a	Reset each of the domain's trusts on both sides with one password, breach response or not. Where the partner is a domain of the forest with a live domain controller that has a DC agent, the agents carry out two steps: **Reset <domain>'s side of the trust with <partner>** on the restored domain controller, and **Reset <partner>'s side of the trust with <domain>, on a live DC** with the partner domain's credential. Otherwise, one step is carried out by hand: this side, and the mirror on a live domain controller of the partner.	Acknowledge each agent step. For a step carried out by hand, acknowledge, carry it out, and attest; it stops for a person even in an unattended plan.
DR-11	Reset the krbtgt password twice, back to back while the domain is still isolated. In a plan created with a gap between the resets, the second reset waits it out.	Acknowledge each reset.
DR-13, DR-14	Remove the global catalog flag from the restored domain controller, and set it to take time from the domain hierarchy.	Watch.
DR-14a	Reconnect the domain to production and the rest of the forest. From here until the last rebuilt domain controller is confirmed, the domain serves on its restored domain controller alone, and the page shows **Live on reduced DCs**.	Reconnect the network, then acknowledge. This cannot be undone.
DR-14b	Domain health gate: replication with the live forest converges and dcdiag passes.	Watch.
DR-15a	Logon continuity before the global catalogs are disturbed.	Attest that every domain's built-in Administrator is enabled and its password known.
DR-15	Reconcile the domain on every global catalog outside it, one site at a time.	Watch.
DR-16	Re-create the domain's group managed service	Acknowledge.

Step	What happens	Your part
	accounts (breach response only).	
DR-17	Add the global catalog back if the restored domain controller was one, take a backup checkpoint, rebuild the other domain controllers (reinstalling the roles and features each had before promoting it), set Initial Synchronizations back to 1, set the new DSRM password you supplied on every recovered domain controller (on a newly promoted domain controller, this waits up to 15 minutes for its first RID pool), and produce the recovery report and email it to the plan's notification addresses.	Carry out the backup checkpoint by hand and attest; an unattended plan passes over it and the report says so. Make sure each rebuild has its machine. Supply the new DSRM password if the plan does not hold one. Read the report.

How to respond to each kind of decision is the same as in a forest recovery. See Recovering a Forest. In short:

- **Waiting for acknowledgement**–select **Acknowledge…**, type the step code (for example `DR-06`), and select **Acknowledge DR-06**.
- **Waiting for you**–do what the prompt asks, and then select **Attest**, or select **Reject…** with a reason to halt the plan.
- **Waiting for a credential**–select **Supply credentials…** or **Supply DSRM password…**, enter it, and select **Hand to the plan**. The step runs by itself once it is supplied.
- **Waiting (krbtgt)**–only in a plan created with a gap between the krbtgt resets. The second reset runs by itself at the time shown. To run it early, select **Run it now…**, give a reason, and select **Override the wait**.
- **Waiting for its machine to be built**–if no candidate is bound to the rebuild, install Windows Server cleanly on the domain controller's own machine, with its name, and install the DC agent. The step runs once the agent connects.

If the plan has notification addresses, each wait is also emailed to them once.


Recovery of mfg.fabrikam.net
All recoveries
Open report
PDF
Email...
Notifications...
Plan
Checks complete · advisories
Approve
Execute
Run
Running
Simulated — nothing will be changed
Domain mfg.fabrikam.net
Forest fabrikam.net
Domains 2
Progress 4 of 21 steps done
Notifies nobody
started 9s ago
Decision required
DR-06
Seize the domain-wide FSMO roles, and set Perform Initial Synchronizations = 0 — mfg-dc01.mfg.fabrikam.net
A point of no return: the plan goes no further until you acknowledge it.
Acknowledge...
Preflight, as approved
In progress
Approving records the plan as reviewed. Executing it needs a separate capability that no other role includes.
Waiting on you — see above.
Abort...
Domain credentials
The account the plan acts as in each domain, held in memory only and dropped when the plan ends.
fabrikam.net · forest root, needs Enterprise Admins and Schema Admins membership
Not held
Supply...
mfg.fabrikam.net
Not held
Supply...
New DSRM password · set on every DC the plan recovers
Not held
Supply...
Credentials are never read back out; the manager says only which domains it holds one for. It forgets them all if it restarts, and a step that needs one it no longer has fails and asks for it.
Steps
4 of 21 steps done. Grouped by domain, in the order the engine walks them.
Forest-wide · 2 of 2 done
mfg.fabrikam.net · 2 of 13 done
DR-03 Restore mfg-dc01.mfg.fabrikam.net — nonauthoritative AD DS, authoritative SYSVOL · mfg-dc01.mfg.fabrikam.net Done
DR-04 Verify the restored data is pre-damage · mfg-dc01.mfg.fabrikam.net Done
DR-06 Seize the domain-wide FSMO roles, and set Perform Initial Synchronizations = 0 · mfg-dc01.mfg.fabrikam.net Waiting for acknowledgement
DR-07 Metadata cleanup of the other DCs in MFG (1) · mfg-dc01.mfg.fabrikam.net Pending
DR-08 DNS repair — a live forest-root DNS server as preferred DNS, stale NS and SRV records removed · mfg-dc01.mfg.fabrikam.net Pending
DR-09 Raise the available RID pool by 100,000, and invalidate the local pool · mfg-dc01.mfg.fabrikam.net Pending
DR-10 Machine account password reset ×2 · mfg-dc01.mfg.fabrikam.net Pending
DR-11 krbtgt reset 1 of 2 · mfg-dc01.mfg.fabrikam.net Pending
DR-11 krbtgt reset 2 of 2, while still isolated · mfg-dc01.mfg.fabrikam.net Pending
DR-14 Take time from the domain hierarchy · mfg-dc01.mfg.fabrikam.net Pending
DR-05a Reset the trust mfg.fabrikam.net ↔ fabrikam.net: this side, and the mirror on a live DC of fabrikam.net, one generated password Pending
DR-14a Reconnect mfg.fabrikam.net to the production network and the rest of the forest Pending
DR-14b Domain health gate — replication with the live forest converging, dcdiag acceptable Pending
Forest-wide · 0 of 1 done
fabrikam.net · 0 of 2 done
mfg.fabrikam.net · 0 of 3 done

Halts, retries, and aborting

If a step fails, the plan's status changes to **Halted — a step failed**, and the page names the step, the machine, and the error. Nothing more is issued until the step is retried or the plan is aborted. Fix what failed it, select **Retry <step>…** in the halted banner, and select **Retry the step**: the step runs again from its checkpoint and the plan carries on.

You can abort at any time before the plan finishes: select **Abort…**, enter a reason, and select **Abort recovery**. The dialog says what aborting now leaves. Steps that have already run are not undone. Each restore the plan started is asked to stop; one that is already writing carries on, and if its machine is gone, close it as abandoned from its own page.

Using the command line

```
cdr recovery plan corp.example.com --domain child.corp.example.com --restore "dc11.child.-
corp.example.com=09/02/2026-23:00" --rebuild dc12.child.corp.example.com --gc-strategy linger-
ing-objects
cdr recovery show <plan>
cdr recovery approve <plan>
cdr recovery secrets <plan> --domain child.corp.example.com --admin "CHILD\Administrator"
cdr recovery secrets <plan> --dsrm
cdr recovery execute <plan>
cdr recovery ack <plan> DR-06@child.corp.example.com --confirm DR-06
cdr recovery attest <plan> <step>
cdr recovery retry <plan> <step>
cdr recovery report <plan>
cdr recovery abort <plan> --reason "Re-planning with an older restore point"
```

`--restore` takes `<fqdn>=<version>`, with an optional `:rollback` or `:server` mode (full system recovery by default); `--onto` binds a candidate to a restored or rebuilt domain controller; `--breach` sets the breach response; `--unattended`, `--notify`, and `--dsrm` match the console's options. `cdr recovery show` lists each step ID and what a waiting step needs. For every option, see Recovery Plan Commands.

Result and next steps

When every step is done, the plan's status is **Completed**. The domain runs from the restored domain controller and its rebuilt partners, connected to the live forest. Every domain controller the plan restored or rebuilt that has an enabled backup policy is due for a backup at once; each backup starts when the domain controller's agent connects. Next, read the recovery report, complete its follow-ups, and check that those backups completed. The plan page's **Open report**, **PDF**, and **Email…** actions give you the report at any time; see After a Recovery.

Recovering a Forest

A forest recovery brings every domain of a forest back from backup, following the Microsoft Active Directory forest recovery procedure. In each domain, one domain controller is restored on isolated targets—the forest root first, then each child domain after its parent. Each restored domain is fenced off from its pre-recovery state, the restored domain controllers are joined and checked, and only then is the forest reconnected to production. The remaining domain controllers are promoted fresh afterward. Use it when the forest root domain is lost, when the whole forest is compromised, or when a single-domain forest's domain is lost. By default the plan stops for you at every point of no return and at every check only a person can make. A plan built to run unattended accepts those pauses in the name of the operator who executes it.

The steps are built on Microsoft's Active Directory Forest Recovery Guide. For how each one maps to the guide, and why the `krbtgt` resets run back to back by default, see Alignment with Microsoft Guidance.

Before you begin

- You need the Restore Operator role, or one operator who can approve and another who can execute. Plan for several people over several hours, depending on the size of the forest and the number of domain controllers to rebuild.
- Someone with the Backup Operator role is available for the backup checkpoint step, unless the plan runs unattended. The Restore Operator role cannot start a backup.
- No other recovery plan is active for the forest.
- In every domain, at least one writable domain controller has a usable full system restore point and a recovery image for its operating system, and either its own connected agent or a prepared candidate machine. Choose restore points from one backup window across the forest, and from before the compromise if the incident is an attack.
- The topology knows the forest root domain. Refresh it if the **Forests** screen shows **Root domain unidentified** or a stale topology.
- The restore targets are isolated: cut off from production, with firewalled paths only to the Manager's agent port and the storage machine's SMB. Do not simply unplug the machines. See Isolating Recovery Targets.
- You have at least one external time source (NTP server) that the root PDC emulator can reach from the isolated network.
- For each domain, you have an administrator account and password valid as of the restore point—usually the built-in Administrator. The forest root needs an Enterprise Admin and Schema Admin; other domains need a Domain Admin. The Manager holds these in memory only.

- You have chosen a new Directory Services Restore Mode (DSRM) administrator password for the recovered domain controllers, and somewhere to keep it under your password policy.
- If you plan the recovery as a breach response, an administrator of each external trust's partner forest is ready to reset the partner's side.

For the full list, see Recovery Requirements. For what is not recovered, see What Recovery Does Not Cover.

Build the plan

To build a forest recovery plan:

1. Go to **Recovery** > **Recovery Plans**, and under **Active Directory recovery**, select **Forest recovery**.

   The **New forest recovery** page opens at step 1, **Choose**.

2. Select the **Forest**.

   The page shows one table per domain, the root first. In each domain, the connected writable PDC emulator (or the first eligible domain controller) is set to **Restore** and every other domain controller to **Rebuild**. A domain controller under an active exclusion is marked "excluded from protection" and starts as **Exclude**; if only excluded domain controllers in a domain can be restored, one of them is still set to **Restore**. Retired domain controllers are not listed. Domain controllers with no known domain—usually a demotion that did not finish—are not in the plan, and a notice above the tables counts them.

3. Set the **Recovery action** of each domain controller:

   - **Restore**—brought back from backup by a full system recovery. Every domain needs exactly one. A plan that restores two domain controllers in one domain cannot be approved: restore one and rebuild the others, which replicate the whole directory from it.
   - **Rebuild**—removed from the directory now and promoted fresh once its domain is back. Its Windows roles and features are reinstalled from its last backup's record before it is promoted; their configuration and its installed programs are not. Read-only domain controllers can only be rebuilt.
   - **Exclude**—neither restored nor rebuilt. Its metadata is removed from the directory along with the domain's other domain controllers, and it is not brought back. Keep the machine off the network.

   To set several at once, select their checkboxes and use the bar at the bottom of the screen.

4. For each domain controller you restore, select the **Restore point** and, in **Restore destination**, either **Existing machine** or a candidate machine (**Replacement**).

   See Full System Recovery onto a Candidate for how to prepare a candidate. Each machine can be bound to only one domain controller.

5. For each domain controller you rebuild, choose in **Rebuild destination** either **Existing machine, reinstalled** or a candidate machine (**Replacement**).

   A bound candidate is renamed to the domain controller's name, given its roles and features back, and promoted.

6. Under **Options**, set:

   - **External time sources (required)**—one or more NTP servers, comma-separated, reachable from the isolated network. The root PDC emulator takes its time from these.
   - **RID pool increase**—how far each domain's available RID pool is raised. The default, 100,000, is Microsoft's figure.
   - **Security breach**—select this when the forest was compromised. It adds the breach resets on every restored domain controller and resets every trust. Without it, no trust is reset: each side of every trust keeps the trust password its own restore point holds.
   - **Re-verify each backup in full before restoring it**—optional; adds minutes per domain controller.
   - **Run unattended**—optional. See Recovering a Forest.
   - **Notifications** > **Email addresses, comma-separated**—optional. These addresses receive the plan's halts, aborts, and waits for a person, and its recovery report when it completes. Leave it empty to email nobody.
   - **New DSRM password** > **Password** and **Password again**—optional. The password the plan sets on every domain controller it recovers. **Password again** appears once you fill in **Password**. Leave it empty to supply it later; the DSRM step waits for it.

   The notification and DSRM password fields are shown only to an operator with the `plan.execute` capability, because the addresses receive the plan's report outside the Manager.

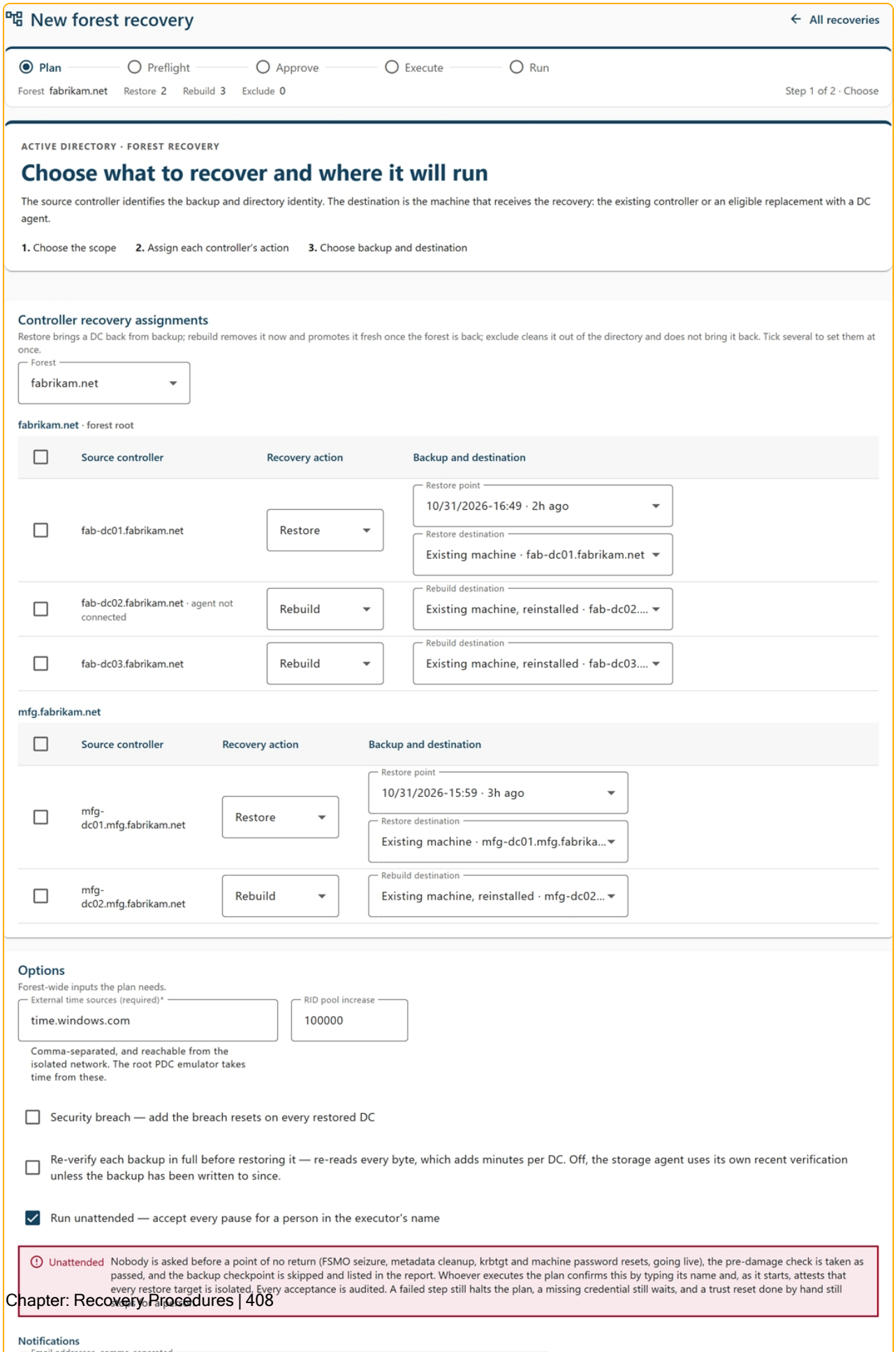
7. Select **Review the plan**.

   Step 2, **Review**, states what is restored, rebuilt, and left out in each domain, the forest-wide choices, whether the plan runs unattended, who it emails, and whether it holds a DSRM password.

8. Select **Build the plan**.

The plan's page opens with the status **Draft**. Building writes nothing to any machine.

The console restores every domain controller by full system recovery and adds the global catalog back on the first restored root domain controller. To restore a domain controller by DC state recovery, or to choose which restored root domain controller becomes the first global catalog, build the plan from the command line.


New forest recovery
All recoveries
Plan
Preflight
Approve
Execute
Run
Forest fabrikam.net
Restore 2
Rebuild 3
Exclude 0
Step 1 of 2 · Choose
ACTIVE DIRECTORY · FOREST RECOVERY
Choose what to recover and where it will run
The source controller identifies the backup and directory identity. The destination is the machine that receives the recovery: the existing controller or an eligible replacement with a DC agent.
1. Choose the scope
2. Assign each controller's action
3. Choose backup and destination
Controller recovery assignments
Restore brings a DC back from backup; rebuild removes it now and promotes it fresh once the forest is back; exclude cleans it out of the directory and does not bring it back. Tick several to set them at once.
Forest
fabrikam.net
fabrikam.net · forest root
Source controller
Recovery action
Backup and destination
fab-dc01.fabrikam.net
Restore
Restore point
10/31/2026-16:49 · 2h ago
Restore destination
Existing machine · fab-dc01.fabrikam.net
fab-dc02.fabrikam.net · agent not connected
Rebuild
Rebuild destination
Existing machine, reinstalled · fab-dc02....
fab-dc03.fabrikam.net
Rebuild
Rebuild destination
Existing machine, reinstalled · fab-dc03....
mfg.fabrikam.net
Source controller
Recovery action
Backup and destination
mfg-dc01.mfg.fabrikam.net
Restore
Restore point
10/31/2026-15:59 · 3h ago
Restore destination
Existing machine · mfg-dc01.mfg.fabrika...
mfg-dc02.mfg.fabrikam.net
Rebuild
Rebuild destination
Existing machine, reinstalled · mfg-dc02...
Options
Forest-wide inputs the plan needs.
External time sources (required)*
time.windows.com
Comma-separated, and reachable from the isolated network. The root PDC emulator takes time from these.
RID pool increase
100000
Security breach — add the breach resets on every restored DC
Re-verify each backup in full before restoring it — re-reads every byte, which adds minutes per DC. Off, the storage agent uses its own recent verification unless the backup has been written to since.
Run unattended — accept every pause for a person in the executor's name
Unattended
Nobody is asked before a point of no return (FSMO seizure, metadata cleanup, krbtgt and machine password resets, going live), the pre-damage check is taken as passed, and the backup checkpoint is skipped and listed in the report. Whoever executes the plan confirms this by typing its name and, as it starts, attests that every restore target is isolated. Every acceptance is audited. A failed step still halts the plan, a missing credential still waits, and a trust reset done by hand still
Notifications

Approve, supply credentials, and execute

To start the recovery:

1. On the plan's page, review the **Preflight findings** card and resolve every hard check. To change your choices, select **Edit this draft**.

   For each domain controller restored onto a candidate machine, the checks compare the candidate's firmware and disks with the chosen restore point. They run again when you approve, against the machines as they are then, and approval is refused if one of them blocks. Read the advisory checks too. In particular, look for a warning that the chosen restore points are not from one backup window and, in a plan that is not a breach response, a warning that the two sides of a trust are restored from points 30 days or more apart, so their trust passwords may no longer match. For each check, see Preflight Check Reference.

2. Review the **Plan as built** card. It shows what happens to each domain controller, at which step, and where; what each rebuild reinstalls and what it does not carry; the time source; the RID pool increase; the gap between krbtgt resets (**None: both resets run while isolated**, unless the plan was created with a gap); and the first global catalog. It does not change once built.

3. Select **Approve this plan**.

4. On the **Domain credentials** card, select **Supply…** for each domain, enter the **Administrator account** (for example `CORP\Administrator`) and **Password**, and select **Hand to the plan**.

   The root row reads "forest root, needs Enterprise Admins and Schema Admins membership". Each row shows **Held by the manager** or **Not held**, read from the plan itself, so the card is accurate when you open the page and after a Manager restart. A step that cannot run without a domain's credential waits in **Waiting for a credential** until one is supplied. The Manager forgets every credential if it restarts; supply them again when the card shows **Not held**.

5. If the **New DSRM password** row shows **Not held**, select **Supply…**, enter the **Password** twice, and select **Hand to the plan**.

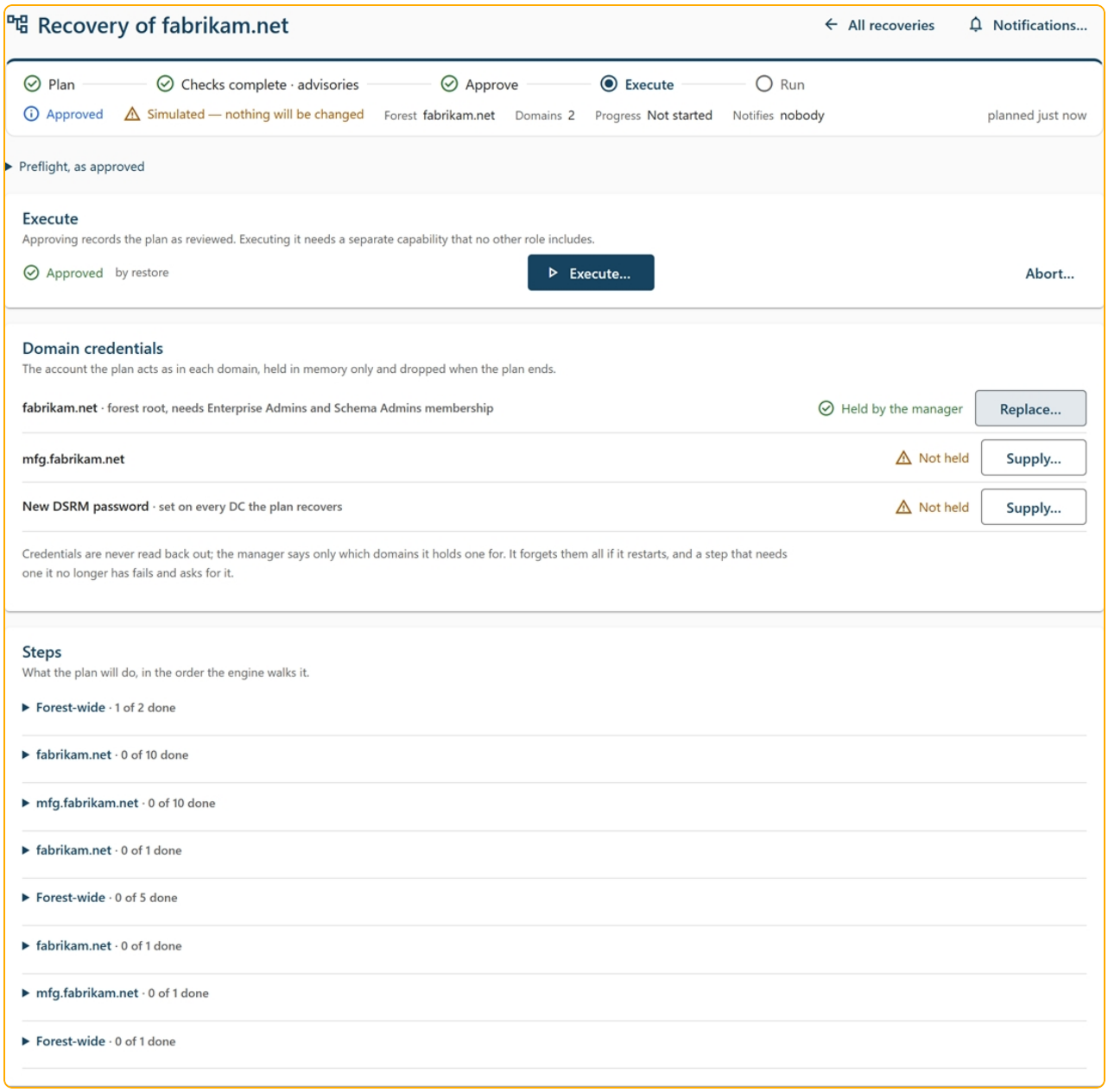
   The row then shows **Held by the manager**. You can do this later; the DSRM step waits for it.

6. Select **Execute…**.

   The confirmation lists the number of steps and the machines they touch. For an unattended plan, the dialog is different; see Recovering a Forest.

7. Enter the forest's FQDN and select **Execute recovery**.

Note: Aborting later stops the plan but does not undo steps that have already run.



Running a plan unattended

A recovery can run for hours. A plan built with **Run unattended** selected does not stop for a person at its pauses: the person who executes it accepts them all, in advance, in their own name.

Executing an unattended plan opens the dialog **Execute the UNATTENDED recovery of <forest>**. It lists what will be accepted without asking. To start the plan, type the forest's name, select the attestation that every restore target is cut off from production with only the firewalled paths to the Manager and storage machine, and select **Execute unattended**. The isolation attestation is taken at execution, not approval, so that it is current.

Once it runs, an unattended plan:

- Acknowledges every point of no return.
- Attests every operator step: isolation, the pre-damage check, and the reconnect.
- Accepts the restore preflight confirmations of each restore.
- Passes over the backup checkpoint without carrying it out. The recovery report lists it as a follow-up.

Each acceptance is recorded in the audit trail and listed in the recovery report. The plan still stops when:

- A step fails. The plan halts.
- A step needs a domain credential or the new DSRM password the Manager does not hold. It waits until one is supplied.
- The plan was created with a gap between the krbtgt resets. The gap is waited out.
- It reaches a trust reset that is carried out by hand: this forest's side of an external trust, or a trust inside the forest with a side that has no restored domain controller. These exist only in a breach response, and they stop for a person because the old trust passwords still work until the reset is done. The trust resets the agents carry out are accepted like any other point of no return.

Note: **Important:** An unattended plan also attests that the restored data predates the damage, and attests the reconnect, without anyone checking. Cygna Forest Recovery reconnects nothing itself. Before you execute an unattended plan, make sure every original domain controller the recovery replaces is already powered off, wiped, or isolated, and that the network is ready for the recovered forest to go live and for the rebuilt domain controllers to reach the restored ones.

To stop accepting pauses on a running plan, select **Take back**, then **Take it back**. From then on every pause waits for a person. A plan that has been taken back cannot be made unattended again. While a plan runs unattended, its page shows an **Unattended** chip.

What happens during the recovery

The **Steps** card lists every step, grouped by domain, in the order the Manager runs it. Each row shows the step code, title, machine, and state, the steps a pending step waits for, and the reason for any check that failed. The summary shows how many steps are done. The steps:

Step	What happens	Your part
FR-02	Isolate every restore target from production.	Attest that no domain controller, member, or client of the directory, and no other production machine, can reach the targets, and that their only links are the firewalled paths to the Manager's agent port and the storage machine's SMB. See Isolating Recovery Targets.
FR-03	Restore the domain's domain controller: non-authoritative AD DS, authoritative SYSVOL. Select **Follow the restore** to open its restore run.	Watch the restore run. If its preflight raises a check only you can accept, decide on the restore run.
FR-04	Verify the restored data is from before the damage.	Sign in to the restored domain controller and spot-check the directory. Attest, or reject and re-plan with an older restore point.
FR-05	Breach resets on the restored domain controller (breach response only).	Acknowledge, carry out by hand, and attest.
FR-06	Seize the operations master roles—all five in the root, the domain-wide roles in a child—and set Perform Initial Synchronizations to 0.	Acknowledge.
FR-07	Remove the metadata of every other domain controller in the domain. A domain controller already absent from the restored directory counts as removed.	Acknowledge.

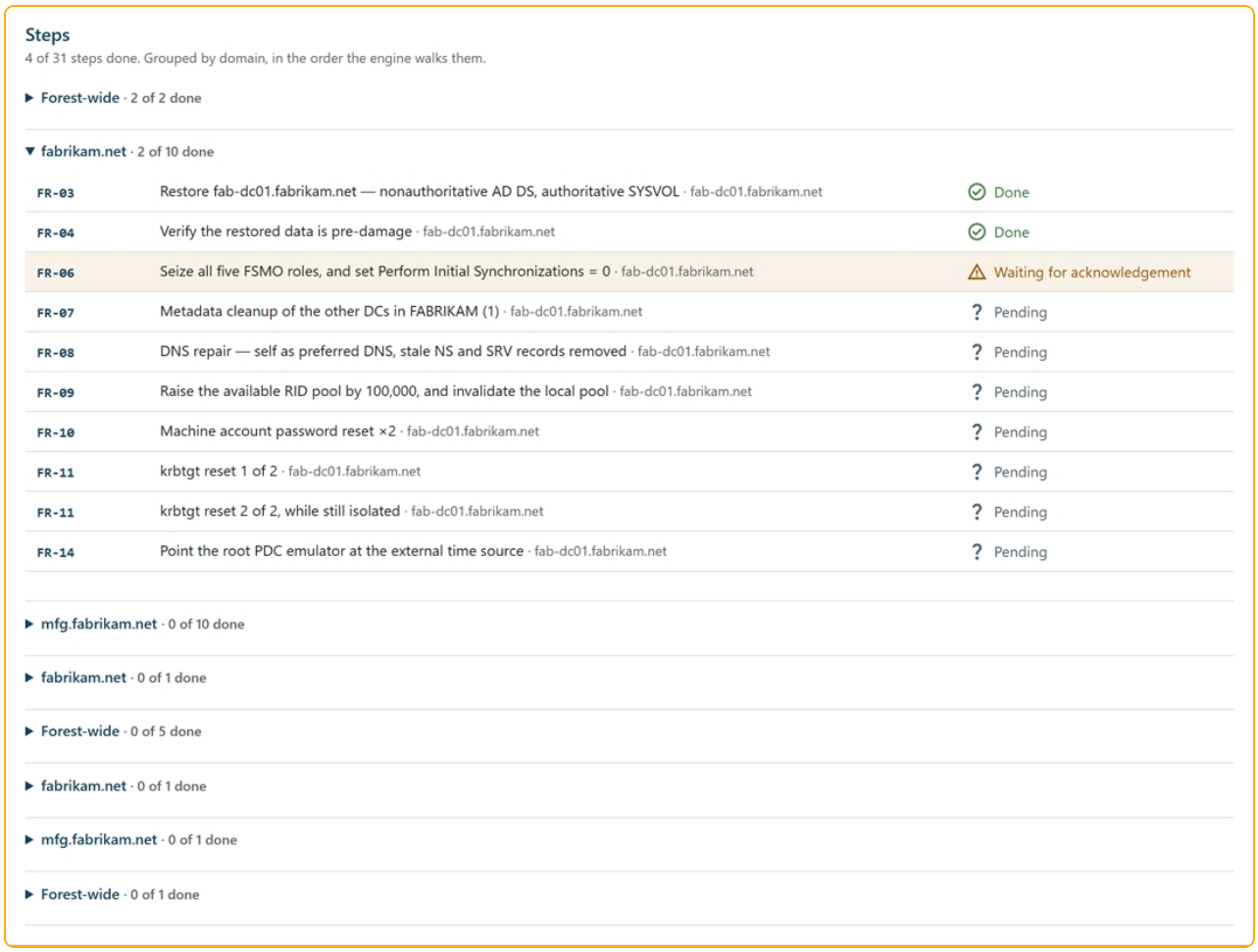
Step	What happens	Your part
FR-08	Repair DNS: set the preferred DNS server and remove stale name server and service records. In a child domain, the preferred DNS server is the forest root's restored domain controller.	Watch.
FR-09	Raise the available RID pool and invalidate the local pool.	Acknowledge.
FR-10	Reset the domain controller's machine account password twice.	Acknowledge.
FR-11	Reset the krbtgt password twice, back to back while the forest is still isolated. In a plan created with a gap between the resets, the second reset waits it out.	Acknowledge each reset.
FR-13, FR-14	Remove the global catalog flag from every domain's restored domain controller (forests with more than one domain), and set the time source: the external source on the root PDC emulator, the domain hierarchy elsewhere.	Watch.
FR-12	Breach response only: reset every trust. For a trust inside the forest, there is one step per side, run by the agent on that domain's restored domain controller with that domain's credential; both sides are reset to one password the Manager generates and holds in memory. For an external trust, only this forest's side is reset. Without the breach flag, this step is not in the plan and no trust is reset.	Acknowledge each side. For an external trust, carry out this forest's side by hand and attest; the partner forest's administrator resets their side. That step stops for a person even in an unattended plan.
FR-16	Join the restored domain controllers to one isolated network and fix name resolution.	Watch.
FR-17	Forest health gate, one step per restored domain controller, run by its own agent: ask its KCC to drop links to removed domain controllers, pull replication into it from its partners (retrying up to four times, 30 seconds apart, while another	Watch. Nothing is promoted before every restored domain controller's gate passes. If one fails, resolve what its

Step	What happens	Your part
	restored domain controller cannot be pulled from yet), check that its inbound replication converges and that its domain lists exactly its restored domain controller, and run dcdiag. In a forest with more than one domain, dcdiag reporting no global catalog is expected at this point and does not block, nor do the event-log tests.	checks name and retry it.
FR-18	Add the global catalog back on the designated root domain controller, and wait until it serves as a global catalog: the step pulls every partition from its partners every 2 minutes until the global catalog completes its first full sync. It fails after 30 minutes.	Watch. If it fails, check with `repadmin /showrepl` on the domain controller for a partition that is not replicating, then retry the step.
FR-19	Backup checkpoint: a fresh backup of every restored domain controller before redeployment begins.	Run **Back up now** on each restored domain controller, wait for each job to complete, and attest. An unattended plan passes over this step, and the recovery report says so.
FR-20	Reconnect to the production network. This is the moment the recovered forest goes live. From here until the last rebuilt domain controller is confirmed, each domain serves on its restored domain controllers alone, and the page shows **Live on reduced DCs**.	Acknowledge, reconnect, and attest. Confirm that none of the original machines the recovery replaced can rejoin.

Step	What happens	Your part
FR-21	Rebuild each remaining domain controller: rename the bound candidate, if any, and restart it; point its DNS at the recovered domain; reinstall the roles and features it had, as its last backup recorded them; promote it into its site; and confirm it came back. A role or feature that will not install is named, and the promotion goes ahead.	Make sure each rebuild has its machine: the bound candidate, or the domain controller's own machine reinstalled with the DC agent.
FR-22	Clean up: set Initial Synchronizations back to 1 on each restored domain controller, set the new DSRM administrator password you supplied on every recovered domain controller (on a newly promoted domain controller, this waits up to 15 minutes for its first RID pool), and produce the recovery report and email it to the plan's notification addresses.	Supply the new DSRM password if the plan does not hold one. Then read the report and complete its follow-ups. See After a Recovery.

Steps FR-03 to FR-14 run for the root first, and for each child domain after its parent. The restore at FR-03 runs as an ordinary restore run, so its page shows the same phases as Full System Recovery in Place.

The recovery leaves the operations master roles on the restored domain controllers. You move them to their final placement; the recovery report lists this follow-up.



Responding when the plan waits for you

When a step needs you, the **Decision required** section at the top of the plan's page explains what is needed, and the banner on every other screen reads "A recovery is waiting for you". Responding needs execution rights. If the plan has notification addresses, each wait is also emailed to them once.

Acknowledge a point of no return

A step marked **Point of no return** cannot be walked back once it runs, so the plan goes no further until you acknowledge it. Its state is **Waiting for acknowledgement**.

1. Read the step's title, machine, and detail.
2. Select **Acknowledge…**.
3. Type the step code shown in the dialog, for example `FR-06`, and select **Acknowledge FR-06**.

Note: **Important:** Acknowledge FR-20 only when you are ready for the recovered forest to go live. From then on it cannot be isolated again without a new recovery, and none of the original domain controllers it replaced may ever return to the network: each must be powered off, wiped, or reinstalled clean, because it still holds the forest as it was before the recovery.

Attest or reject an operator step

Some steps are checks only a person can make, such as isolation (FR-02), the pre-damage check (FR-04), and reconnecting (FR-20). Others—breach resets, the reset of this forest's side of an external trust, and the backup checkpoint—are carried out by you; the step's detail says what to do and, after "It is done when:", when the step is done. Their state is **Waiting for you**.

- When the check passes or you have carried out the step, select **Attest**. There is no further confirmation.
- If it does not pass—for example the restored directory still shows the damage—select **Reject…**, enter why, and select **Reject and halt**. The plan halts.

Decide on a restore's preflight

If the preflight of a restore at FR-03 raises a check only a person can accept—for example, the target cannot resolve the storage machine's name from where it is now—the step waits for you and offers **Decide on the restore** instead of **Attest**. Select it to open the restore run, then either approve the run accepting each such check, or select **Cancel this restore…**, which fails the step and halts the plan. See Full System Recovery in Place.

Supply a credential or the DSRM password

A step in the state **Waiting for a credential** needs something the Manager does not hold: a domain's administrator credential, or the new DSRM password. Select **Supply credentials…** or **Supply DSRM password…** in the **Decision required** section, enter it, and select **Hand to the plan**. The step runs by itself once it is supplied. The Manager holds both in memory only, so after a Manager restart a step may ask again.

A gap between the krbtgt resets

By default the second krbtgt reset follows the first as soon as it is acknowledged: both run while the forest is isolated, and going live waits for the second, so no client holds a ticket issued between them. In a plan created with a gap between the resets, the second reset waits it out. Its state is **Waiting (krbtgt)**, and the page shows when it runs by itself. To run it early:

1. Select **Run it now…**.
2. Enter a **Reason**. It is recorded in the audit trail and the recovery report.
3. Select **Override the wait**.

   Kerberos tickets issued between the two resets are invalidated.

A rebuild waiting for its machine

A rebuild step in the state **Waiting for its machine to be built** needs a machine to promote that has never connected. If the plan bound no candidate to the domain controller, install Windows Server cleanly on the domain controller's own machine, with its name, and install the DC agent on it. The step runs by itself once the agent connects. While a rebuilt machine restarts between steps, its step shows **Restarting** and needs nothing from you.

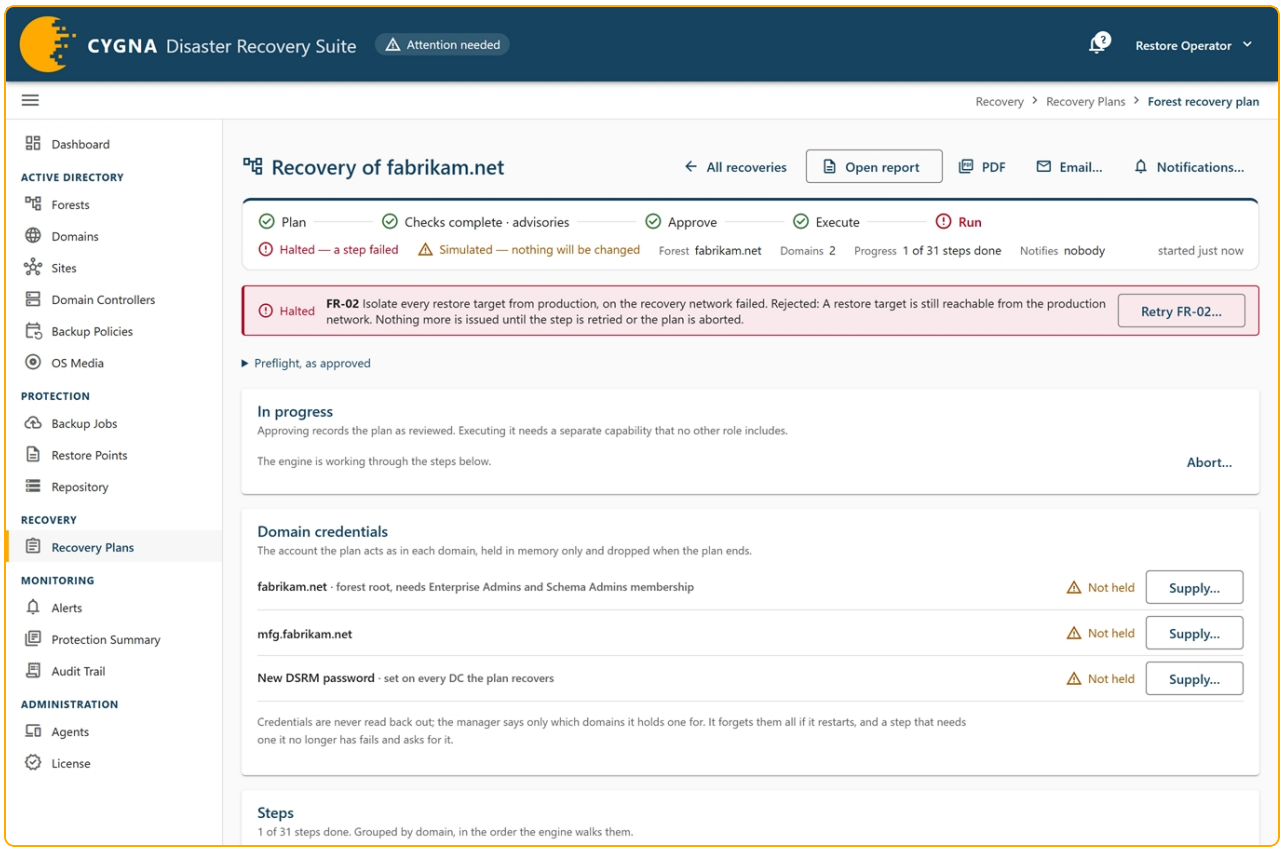
Halts, retries, and aborting

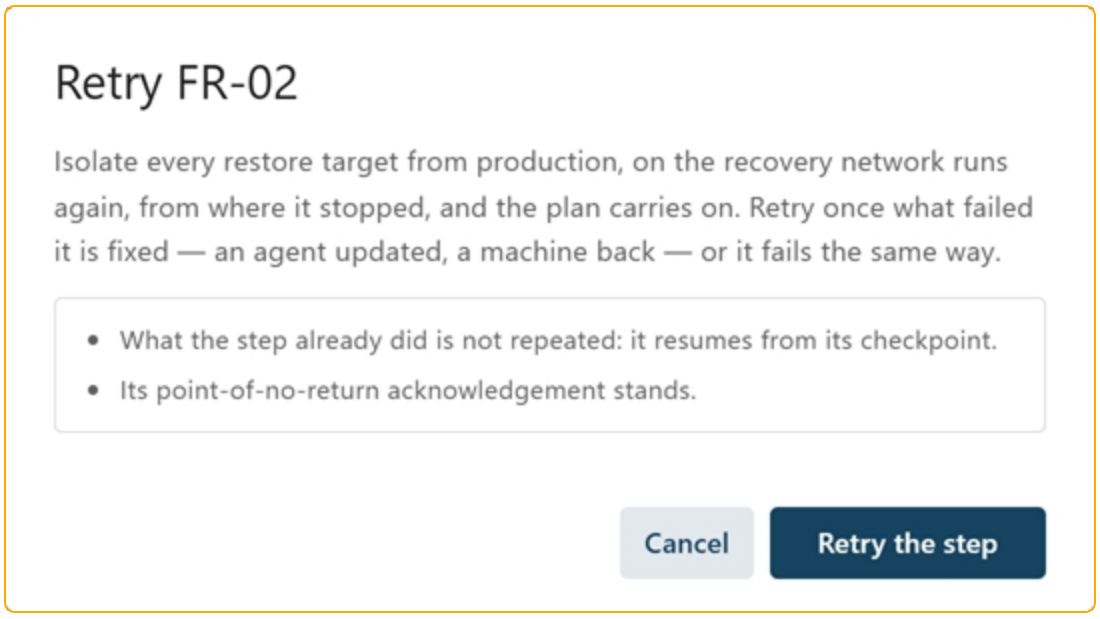
If a step fails, the plan's status changes to **Halted — a step failed**, and the page names the step, the machine, and the error, with each check that did not pass. Nothing more is issued until the step is retried or the plan is aborted. A step that was already running on a machine when the plan halted may still finish and report.

To retry the step that halted the plan:

1. Fix what failed it—for example, update an out-of-date agent or bring a machine back.
2. In the halted banner, select **Retry <step>…**.
3. Select **Retry the step**.

   The step runs again from its checkpoint, without repeating what it already did, and the plan carries on. Its point-of-no-return acknowledgment stands.





To abort a recovery:

1. Select **Abort...**.

   The dialog says what aborting now leaves: which points of no return are behind the plan, and which changes stay to finish or undo by hand.

2. In **Why are you abandoning this recovery?**, enter the reason.
3. Select **Abort recovery**.

   The plan stops where it is and its status is **Aborted**. Steps that have already run are not undone. Each restore the plan started is asked to stop; one that is already writing carries on, and if its machine is gone, close it as abandoned from its own page.

The recovery report and notifications

The last step produces the recovery report: what happened to each domain controller, the manual follow-ups the recovery left, the decisions and interventions on the way, and every step. It is emailed to the plan's notification addresses. Once the plan has started, you can read it at any time with **Open report**, save it with **PDF**, or send it with **Email…**. For what it contains, see After a Recovery.

To change who the plan emails, select **Notifications…**, edit **Addresses**, and select **Save**. Emails need the Manager's email settings; the PDF is made by the Manager itself. See Settings Reference.

Using the command line

```
cdr recovery plan corp.example.com --restore "dc01.corp.example.com=09/02/2026-23:00" --restore
"dc11.child.corp.example.com=09/02/2026-23:05" --rebuild dc02.corp.example.com --ntp time.win-
dows.com --breach --notify ad-team@corp.example.com --dsrm
cdr recovery plan corp.example.com --restore "dc01.corp.example.com=<version>:rollback" --gc
dc01.corp.example.com --ntp time.windows.com
cdr recovery show <plan> --wait
cdr recovery approve <plan>
cdr recovery secrets <plan> --domain corp.example.com --admin "CORP\Administrator"
cdr recovery secrets <plan> --dsrm
cdr recovery execute <plan>
cdr recovery execute <plan> --confirm corp.example.com --isolated
cdr recovery ack <plan> FR-06@corp.example.com --confirm FR-06
cdr recovery attest <plan> <step>
cdr recovery attest <plan> <step> --reject --note "Encrypted objects still present"
cdr recovery retry <plan> <step>
cdr recovery report <plan> --pdf corp-recovery.pdf
cdr recovery abort <plan> --reason "Re-planning"
```

Other options include `--onto` (bind a candidate to a restored or rebuilt domain controller), `--exclude`, `--rid-increase`, `--reverify`, and `--unattended`. An unattended plan is executed with `--confirm` and `--isolated`. `cdr recovery show` prints each step ID and what a waiting step needs; while the plan is a draft, approved, running, or halted, it also lists the domains whose credential the Manager holds and, for each domain a remaining step needs but the Manager does not hold, the `cdr recovery secrets` command that supplies it. For every option, see Recovery Plan Commands.

Result and next steps

When every step is done, the plan's status is **Completed**. The forest runs on the restored and rebuilt domain controllers and is connected to production. Every domain controller the plan restored or rebuilt that has an enabled backup policy is due for a backup at once; each backup starts when the domain controller's agent connects. Next, read the recovery report, complete its follow-ups, check that those backups completed, and review the audit trail; see After a Recovery.

After a Recovery

A recovery ends when its run shows **Restored** or its plan shows **Completed**, but the directory is not fully back in service until you finish the work Cygna Forest Recovery does not do for you. Use this checklist after every recovery. The longer items apply to domain and forest recoveries; after a single-DC restore, most of the work is a fresh backup and a check of the machine's health.

Before you begin

- Taking backups needs the Backup Operator role. Reviewing the audit trail needs an operator who can view it, such as the Viewer / Auditor role.
- Keep the recovery's page open. Its steps, transcripts, and outcome are the record you work from.

1. Review the outcome

- For a single-DC restore, open the run on the **Recovery Plans** screen and confirm the outcome **Restored and validated**. If it reads **Restored, but a validation gate did not pass**, check replication and dcdiag on the domain controller before you rely on it, then select **Validate again**. If its checks show **Not checked**, the restored DC agent could not make them: update the agent and select **Validate again**, or run `cdr restore validate`.
- After a full system recovery that changed the restored system's address, check **Post-recovery directives** on the run's **Preflight** card. A directive that failed is also written to `POST-RECOVERY-README.txt` on the restored machine.
- For a domain or forest recovery, confirm that every step is **Done**.

2. Read the recovery report and complete its follow-ups

The last step of a domain or forest recovery produces the recovery report and emails it to the plan's notification addresses, with a PDF attached. The Manager makes the PDF itself, with the PDF renderer installed beside it, so there is nothing to set up. If it cannot make

one, the email goes without it. The report is built from the plan's own records each time you ask for it, so you can also open it at any time after the plan starts: select **Open report**, **PDF**, or **Email…** on the plan's page, or run `cdr recovery report`. Emailing it needs `plan.execute`. For the email settings, see Settings Reference.

The report has these sections:

- **Summary**—who built and approved the plan, when it ran and how long it took, how long the directory was live on reduced domain controllers, and how many steps finished.
- **Domain controllers**—what happened to each one, at which step, where, and from which restore point, including whether a restored domain controller came back on a current DC agent.
- **Manual follow-ups**—the work the recovery left for you, grouped by area.
- **Decisions and interventions**—acknowledgments, retries and what each had failed with, wait overrides, rejections, credentials supplied (by whom, never the secret), acceptances made by an unattended plan, and any abort.
- **Steps**—every step, with its detail and each check that did not pass.

The follow-ups come only from what this plan did. Work through each one:

- **Plan**—if the plan did not complete, the steps that did not finish. Everything else is as far as it got.
- **Domain controllers**—on each rebuilt domain controller, the roles that were reinstalled without their configuration and data, such as Active Directory Certificate Services or DHCP, to restore from their own backups; the programs it had, to reinstall if still needed; insecure features left out, such as SMB 1.0; and anything the reinstall could not install. On each restored domain controller, colocated services to check, including directory synchronization services, which need their password synchronization run again after the krbtgt resets. A domain left on a single domain controller needs another one. See What Recovery Does Not Cover.
- **FSMO roles**—the operations master roles were seized onto the restored domain controllers. Move them to their final placement now that the other domain controllers are back.
- **DNS**—return each domain controller's DNS client to your final design, and remove name server and glue records, and WINS records if WINS is in use, that still name removed domain controllers.
- **Trusts**—external and outgoing trusts are not restored from backup. Re-create them with each partner's administrator.
- **Agents**—a restored domain controller that kept the DC agent from its backup. Update it.

- **Backups**–take a fresh backup of every domain controller you back up, rebuilt ones first. If an unattended plan passed over the backup checkpoint, this backup stands in for it.
- **Unattended**–any other step an unattended plan passed over without carrying it out. Carry it out by hand now.
- **Timing**–a gap between the krbtgt resets that someone overrode. Record it if your policy required the gap.
- **Not confirmed**–each check a step could not confirm, for example a time source the root PDC emulator could not reach. Check it by hand.

Beyond the report:

- **DSRM password**–the plan set the new DSRM password you supplied on every domain controller it recovered. Make sure it is stored per your policy; Cygna Forest Recovery does not keep it.
- **Trusts after a forest recovery that was not a breach response**–no trust was reset: each side kept the trust password its own restore point held. If the plan warned that the two sides of a trust were restored from points 30 days or more apart, check that the trust authenticates, for example with `nltest /sc_verify:<domain>`, and reset it if it does not.
- **Breach response only**–plan the remaining password resets for users and services, and update the hosts that use the re-created group managed service accounts.

3. Take fresh backups

Backups taken before the recovery describe the directory as it was before the recovery. Take a new backup of every recovered and rebuilt domain controller as soon as it is healthy, and do not plan another recovery from an older restore point without a reason. A rebuilt domain controller's next backup also records the roles and features a future rebuild reinstalls. A backup taken while the recovery was still changing the directory is labeled with the plan and step, and restoring it raises a warning.

When a recovery plan completes, every domain controller it restored or rebuilt that has an enabled backup policy is made due for a backup at once. Each backup starts when the domain controller's agent connects; if the agent has not connected within 2 hours, the backup is refused and recorded like any other refused backup. While a recovery is still working on a domain controller, its backups wait.

- On the **Backup Jobs** screen, check that the backup of each recovered domain controller completed. Where one did not start or did not complete, run **Back up now – Full system**. See Running a Backup Now.

- Check that every rebuilt domain controller has an enabled backup policy; a domain controller without one is not backed up when the plan completes. See Managing Backup Policies.
- After a full system recovery, install any patches released since the restore point, and then take another backup.

4. Reconnect and clean up machines

- Make sure that none of the original machines a recovery replaced can return to the network. Power each one off, wipe it, or reinstall it cleanly.
- Check that the DC agent on every recovered domain controller is connected on the **Agents** screen. If an agent restored from an old backup is behind the Manager's agent package, update it. If it trusts a previous Manager certificate authority, re-point it; see Maintaining Agents.
- Record domain controllers that no longer exist as removed, and remove their backup policies. See Excluding and Retiring Objects.
- If you moved a domain controller to a new subnet, add the subnet to its site and update anything that reached it by its old address.

5. Re-enable what you paused

- Enable any backup policies you disabled during the incident.
- Clear or update exclusions you added during the incident, so the affected objects count again.
- Restore any operator accounts you disabled, and change passwords that were shared during the recovery, including the break-glass account if it was used. See Managing Operators and Roles.

6. Review the audit trail and readiness

- On **Monitoring** > **Audit Trail**, filter by the recovered machines or the time of the recovery, and review every approval, acknowledgment, attestation, wait override, and abort. Keep the record with your incident notes. See Auditing Operator Activity.
- On the **Dashboard**, confirm that each recovered domain and forest is back to fully recoverable, and resolve any alerts.
- If you also run Cygna Auditor, review its change history for the recovered directory, and its Risk Radar posture assessment for privileged groups, delegation, and other settings an attacker may have changed before the backup was taken. See Related Cygna Products.

Using the command line

```
cdr recovery report <plan> --pdf recovery-report.pdf
cdr backup start dc01.corp.example.com --wait
cdr policy list --disabled
cdr exclusion list --active
cdr audit list --from 2026-09-02 --subject dc01
cdr alert list
```

See Recovery Plan Commands, Backup Commands, and Alert, Audit, and Report Commands.

Administration

Administering Cygna Forest Recovery means keeping the people, settings, and agents behind your backups in good order: who can sign in and what they may do, how the Manager and its agents are configured, whether every agent is current and connected, and what the product has recorded about its own activity.

Most administration happens on the screens in the **Administration** group of the console: **Agents**, **Operators & Roles**, **Configuration**, and **License**. Monitoring happens on the screens in the **Monitoring** group: **Alerts**, **Protection Summary**, and **Audit Trail**. Everything on these screens is also available with the `cdr` command-line tool, except creating, editing, disabling, and resetting operator accounts, which are done in the console.

Who administers what

Cygna Forest Recovery separates administration from backup and recovery. The System Admin role manages operators, configuration, and agents, but it cannot start a backup, change a backup policy, or run a restore. Backup Operators manage backups, and Restore Operators plan and run recoveries. An operator can hold more than one role. For the full table, see Managing Operators and Roles.

Where settings live

The Manager keeps the settings for itself and for every agent in its own database, and pushes agent settings to each agent when they change. Each agent reports back which settings it is running, so the **Configuration** screen shows both what you set and what is in force. A small number of settings, such as the storage root and the retention floor on the storage machine, are set on the machine itself and shown read-only in the console. See Settings Reference.

Watching the installation

The Manager evaluates its alerts continuously and shows them on the **Alerts** screen and in the notifications menu in the top bar. It can also email alerts: set recipients in the **Alert notifications** settings, and each object an alert names is reported when it is raised and when it clears. See Alerts Reference. To feed alerts into your own monitoring, run `cdr alert list` on a schedule: it exits with code 4 whenever an alert is raised, and `--json` gives machine-readable output.

Regular checks that keep an installation healthy:

- **Alerts**—Review raised alerts and act on each one. If alerts are emailed, check that the line on the **Alerts** screen shows **Email on**. See Alerts Reference.

- **Agents**–Check that every agent is connected and that none needs an update. See Maintaining Agents.
- **Manager recovery**–Check the **Manager recovery** card on the **License** screen: the recovery passphrase should be set and the latest database snapshot recent. See Protecting the Manager.
- **Audit trail**–Review operator activity, especially role changes and recovery actions. See Auditing Operator Activity.
- **License**–Check the license's end date and renew before it ends. When the license has ended and its grace period is over, or no key is installed, the Manager refuses new backups and recovery operations. See Licensing.

In this chapter

- Managing Operators and Roles
- Multi-Factor Authentication
- Settings Reference
- Licensing
- Maintaining Agents
- Alerts Reference
- Auditing Operator Activity
- Logs and Diagnostics

Managing Operators and Roles

Everyone who uses the console or the `cdr` command-line tool signs in as an operator. Operator accounts belong to Cygna Forest Recovery itself: they are stored in the Manager database, not in Active Directory and not as Windows local accounts or groups. That is what lets you sign in and run a recovery while the directory is down. Each account signs in with a username and password and, if the operator turns on multi-factor authentication (MFA), a code from an authenticator app. See Multi-Factor Authentication.

You manage operators on the **Operators & Roles** screen, which needs the System Admin role (or the break-glass account). For a description of the screen itself, see Operators and Roles Screen.

Roles

An operator's roles decide what they can see and do. The Manager checks the operator's capabilities on every request, from the console and from `cdr` alike. The console also hides

the screens and buttons an operator cannot use; a screen opened directly without the capability shows **Not permitted**, with the capability it needs and the roles that hold it.

Role	Can	Cannot
Viewer / Auditor	See everything: the dashboard and alerts; forests, domains, sites, domain controllers, and agents; backup policies, backup jobs, restore points, OS media, and the repository; restore runs and recovery plans; reports; and the audit trail.	Change anything.
Backup Operator	Everything a Viewer / Auditor can see, and also create, edit, and remove backup policies; start and cancel backups; exclude objects from alerts; and record domain controllers, domains, and trusts as removed from the directory.	Plan, approve, or run a restore or recovery plan. Manage operators, configuration, or agents.
Restore Operator	Everything a Viewer / Auditor can see, and also plan, approve, and discard restores and recovery plans; ask a machine to describe itself again; execute, cancel, or close a restore as abandoned; execute or abort a recovery plan; acknowledge points of no return, attest operator steps, supply domain credentials to a plan, and shorten the wait between krbtgt resets.	Change backup policies or start backups. Manage operators, configuration, or agents.
System Admin	Everything a Viewer / Auditor can see, and also create operators and assign their roles; reset passwords and MFA, unlock, disable, enable, remove, and restore accounts; change Manager and agent settings; install the DC agent and update, restart, and uninstall agents from the console; and set the recovery passphrase.	Change backup policies, start or cancel backups, or plan, approve, or run any restore or recovery plan.
Break-glass	Everything. This is not a role you can assign: it belongs to the one break-glass account.	—

An operator can hold several roles and gets every capability of each. Every operator must hold at least one role, because an account with none could sign in and see nothing.

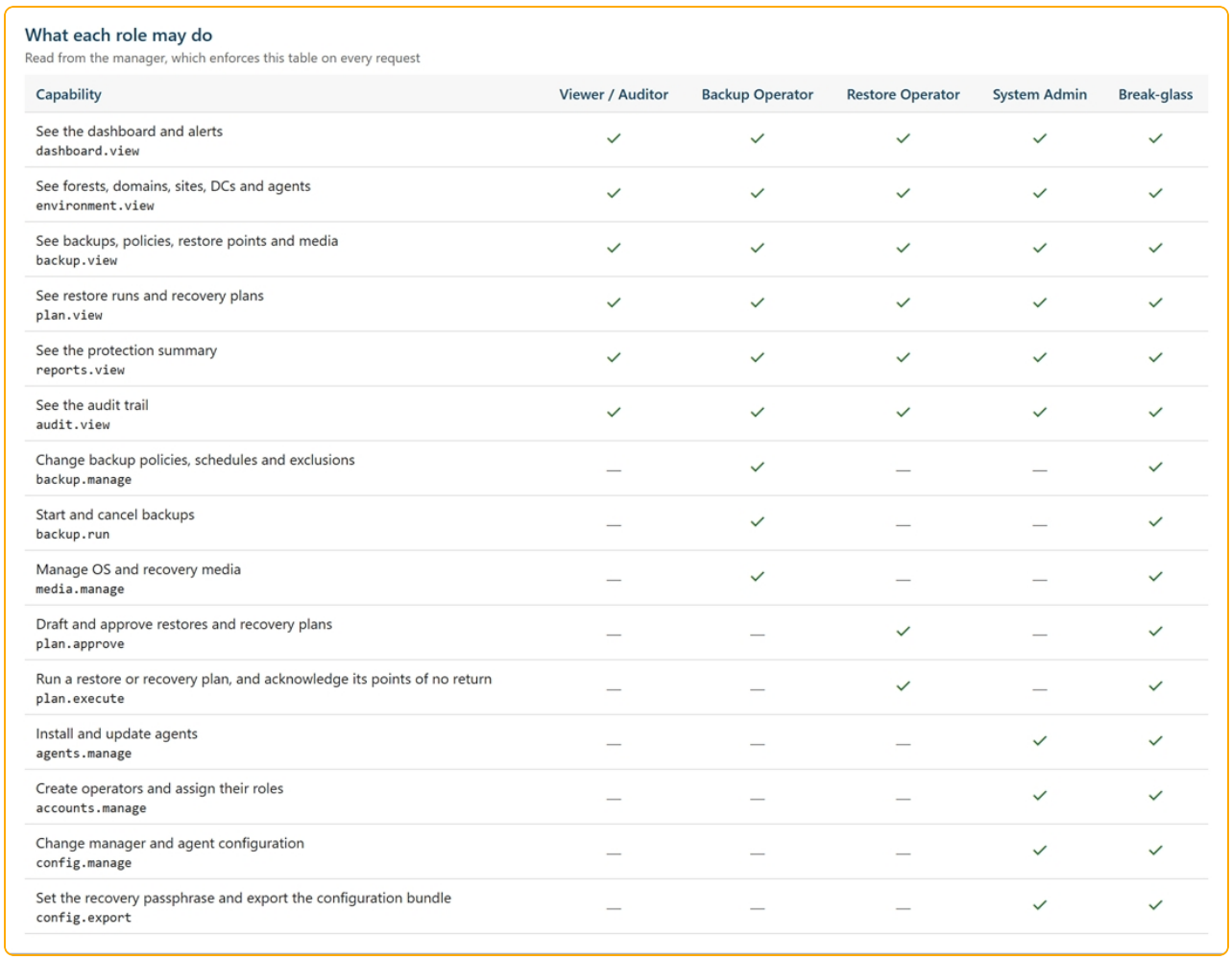
Capabilities

The **Roles** tab lists the capabilities behind each role. The **Not permitted** screen and the command-line tool name capabilities by code, so this table maps the codes to what they allow.

Capability	Allows	Held by
`dashboard.view`	See the dashboard and alerts	Every role
`environment.view`	See forests, domains, sites, DCs and agents	Every role
`backup.view`	See backups, policies, restore points and media	Every role
`plan.view`	See restore runs and recovery plans, and open or download a plan's recovery report	Every role
`reports.view`	See the protection summary	Every role
`audit.view`	See the audit trail	Every role
`backup.manage`	Change backup policies, schedules and exclusions	Backup Operator
`backup.run`	Start and cancel backups	Backup Operator
`plan.approve`	Draft and approve restores and recovery plans, accept the preflight gates that need an approver's acceptance, and ask a restored domain controller to validate itself again	Restore Operator
`plan.execute`	Run a restore or recovery plan, and acknowledge its points of no return. Also: run a plan unattended, take an unattended plan back, retry a halted step, set a plan's notification addresses (including when the plan is created), and email its recovery report	Restore Operator
`agents.manage`	Install and update agents from the console, approve and drain Isolated Recovery Hosts, and delete an agent's record with `cdr agent forget`	System Admin
`accounts.manage`	Create operators and assign their roles; reset passwords and MFA; unlock, disable, enable, remove, and restore accounts	System Admin

Capability	Allows	Held by
`config.manage`	Change Manager and agent configuration	System Admin
`config.export`	Set the recovery passphrase, and create, download, and remove support packages	System Admin

Note: The break-glass account holds every capability in the table. `cdr agent deploy` needs no Manager capability to install agents, because it installs with the Windows credential you give it; the console's install and update actions, and `cdr agent forget`, need `agents.manage`.



What each role may do

Read from the manager, which enforces this table on every request

Capability	Viewer / Auditor	Backup Operator	Restore Operator	System Admin	Break-glass
See the dashboard and alerts `dashboard.view`	✓	✓	✓	✓	✓
See forests, domains, sites, DCs and agents `environment.view`	✓	✓	✓	✓	✓
See backups, policies, restore points and media `backup.view`	✓	✓	✓	✓	✓
See restore runs and recovery plans `plan.view`	✓	✓	✓	✓	✓
See the protection summary `reports.view`	✓	✓	✓	✓	✓
See the audit trail `audit.view`	✓	✓	✓	✓	✓
Change backup policies, schedules and exclusions `backup.manage`	—	✓	—	—	✓
Start and cancel backups `backup.run`	—	✓	—	—	✓
Manage OS and recovery media `media.manage`	—	✓	—	—	✓
Draft and approve restores and recovery plans `plan.approve`	—	—	✓	—	✓
Run a restore or recovery plan, and acknowledge its points of no return `plan.execute`	—	—	✓	—	✓
Install and update agents `agents.manage`	—	—	—	✓	✓
Create operators and assign their roles `accounts.manage`	—	—	—	✓	✓
Change manager and agent configuration `config.manage`	—	—	—	✓	✓
Set the recovery passphrase and export the configuration bundle `config.export`	—	—	—	✓	✓

Separation of duties

The roles are built so that no single role reaches from one area into another:

- Administering the Manager does not carry backup or recovery rights. A System Admin can configure retention and agents but cannot run a restore.
- Recovery rights do not carry backup policy control, and backup rights do not carry recovery.
- Nobody can change their own roles. Another operator with `accounts.manage` has to make the change, so nobody grants themselves a capability without a second person.
- No administrator can change the break-glass account.
- An action that crosses two roles needs both capabilities. Replacing the storage machine changes which agent holds the backups and ends every restore point, so it needs `agents.manage` and `backup.manage`. An operator holding both, such as the break-glass account, does it alone. Otherwise one operator asks, and an operator holding the other capability approves: a System Admin's request waits for a Backup Operator, and a Backup Operator's for a System Admin. A request lapses after 24 hours, and each step is audited. See Requests.

Two things to plan for:

- **Approve and execute are in the same role.** Approval and execution are separate actions, and the Manager records who did each. But both capabilities belong to Restore Operator, so one operator can approve a restore and then execute it. If your policy requires two people, give the Restore Operator role to a small group, agree that the person who approves a recovery and the person who executes it are different people, and check the **Restore approved** and **Restore execution requested** entries in the audit trail.
- **Account administration can create accounts with any role.** A System Admin cannot change their own roles, but can create a new operator with the Restore Operator role. Restoring a removed operator assigns roles in the same way. Limit who holds System Admin, and review **Operator created**, **Operator roles or name changed**, and `operator.restored` entries in the audit trail.

The break-glass account

The break-glass account, `breakglass`, is created at the first sign-in to a new Manager. The Manager generates its password and shows it once, and you set the recovery passphrase in the same step. See Signing In for the First Time.

- It holds every capability.
- It is never locked out by wrong passwords. An address that has made too many failed sign-ins is refused for it as for any account, except on the Manager machine itself: a sign-in there is never refused for its address, so the account always has a way in from the Manager's console.
- A password it chooses for itself must be at least 20 characters.

- No administrator can change its roles, disable it, or reset its password or MFA. It changes its own password, and turns MFA on or off, from the account menu while signed in as itself.
- It can use MFA like any other account. If it does, keep its recovery codes in the safe with its password: they are its only way in without the phone. Wrong authenticator codes lock its codes as they do for any account, and its recovery codes still work while they are locked.
- It comes back with the Manager database after a Manager recovery, which makes it the one account certain to work on a rebuilt Manager.

Keep the break-glass credential offline, with the recovery passphrase, and use it only when no named account will do. Create named operators for day-to-day work, so the audit trail records who did what. When you change the break-glass password, update the offline record in the same sitting.

Creating an operator

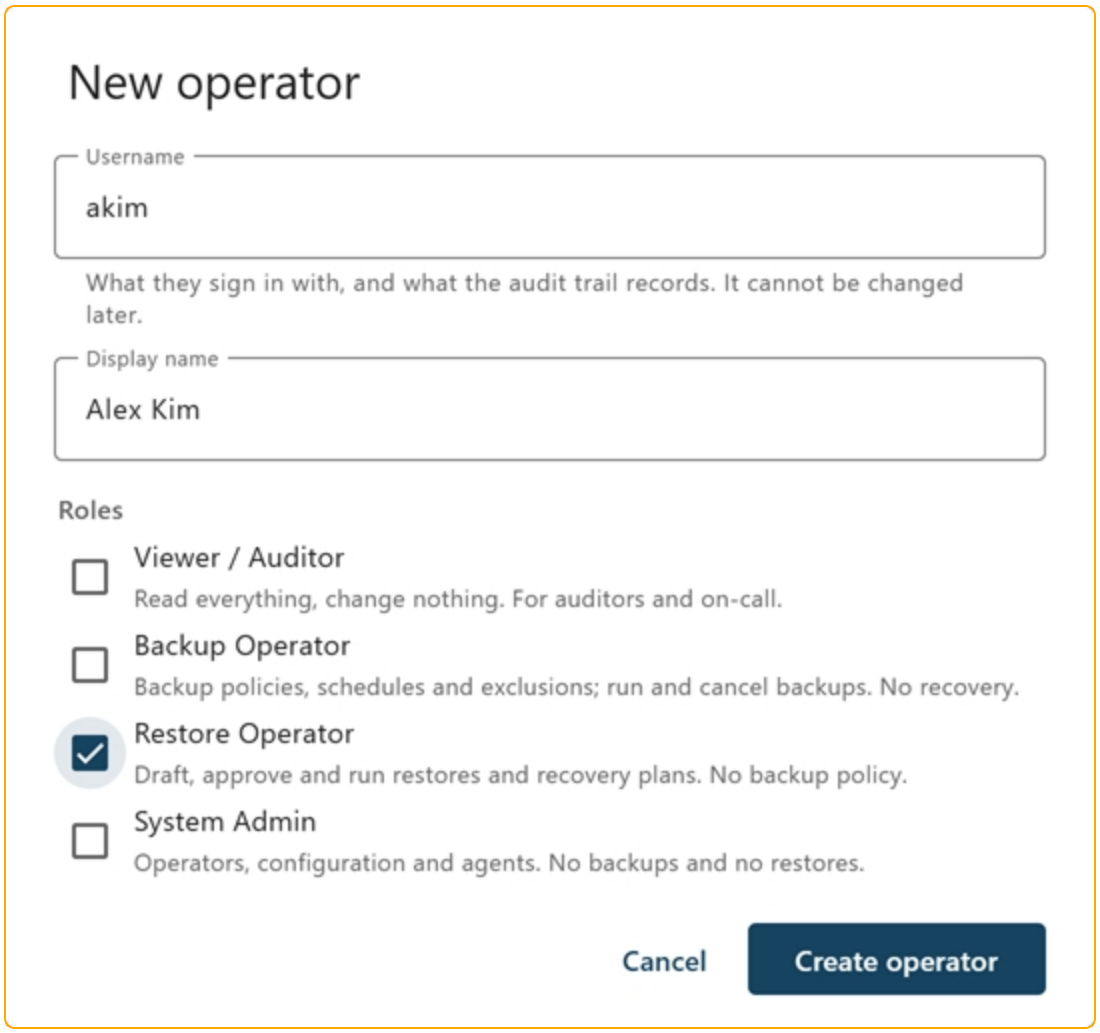
To create an operator:

1. In the console, select **Administration** > **Operators & Roles**.
2. Select **New operator**.
3. In **Username**, enter the name the operator signs in with.

   A username is 2 to 64 characters: letters, digits, dot, hyphen, and underscore, starting with a letter or digit. It is what the audit trail records, and it cannot be changed later. Names that the product uses for itself, such as `system`, `manager`, `operator`, and `breakglass`, are reserved.
4. In **Display name**, enter the person's name.
5. Under **Roles**, select one or more roles.
6. Select **Create operator**.

   The Manager generates a temporary password and shows it once.
7. Select **Copy**, and give the temporary password to the operator by a channel you trust.
8. Select **Done**.

   The operator's status is **Temporary password** until they sign in. At their first sign-in, in the console or with `cdr login`, they must choose their own password before they get a session.



Changing an operator

Open the row menu for an operator on the **Operators** tab to change it:

- **Edit name and roles…**—Change the display name and roles. On your own row this reads **Edit display name…**, because you cannot change your own roles.
- **Reset password…**—Issues a new temporary password, shown to you once. The operator's current password stops working at once, and every session they have, in the console and in `cdr`, ends. Not available on your own row: change your own password from the account menu.
- **Reset MFA…**—For an operator who has lost their phone and their recovery codes. Turns their MFA off, so they sign in with their password alone until they set it up again, and ends every session they have. Shown only for operators who use MFA, and not on your own row. See Multi-Factor Authentication.
- **Unlock**—Clears a lockout, or a lock on MFA codes, before it expires. On the break-glass account's row, **Unlock MFA codes** clears a lock on its MFA codes.

- **Disable… / Enable…**—A disabled operator cannot sign in, and their sessions end on their next request. The account and its history are kept. You cannot disable your own account.
- **Remove…**—For someone who has left. See the next section. Not available on your own row.

If another administrator removes or restores the account while you make one of these changes, the Manager refuses your change and says it was not saved; reload and try again. A refused change ends none of the operator's sessions.

Removing and restoring an operator

Operators are never deleted. The audit trail, recovery plans, and approvals name operators by username, so deleting an account would free its name for someone else, who would then appear to have done everything the old account did. Use one of these instead:

- **Disable**—For a temporary absence, or a credential you suspect. The account keeps its roles and MFA, stays in the list, and **Enable…** brings it back as it was.
- **Remove**—For someone who has left. Every session ends at once, in the console and in `cdr`. The password, MFA, and roles are cleared, not kept. The account stays in the audit trail, its username stays reserved, and the **Operators** tab hides it unless you show removed operators. A sign-in with a removed username fails in the same way as one with an unknown username.

You cannot remove your own account or the break-glass account. A removed account cannot be edited, enabled, reset, or unlocked; restore it first. Creating an operator with a removed operator's username is refused, with who removed the account and when: if it is the same person returning, restore the account; otherwise choose another username.

To remove an operator:

1. In the console, select **Administration** > **Operators & Roles**.
2. Open the row menu for the operator, and select **Remove…**.
3. Check the consequences listed, and select **Remove the operator**.

   The account disappears from the list. The audit trail records the removal under your name, with the roles the account held.

Restore an account only for the same person returning. Nothing comes back from before: the account gets a new temporary password, no MFA, and only the roles you choose now.

To restore a removed operator:

1. On the **Operators** tab, select **Show removed**.
2. Open the row menu for the removed operator, and select **Restore…**.

3. Check the **Display name**, and change it if needed.
4. Under **Roles**, select one or more roles.

   No role is selected to begin with, because the old roles were cleared at removal.
5. Select **Restore operator**.

   The Manager generates a temporary password and shows it once.
6. Select **Copy**, give the temporary password to the operator by a channel you trust, and select **Done**.

   As for a new account, the operator must choose their own password at their next sign-in. MFA stays off until they set it up again from their account menu.

You can also list, remove, and restore operators with `cdr operator list`, `cdr operator remove`, and `cdr operator restore`. See Connecting and Signing In.

Passwords, lockout, and sessions

Rule	Value
Password length	At least 12 characters, at most 1,024
Password content	Must not contain the username. A new password must differ from the current one.
Lockout	5 failed sign-ins in a row lock the account for 15 minutes. From any one address, no more than 10 failed sign-ins in 15 minutes, for any accounts. A wrong current password in **Change password** counts as a failed sign-in. The break-glass account is never locked by wrong passwords, and is refused for its address only when it does not sign in on the Manager machine itself. A refused sign-in does not say why; the audit trail does. Wrong MFA codes have a lock of their own, which covers every account.
Session lifetime	12 hours from sign-in, however active the session is
Idle timeout	30 minutes in the console; 4 hours for `cdr`. The console counts only keyboard, pointer, and touch input, so a console left open is signed out 30 minutes after it was last used, even while its screens refresh.
Recovery passphrase	At least 16 characters

No administrator ever chooses another operator's password: the Manager generates temporary passwords, and each operator chooses their own. To change your own

password, open the account menu in the top bar and select **Change password**. Your other sessions, in another browser or in `cdr`, end when you save.

`cdr` keeps its session for the Windows user who signed in, protected with Windows data protection, until the session expires or you run `cdr logout`.

Related topics

- Operators and Roles Screen
- Multi-Factor Authentication
- Auditing Operator Activity
- Accounts and Permissions

Multi-Factor Authentication

Multi-factor authentication (MFA) adds a second step to signing in. After the password, Cygna Forest Recovery asks for the 6-digit code that an authenticator app on your phone shows, such as Microsoft Authenticator, Google Authenticator, or Authy. Someone who learns your password cannot sign in without the phone. MFA applies to the console and to the `cdr` command-line tool alike.

Neither the Manager nor the phone needs the internet for MFA. The app computes each code from a secret it shares with the Manager and from the current time, so the only requirement is that the two clocks agree to within about 30 seconds.

How MFA works

- **Each operator turns it on for themselves**—from the account menu, which proves the phone works before anything depends on it. An administrator cannot turn MFA on for someone else, because binding an account to a phone the administrator chose would hand them the account. An administrator can only turn it off, for an operator who has lost their phone. See When an operator loses their phone.
- **The code is asked for after the password**—so the sign-in page reveals nothing about MFA to someone who does not have the password.
- **Each code works once**—a code that someone saw over your shoulder cannot be used again, even within its 30 seconds.
- **Recovery codes**—turning MFA on gives you ten one-time recovery codes. Each signs you in once in place of an authenticator code, so a lost phone does not lock you out.
- **The break-glass account can use it too**—if you turn MFA on for the break-glass account, keep its recovery codes in the safe with its password.

Rule	Value
Authenticator code	6 digits, time-based, changing every 30 seconds. A code from the step before or after the current one is accepted, to allow for a small clock difference.
Recovery codes	Ten, shown once, each good for one sign-in. Each is ten letters and digits in two groups of five. You can type it with or without the hyphen, in upper or lower case.
Setup	Must be finished with a code from the app within 10 minutes of starting it.
Code lock	5 wrong authenticator codes in a row lock authenticator codes for 15 minutes, for every account, including break-glass. Recovery codes are never locked. See Wrong codes and the code lock.

Before you begin

- Install an authenticator app that supports time-based codes on your phone.
- Check that the clock on your phone and the clock on the Manager are both right.
- Decide where you will store your recovery codes: somewhere other than the phone, such as on paper or in a password manager. For the break-glass account, the safe that holds its password.

Turning on MFA

To turn on MFA for your account:

1. Open the user menu at the right of the top bar and select **Multi-factor authentication**.

   The menu item shows **Off** while MFA is off. The **Multi-factor authentication** dialog opens.

2. In **Your password**, enter your password, and select **Set up**.

   The dialog shows a QR code and, beside it, the same secret as a key you can type.

3. Scan the QR code with your authenticator app.

   If your app cannot scan it, add the account in the app by typing the key shown under **Cannot scan? Type this key instead (time-based)**, or select **Copy key**. The app lists the account under the product name, with your username and the Manager's machine name, so you can tell two Managers apart.

4. In **Code from the app**, enter the 6-digit code the app shows, and select **Turn on**.

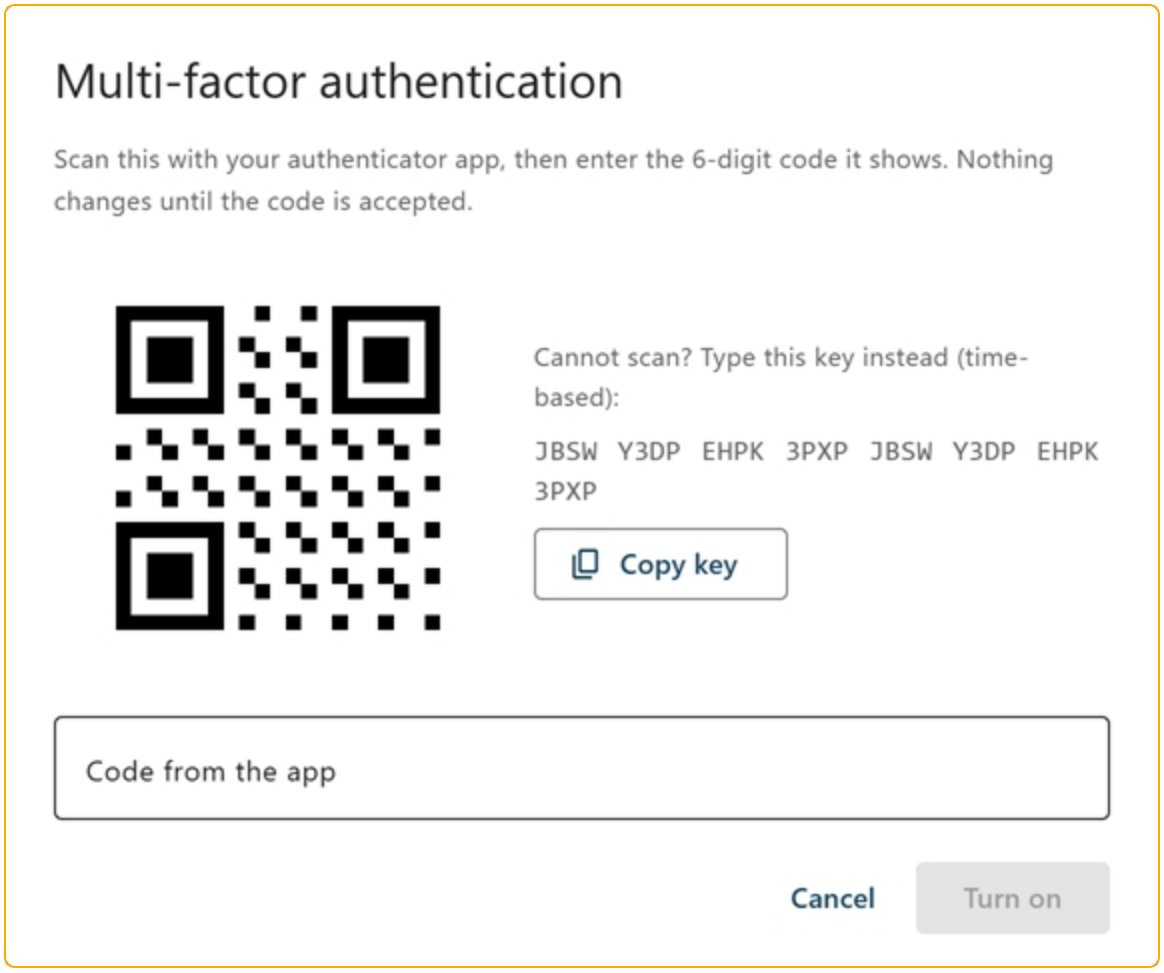
Nothing changes until the code is accepted. The dialog then shows your ten recovery codes.

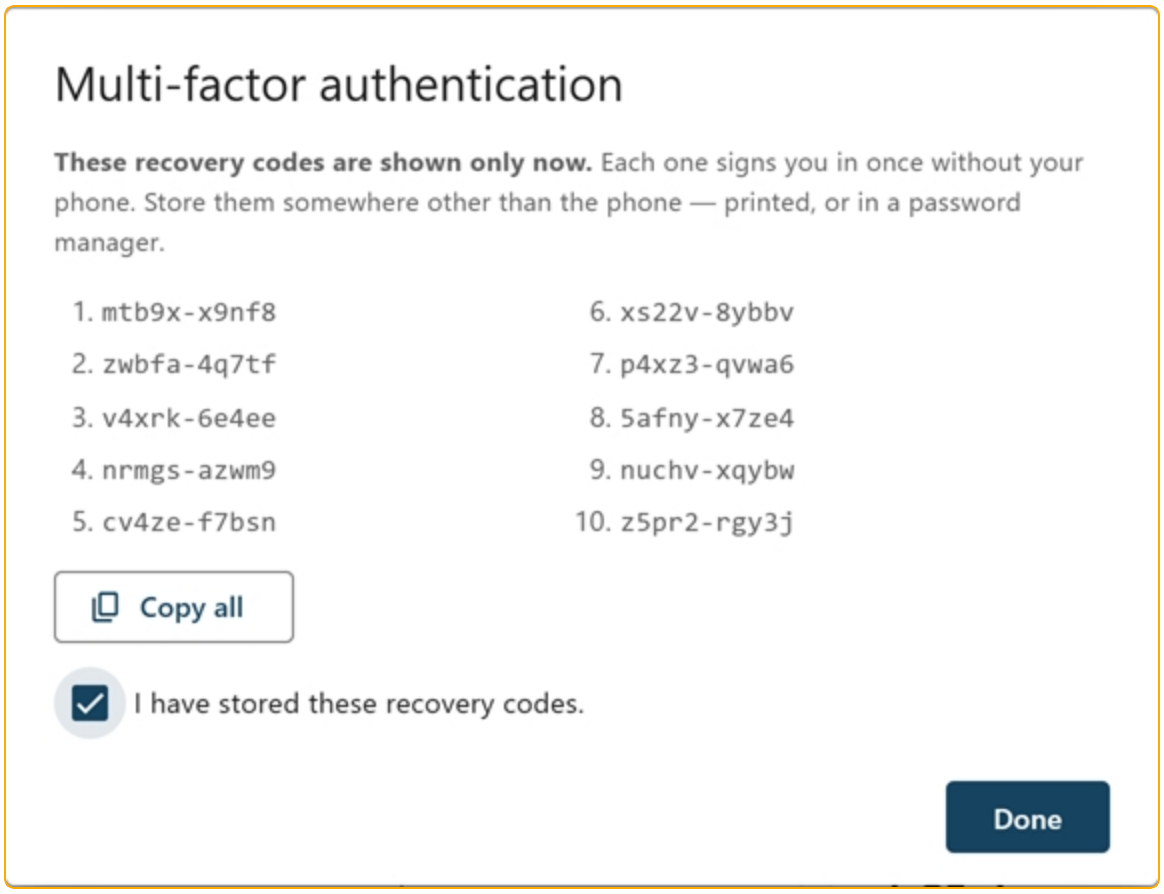
5. Store the recovery codes somewhere other than the phone. Select **Copy all** if you want to copy them, for example into a password manager.

   The codes are shown only now. They cannot be shown again.

6. Select **I have stored these recovery codes.**, and then select **Done**.

   The console confirms with **Multi-factor authentication is on. Your other sessions were signed out.** Sessions in other browsers and in `cdr` end, because they were opened with the password alone. The user menu now shows **On**.



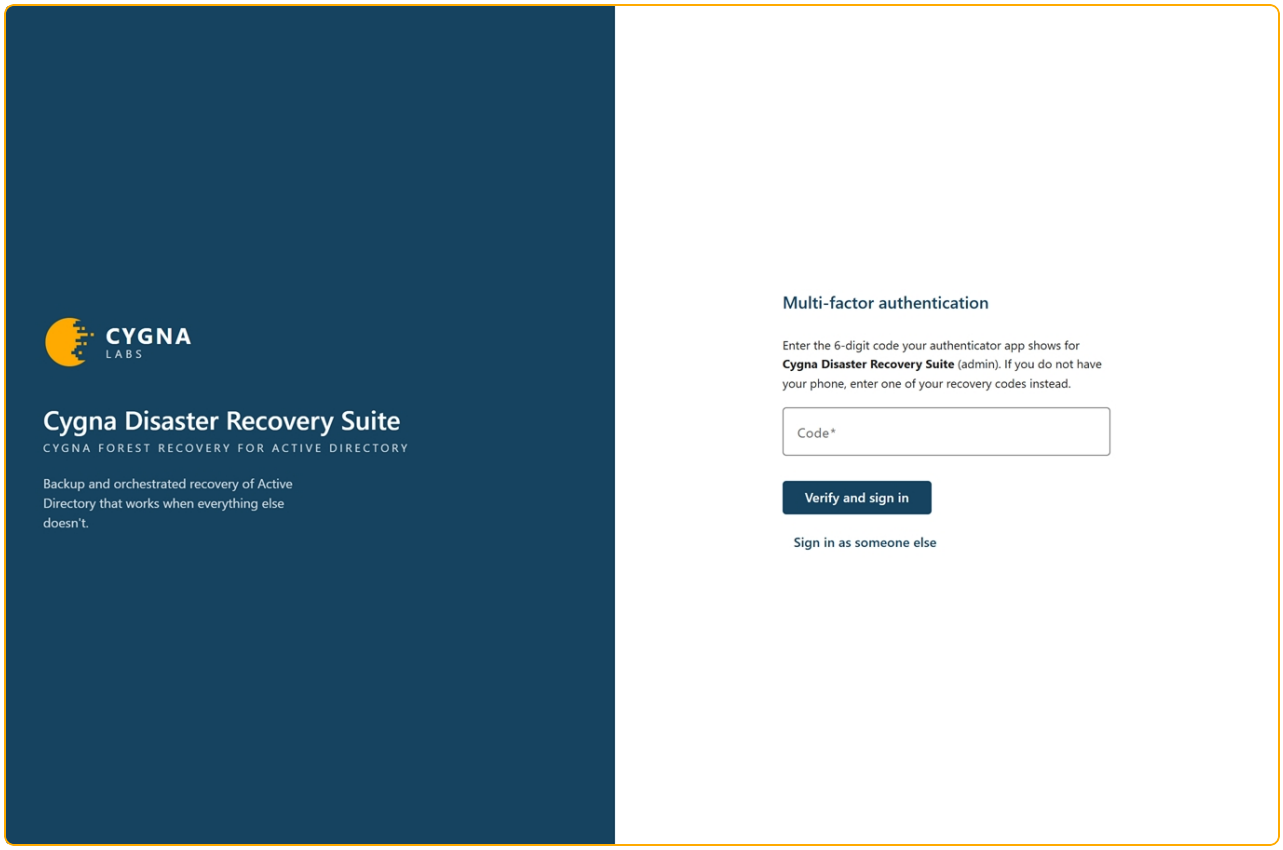


If the setup takes longer than 10 minutes, the dialog says **That setup has expired or was never started. Start it again to get a new code to scan.** Close the dialog and start again; delete the unused entry from your app.

Signing in with MFA

In the console, sign in with your username and password as usual. The sign-in page then shows **Multi-factor authentication** and asks for the code your authenticator app shows for this account. Enter it in **Code** and select **Verify and sign in**. If you do not have your phone, enter one of your recovery codes instead. To start again with another account, select **Sign in as someone else**.

If an administrator has issued you a temporary password, you choose your new password first, and the code is asked for after that.



With `cdr`, `cdr login` prompts for the code after the password, or takes it with `--code`. See Connecting and Signing In.

When a code is refused, the sign-in page says **That code is not correct. Codes change every 30 seconds; if a fresh one is refused too, check that the clock on your phone and on the manager are both right.** Wait for the next code and try again.

Managing your recovery codes

Open the user menu and select **Multi-factor authentication** to see how many unused recovery codes you have. `cdr whoami` shows the count too. When none are left, the dialog says **You have no recovery codes left.** Replace them immediately: without one, a lost phone needs an administrator, and locks the break-glass account out for good.

To replace your recovery codes:

1. Open the user menu and select **Multi-factor authentication**.
2. Select **New recovery codes**.
3. Enter **Your password**, and in **Code from the app, or a recovery code**, enter a current code.
4. Select **Replace the codes**.

The dialog shows ten new codes. Every earlier code stops working, used or not.

5. Store the new codes, select **I have stored these recovery codes.**, and then select **Done**.

   The console confirms with **New recovery codes saved. Your old ones no longer work.**

Each time you sign in with a recovery code, the audit trail records it, with the number of codes left.

Turning off MFA or moving it to a new phone

To move MFA to a new phone, turn it off and set it up again on the new phone. Turning it off needs your password and a code. If the old phone is gone, a recovery code works.

To turn off MFA for your account:

1. Open the user menu and select **Multi-factor authentication**.
2. Select **Turn off**.
3. Enter **Your password**, and in **Code from the app, or a recovery code**, enter a current code.
4. Select **Turn off**.

   The console confirms with **Multi-factor authentication is off. You now sign in with your password alone.** Your authenticator entry and recovery codes stop working.

To set MFA up on the new phone, follow Turning on MFA again, and delete the old entry from the old app if you still have it.

Wrong codes and the code lock

Wrong authenticator codes are limited separately from wrong passwords. After 5 wrong codes in a row, authenticator codes are refused for 15 minutes. After that, each further wrong code locks them again, and the lock doubles for every 5 more wrong codes, up to 24 hours. A correct code or a recovery code ends the run.

- **It covers every account**—including the break-glass account, which is never locked by wrong passwords.
- **Recovery codes are never locked**—while authenticator codes are locked, a recovery code still signs you in. Someone else's guessing cannot lock you out.
- **A lock means your password is known**—only someone with the right password reaches the code step. If you did not enter the wrong codes, change your password.

While codes are locked, the sign-in page says **Too many wrong codes: authenticator codes are refused for now. One of your recovery codes still works, or an administrator**

can unlock the account. If you did not enter those codes, someone else has your password: change it., with the time the lock ends.

On the **Operators & Roles** screen, the operator's **MFA** column shows **On, codes locked**. An administrator can end the lock with **Unlock** in the row menu, or with **Unlock MFA codes** on the break-glass account's row.

The checks in the **Multi-factor authentication** dialog follow the same rules: a wrong password there counts toward the account lockout, and a wrong code toward the code lock.

When an operator loses their phone

- **They still have their recovery codes**—they sign in with a recovery code, turn MFA off with another one, and set it up again on the new phone. No administrator is needed.
- **They have lost their recovery codes too**—an administrator resets their MFA, as described below.
- **The break-glass account**—no administrator can reset it, so its recovery codes are the only way in without its phone. Keep them in the safe with its password.

To reset an operator's MFA:

1. Confirm that it is really the operator asking.

   A reset lets anyone with their password sign in.

2. Select **Administration** > **Operators & Roles**.

   Resetting MFA needs the `accounts.manage` capability.

3. Open the operator's row menu and select **Reset MFA…**.

   The confirmation explains that their authenticator enrollment and recovery codes stop working, that every session they have in the console and in `cdr` ends, and that the reset is recorded in the audit trail under your name.

4. Select **Reset MFA**.

   The operator signs in with their password alone until they set MFA up again from their account menu. If their phone was stolen, consider resetting their password as well.

Reset MFA… appears only for operators who use MFA. It is not available on your own row (turn off your own MFA from your account menu) or on the break-glass account's row.

MFA and Manager recovery

Each operator's authenticator secret is stored in the Manager database, protected for the Manager machine with Windows data protection. How it comes back depends on how you recover the Manager:

- **From a backup of the whole machine**–MFA works as before.
- **From a configuration backup**–the backup carries every operator's MFA enrollment, encrypted under the recovery passphrase like the rest of its contents, and the new Manager protects them again for its own machine. MFA works as before.
- **From a database snapshot**–the new machine cannot read the secrets. An operator who used MFA gets **This manager cannot read your authenticator enrolment: its database was moved from another machine. Sign in with one of your recovery codes, or ask an administrator to reset your MFA.** After signing in with a recovery code, they turn MFA off and set it up again. The break-glass account signs in with its password alone, and MFA is turned off for it, with an entry in the audit trail; turn it on again once the Manager is running.

Support packages never contain authenticator secrets or recovery codes. For the recovery routes, see Recovering the Manager.

MFA in the audit trail

The audit trail records every change to an operator's MFA and every sign-in step that involves it. The **Action** column shows these actions:

Action	Recorded when
`operator.signin.mfa-challenged`	The password was right and the Manager asked for a code. One of these without a sign-in after it means someone tried the password and stopped at the code.
`operator.signin.failed`	A sign-in was refused. For MFA the reason is `bad-code`, `mfa-locked`, or `mfa-unreadable`.
`operator.mfa.enabled`, `operator.mfa.disabled`	An operator turned MFA on or off for their own account.
`operator.mfa.recovery-code-used`	A recovery code was used, with the number left.
`operator.mfa.recovery-codes.replaced`	An operator replaced their recovery codes.
`operator.mfa.challenge.failed`	A wrong password or code was entered in the **Multi-factor authentication** dialog.
`operator.mfa.locked`	Wrong codes locked authenticator codes, with when the lock ends. Someone other than the operator may have their password.

Action	Recorded when
`operator.mfa.reset`	An administrator reset an operator's MFA. The actor is the administrator.
`operator.mfa.unreadable`	The break-glass account signed in on a Manager that could not read its authenticator secret, and MFA was turned off for it.

An administrator ending a code lock is recorded as **Account unlocked** (`operator.unlocked`). For reviewing the trail, see Auditing Operator Activity.

Related topics

- Managing Operators and Roles
- Operators and Roles Screen
- Using the Console
- Connecting and Signing In

Settings Reference

This topic lists every setting you can see on the **Configuration** screen, grouped as the screen groups them: Manager, DC agent, storage agent, and Isolated Recovery Host agent. For each it gives the label shown in the console, what it does, its default, the values it accepts, and any flags that change how it is saved or applied.

You change settings on the **Configuration** screen, which needs the System Admin role, or with `cdr config set`. `cdr config show` prints every setting with its key, its default, and whether anyone has changed it. For the screen, see Configuration; for the commands, see Configuration and Exclusion Commands.

Settings chosen at installation, such as the Manager's ports and the agents' Manager address, are installer properties rather than settings on this screen. See Installing the Manager.

How settings are applied

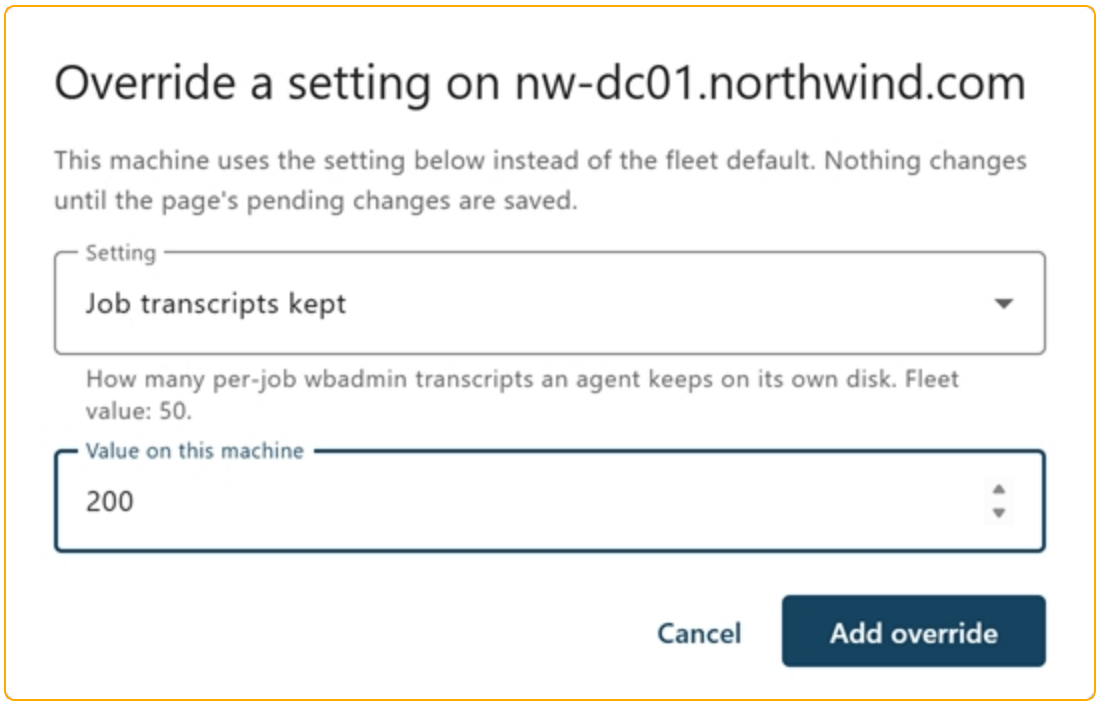
The Manager holds the settings for itself and for every agent in its database. When you save, it pushes agent settings to every connected agent at once; an agent that was offline picks up the change on its next connection. Every agent caches the last values it received and keeps running on them if it cannot reach the Manager. Each agent reports the configuration revision it is running, and the **Machines in this scope** card on each agent tab shows whether each agent is on the current configuration.

Which value wins

For a setting managed from the console, an agent uses, in order:

1. An override for that one machine, if there is one.
2. The fleet value set on the Configuration screen.
3. The product default.

To give one machine a different value, open the agent tab on the **Configuration** screen, find the machine in **Machines in this scope**, and select **Override a setting on this machine…**. The override is saved with the page's other pending changes. **Follow the fleet default** removes it. With `cdr`, use `cdr config set <key> <value> --agent <agent id>`. Per-machine overrides are most useful for trying a change, such as **Heartbeat interval**, on one domain controller before the whole fleet.



Settings set on the machine

Settings flagged **Set on the machine** are read from the machine's own configuration and are shown read-only in the console, as each agent reports them. They are the values that must be known before the service can reach the Manager, such as the Manager address and the storage root, and the storage machine's deletion floors, which are kept out of the Manager's reach on purpose.

Each component reads two local sources:

- **The registry key the installer writes**—`HKLM\SOFTWARE\Cygna Labs\Disaster Recovery Suite\<Manager | DC Agent | Storage Agent | Isolated Recovery`

`Host Agent>\Configuration`, which holds the values given at installation. It is locked to SYSTEM and Administrators.

- **appsettings.json**—The file in the component's program folder. It is read above the registry, so a value written in its `CygnaDR` section wins over the installed one.

To change a local value, add it to the `CygnaDR` section of `appsettings.json` and restart the service. Do not add a key with an empty value to fall back to the installed one: an empty string is a value, and on the log folder settings it turns the file log off. Remove the key instead. To move an agent to a new Manager address or fingerprint, reinstall it with the new installer property rather than editing the file; see Maintaining Agents.

Note: A setting that is managed from the console is ignored if it also appears in a machine's `appsettings.json`. The Configuration screen names any such keys under the machine, so you can remove them from the file. On the Manager, keys set in `appsettings.json` before an upgrade are imported into the database once, on the first start after the upgrade; after that the database wins.

Update policy

One setting decides whether an agent accepts updates from the Manager. Every agent kind shows it, read-only, as **Allow updates from the manager**. It is never set from the console. It is read from the machine's policy key, which no installer creates, changes, or removes, so an upgrade cannot undo it and Group Policy can set it across many machines:

```
HKLM\SOFTWARE\Policies\Cygna Labs\Disaster Recovery Suite\<agent>\CygnaDR\Update
```

`<agent>` is `DC Agent`, `Storage Agent`, or `Isolated Recovery Host Agent`. The values can be strings (`true` or `false`) or REG_DWORD values (0 is false, anything else is true). Restart the agent after a change.

- `AllowRemote`—Default `true`. Set it to `false` to stop the agent accepting updates from the Manager. A DC agent is then updated by push install, and a storage agent or Isolated Recovery Host agent by hand.

Any other value under the key is ignored, and the agent says so when it starts.

Flags

The console marks settings with these flags:

Flag	Meaning
Restart required	Saved at once, but not in force until the service restarts. The machine's entry on the Configuration screen lists the settings that are waiting for a restart.
Can delete data	Changing the setting can delete data. Saving it needs you to type `CONFIRM`, and `cdr config set` asks for confirmation (or `--yes` when there is no terminal).
Cannot weaken the machine's floor	The storage machine takes the more protective of its own local value and the value from the Manager. When it declines a value, the console shows the value it sent and the value in use. Weakening such a setting has to be done on the storage machine itself.
Set on the machine	Read from the machine's own configuration and shown read-only in the console.

The Manager also checks settings against each other when you save. It refuses a save that leaves the storage agents' **Reservation timeout** less than 30 minutes longer than the Manager's **Backup job timeout**, and it warns when a retention ceiling is below the storage machine's own floor, because the floor wins.

Manager settings

These settings are read by the Manager. Unless a setting is flagged **Restart required**, a change takes effect on the next sweep or dispatch.

Backup execution

Setting	Description	Default	Range	Flags
Backup job timeout	How long the Manager waits for a DC agent to report the result of a backup. A timeout does not stop the backup on the domain controller, so a value that is too low records a failure for a backup that is still running. Size it against your slowest backup under load, and keep it at least 30 minutes below the storage agents' **Reservation timeout**.	240 minutes	1-10,080	–
Vault commit timeout	How long the Manager waits for the storage agent to report that a backup has been hashed and moved into the vault. Hashing reads every byte, and commits run side by side, so size it against your largest backup with other backups running. A commit reported after this timeout is recorded as confirmed late, with the advice to raise it. A commit reported late for another reason, such as a Manager restart, names that reason instead.	90 minutes	1-10,080	–

Setting	Description	Default	Range	Flags
Maximum concurrent backups	How many domain controllers may back up at once.	4	1-512	Restart required
Schedule jitter	A random spread applied to each policy's interval, so that many policies do not fire at the same moment. 0 means every policy fires exactly on its interval. Applies to policies with no start time; a policy with a start time uses **Start time spread** instead.	5%	0-50	–
Start time spread	How long after its start time a policy may be due, so that domain controllers sharing a start time do not all ask for a backup slot at once. Never before the start time, and at most half the policy's interval. The backup slots (**Maximum concurrent backups**) then decide when each backup actually starts. 0 means every policy with a start time is due exactly at it.	30 minutes	0-240	–
Size estimate sample count	How many recent backups of a domain controller are averaged to estimate the size of its next	5	1-100	–

Setting	Description	Default	Range	Flags
	backup. The storage agent reserves vault space against this estimate before it accepts a backup.			
Restart reconciliation grace	How long the Manager waits after it starts for agents to reconnect before it decides the outcome of backups that were running when it stopped. A Manager restart does not stop a running backup, and a backup whose vault commit the storage agent is still running is left to complete.	5 minutes	1–1,440	Restart required

Test restores

These settings control scheduled test restores on an Isolated Recovery Host. They do nothing until a host is enrolled and ready. Test restores that an operator starts are not held back by them. See Test Restores.

Setting	Description	Default	Range	Flags
Prove new recovery images	Test-restores one backup with each recovery image that no restore has proven yet: the smallest full system backup from the last 30 days of a domain controller the image serves, starting from that image. Only the newest image of each operating system variant on each storage agent is tested, because that is the image a restore would start from.	On	On, Off	–
Test each DC's new backups within	How long a domain controller's backups may go untested. Before then its newest backup is test-restored, unless that one has already passed. The time runs from the domain controller's first backup taken after the newest one that passed; a backup that passed is never tested again. One domain controller at a time per host, the earliest deadline first, each started as late as still lets every domain controller waiting finish in time. Each test reads a whole full system backup from the vault, so the host's speed and the vault's load decide how many domain controllers a period can cover; with more than a host can reach, the tests start earlier and run back to back.	30 days	0–365 days	–

Setting	Description	Default	Range	Flags
	A domain controller whose deadline passes raises the **Domain controllers with no recent verified backup** alert. 0 turns off both the rotation and that alert.			
Keep clear of backups for	A scheduled test restore does not start while a backup runs, or when one is due within this long. 0 starts whenever no backup is running.	120 minutes	0–1,440 minutes	—
After a scheduled test restore fails, wait	How long before the same domain controller or recovery image is scheduled again after a scheduled test restore that did not pass. The schedule moves on to the next one meanwhile.	24 hours	1–720 hours	—

Database retention

These settings control how long the Manager keeps records in its own database. They do not affect backups in the vault; vault retention is set on the storage agents tab. For the day settings, 0 means keep forever.

Setting	Description	Default	Range	Flags
Job progress	How long the detailed progress of each backup job is kept. The job record itself is kept separately.	90 days	0–36,500	—
Agent sessions	How long the history of closed agent connections is kept. A live session is always kept.	365 days	0–36,500	—
Backup jobs	How long backup job records are kept. These	0 (keep forever)	0–36,500	Can delete

Setting	Description	Default	Range	Flags
	records are what reports are built from, and they cannot be rebuilt from the vault.			data
Manifests	How long backup manifests are kept. A manifest describes the domain controller at backup time and is what restore preflight checks against; a restore point whose manifest is deleted cannot be planned.	0 (keep forever)	0–36,500	Can delete data
Audit trail	How long audit entries are kept. Shortening it deletes the record that attestations are drawn from. Set a window only where your own data-protection rules require it.	0 (keep forever)	0–36,500	Can delete data
Sweep interval	How often the database retention sweep runs. 0 is not accepted.	24 hours	1–8,760	—

Agent deployment

Setting	Description	Default	Range	Flags
Package folder	A folder on the Manager machine that holds the agent packages to deploy and compare against: the DC agent, storage agent, and Isolated Recovery Host agent packages. The newest package of each kind is the version every agent of that kind is compared against; the newest DC agent package is also the one the console installs. The default is the `redist` folder that the Manager installs beside itself, so upgrading the Manager also updates the packages. Other MSI files in the folder are reported and ignored. An empty value turns console installs and update checks off.	`redist`	Folder path	—
Remote management transport	How the Manager starts the installer on a domain controller: `winrm` (TCP 5985) or `dcom` (TCP 135 and dynamic RPC). WinRM from the Manager, which is outside the domain, needs each target in the Manager machine's WinRM TrustedHosts list; DCOM needs the Windows Management	`winrm`	`winrm`, `dcom`	—

Setting	Description	Default	Range	Flags
	Instrumentation (WMI-In) firewall group enabled on the targets. A single deployment can use the other transport.			
Update agents automatically	Brings every agent up to the build in the package folder without an operator asking, as each one can take it. On every check, each DC agent, the storage machine, and each approved Isolated Recovery Host that is behind the package for its kind is updated as **Update** on the **Agents** screen does it: over its own connection, checked by the agent, with no credential. An agent that is not connected or is busy waits for a later check; one that cannot update itself is listed and never retried; a failed update is tried again after six hours, or at once when a newer package arrives. Off by default, because an update restarts the agent and some estates want to choose when. See Updating agents automatically.	Off	On, Off	–
Check for agents to update every	How often automatic updates look for outdated agents that can take an	30 minutes	5–1440 minutes	–

Setting	Description	Default	Range	Flags
	update now. An agent passed over is looked at again on the next check. Agents waiting for a free place among those updating at once do not wait for the interval: one starts within a minute of another finishing. Has no effect while automatic updates are off.			
Agents updated at once	How many agents automatic updates keep updating together; another starts as soon as one finishes. A higher number works through a large estate sooner, at the cost of more package traffic at once over the links to remote sites. Updates an operator starts are not counted against it. Lowering it lets those already running finish. Has no effect while automatic updates are off.	8	1-50	–

Email

The mail server the Manager sends every email through: recovery plan notifications, the recovery report, license expiry warnings, and alerts. Who receives each email is not set here; each recovery plan names its own notification addresses, license warnings go to the recipients in the License settings, and alerts to the recipients in the Alert notifications settings. One email goes to at most 25 recipients. A change takes effect with the next email. The emailed recovery report has its PDF attached, made by the Manager's own PDF renderer, which needs no settings.

Setting	Description	Default	Range	Flags
SMTP server	The mail server the Manager sends every email through. Empty: nothing is emailed.	Empty	Host name or IP address	–
SMTP port	The mail server's port: 587 for STARTTLS, 465 for TLS from the first byte, 25 for an unencrypted relay.	587	1-65,535	–
SMTP security	`starttls` upgrades the connection and refuses a server that will not upgrade; `tls` is encrypted from the start; `none` sends in the clear. The Manager's email names the forest's domain controllers and what is wrong with them, so use `none` only for a relay on the same host or on an isolated recovery network.	`starttls`	`starttls`, `tls`, `none`	–
SMTP user	The account the Manager signs in to the mail server as. Empty: no sign-in, for a relay that accepts the Manager.	Empty	Text	–
SMTP password	The SMTP user's password. It is write-only: once saved, it is never shown or returned again, and the console and `cdr config show` say only whether it is set. It is kept encrypted on the Manager machine with Windows DPAPI, as the Manager's CA key is, and it is never written to the audit trail or a log; the audit entry says only that it was set, changed, or cleared. It is never sent to an agent. To keep it, leave the field as it is. To replace it, enter a new one. To remove it, clear it. Changing **SMTP server** or **SMTP security** while a password is stored needs the password entered again, or cleared, in the same save; otherwise	Not set	Text	–

Setting	Description	Default	Range	Flags
	the save is refused. The stored password is never sent to a server other than the one it was entered for. A database moved to another machine cannot decrypt it. The Manager then treats it as not set and logs that it needs entering again.			
From address	The address the Manager's email comes from. Required to send any email.	Empty	Email address	–
Console address in emails	The console's address as emails link to it, such as `https://drmanager.corp.example.com:5210`. Empty: the Manager machine's DNS name and HTTPS port. Set it when recipients reach the console by another name: an alias, a DNS name in another domain, or a reverse proxy. Use a name the Manager's certificate carries, or the browser warns about the link.	Empty	An `https` address, with no query or fragment	–

License

Who is told by email as the license nears its end. A change takes effect with the next warning. For the license itself, see Licensing.

Setting	Description	Default	Range	Flags
Expiry warning recipients	Who is emailed as the license nears its end, separated by commas. Empty: warnings are not emailed. Sent 30, 15, 7, 5, 4, 3, 2, and 1 days before the license ends; when the grace period begins and on the same days before it is over; and on the day backups and recovery operations stop. A trial has no grace period. The console's alerts and banner say the same whether or not anybody is emailed, and every warning is in the Manager's log and the audit trail. Sent through the server in the **Email** settings, so set those up too.	Empty	Email addresses	–

Alert notifications

Who is emailed when an alert is raised or cleared, how often alerts are checked, and which severities are emailed. A change takes effect at once. For how alert email works, see Alerts Reference; the **Alerts** screen shows whether it is working (see Alerts).

Setting	Description	Default	Range	Flags
Recipients	Who is emailed when an alert is raised or cleared, separated by commas. Empty: alerts are not emailed. The Manager refuses a save with more than 25 addresses or with a value that is not an email address; use a distribution list for more. When recipients are first set, the first check reports everything raised at that moment. Clearing them stops alert email and forgets what was reported. Sent through the server in the **Email** settings, so set those up too. Each email sent, and each failure to send, is in the audit trail and the Manager's log.	Empty	Up to 25 email addresses	–
Check interval	How often alerts are checked for anything raised or cleared since the last check. An object counts as cleared once two checks in a row have not found it. The first check after the Manager starts waits five minutes, or this long if it is longer, so that agents have reconnected.	5 minutes	1-1,440 minutes	–
Reminder interval	How long after it was last emailed an alert still raised is reported again.	24 hours	0-720 hours	–

Setting	Description	Default	Range	Flags
	Any email that lists an object counts, so a reminder comes this long after the last email of any kind, not on a fixed schedule. A reminder lists everything still raised. 0 means an alert still raised is not reported again.			
Minimum severity	The least severe alert that is emailed, in the order the **Alerts** screen uses: `critical`, `warning`, `unknown` (a fact nothing could confirm, such as an agent that is not connected), `info`. Raising it stops reports about the alerts it leaves out without reporting them cleared; lowering it reports the alerts it brings in as raised. On the Configuration screen the choices read **Critical only**, **Warning and above**, **Could not find out, and above**, and **Every severity**.	`warning`	`critical`, `warning`, `unknown`, `info`	—

Configuration backups

How often the Manager sends an encrypted configuration backup to the storage machine, and how many the storage machine keeps. See Manager Configuration Backups.

Setting	Description	Default	Range	Flags
Back up this manager's configuration every	How often the Manager sends an encrypted copy of its database, enrollment token, and secret settings to the storage machine. Backups wait until the recovery passphrase is set. 0 means no scheduled backups; **Back up now** still works.	24 hours	0–168	—
Configuration backups to keep	How many configuration backups the storage machine keeps; the oldest is removed when a new one is stored. The storage machine keeps more if its own minimums say so. Keep enough to reach back past a change made in error or a compromise that goes unnoticed for a while.	14	1–365	—

Local to each machine

Setting	Description	Default	Range	Flags
Log folder	Where the Manager writes its service log. An installed Manager writes to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\logs`. An empty value turns the file log off.	Set by the installer	Folder path	Set on the machine, Restart required
Log retention (days)	How many days of service log files are kept before the oldest are deleted. 0 keeps every file.	30	0 or more	Set on the machine, Restart required

DC agent settings

These settings are sent to every DC agent as a fleet default. Agents apply a change when they receive it, and an agent that was offline catches up on its next connection.

Setting	Description	Default	Range	Flags
Heartbeat interval	How often each DC agent sends a heartbeat to the Manager. The Manager shows an agent as disconnected after it misses three heartbeats plus a minute: under two minutes at the default, about 16 minutes at 300 seconds. Raise it to reduce the packet rate from a large fleet or over a metered link. The heartbeat is also what keeps an idle connection open through firewalls, so do not raise it beyond what your firewalls keep open.	15 seconds	15–300	—
Topology report interval	How often each DC agent reports what it can see of the directory: forests, domains, sites, and domain controllers. The Manager treats topology as stale when no report has arrived for twice this interval.	12 hours	1–8,760	—
Job transcripts kept	How many wbadmin transcripts each DC agent keeps on the domain controller. Transcripts are also sent to the Manager, so this	50	1–10,000	—

Setting	Description	Default	Range	Flags
	only limits what is kept on the DC's disk.			
Unreported result max age	How far back an agent looks, when it reconnects, for finished jobs whose outcome the Manager was never told.	48 hours	1-8,760	–
Install the Windows Server Backup feature	Lets the agent install the Windows Server Backup feature itself if wbadmin is missing; without it every backup of the DC fails. The agent always checks for the feature and reports whether it is there. Turn this off, for the fleet or for one DC, where adding a Windows feature to a domain controller must go through change control; that DC then reports the feature missing, and its backups fail until it is installed by hand.	On	On, Off	Restart required
Manager address	The address the agent dials, set at installation with `MANAGERADDRESS`.	Set by the installer	`https://` address	Set on the machine, Restart required
Manager CA fingerprint	The SHA-256 fingerprint of the Manager CA that the agent trusts, set at installation with `MANAGERCAFINGERPRINT`. Public, not secret.	Set by the installer	Fingerprint	Set on the machine, Restart required
Job log	Where wbadmin	Set by the	Folder path	Set on the

Setting	Description	Default	Range	Flags
folder	transcripts are written on the DC. An installed agent uses `%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent\job-logs`.	installer		machine, Restart required
Restore staging volume	The volume a DC state recovery copies the backup onto before the DC restarts into Directory Services Restore Mode. Empty picks the fixed volume, other than the system volume, with the most free space. The system volume is never used.	Empty	Volume	Set on the machine
Log folder	Where the DC agent writes its service log. An installed agent writes to `%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent\logs`. An empty value turns the file log off.	Set by the installer	Folder path	Set on the machine, Restart required
Log retention (days)	How many days of service log files are kept. 0 keeps every file.	30	0 or more	Set on the machine, Restart required

Storage agent settings

These settings are sent to the storage agents, which cache them and keep running on the last values they received if the Manager is unreachable.

Staging

Setting	Description	Default	Range	Flags
Staging share override (UNC)	Leave empty to have the agent find the share on its own machine that publishes the staging folder and advertise it by IP address. Set it only when staging is reached some other way. Point it at a share of the staging folder only. Do not share the storage root: a share over the storage root also exposes the vault.	Empty	UNC path	Restart required
Advertised host	Overrides the address domain controllers are given for staging. Required on a storage machine with more than one network address: the agent cannot tell which one the DCs reach, so it advertises its machine name instead, which makes backups depend on DNS name resolution. Set it to the address domain controllers should use. The same address is what a full system recovery's recovery environment dials to fetch the recovery image and read the vault, so an IP address needs no DNS during a recovery. A	Empty	IP address or name	Restart required

Setting	Description	Default	Range	Flags
	host name is also accepted, but then every full system recovery depends on DNS resolving it, and restore preflight says so.			
Manage the staging share's permissions	Lets the agent grant the staging writers group Change permission on the staging share if it is missing. Turned off, the agent still checks the share and reports what is wrong. It never removes or narrows an existing entry.	On	On, Off	Restart required
Credential lifetime	The expiry stamped on the per-job credential a domain controller uses to write its backup. The account is deleted as soon as the backup is sealed, which is normally much sooner.	24 hours	1–8,760	—

Vault

Setting	Description	Default	Range	Flags
Capacity measurement interval	How often the agent measures its vault and reports capacity to the Manager. Raise it on a very large vault, because the first measurement after a restart reads the whole vault.	5 minutes	1-10,080	–
Remove BUILTIN\Administrators from the vault	Leaves only SYSTEM and the `CygnaDR Administrators` group on the vault and staging permissions. A local administrator can still take ownership, so this is evidence rather than a barrier: reading the backups then needs a deliberate ownership change. The agent refuses to apply it while the `CygnaDR Administrators` group is empty.	Off	On, Off	Restart required
Size estimate margin	Padding added to the Manager's	20%	0-500	–

Setting	Description	Default	Range	Flags
	size estimate before vault space is reserved for an incoming backup.			
Reservation timeout	Releases vault space held for a backup that failed without committing. Must be at least 30 minutes longer than the Manager's **Backup job timeout**.	270 minutes	1-10,080	–

Setting	Description	Default	Range	Flags
Abandoned staging window	How long a staging folder may sit with no writes before it is treated as a job's leftovers. This is what protects a backup that is still running.	4 hours	1-8,760	—
Delete abandoned staging folders	On, the sweep deletes idle staging folders to reclaim space. A backup that finished but was never committed is committed by the Manager from staging within the hour, or as soon as the storage agent reconnects, well before the sweep would remove it. Off, the folders are reported and kept, and the space is not reclaimed.	On	On, Off	Can delete data, Cannot weaken the machine's floor

Vault retention

For how these settings work together, see Retention.

Setting	Description	Default	Range	Flags
Retention floor	No version inside this window is deleted to make room for an incoming backup; the backup is refused instead. Applies to space pressure only. 0 means no version is protected by age, which the agent warns about at startup.	180 days	0-36,500	Set on the machine, Cannot weaken the machine's floor
Minimum versions per DC	No rule, for any reason, takes a domain controller below this many versions.	1	1-10,000	Set on the machine, Cannot weaken the machine's floor
Maximum version age	Versions older than this are deleted. A value from the Manager cannot go inside the storage machine's retention floor. 0 means keep forever.	0 (keep forever)	0-36,500	Can delete data, Cannot weaken the machine's floor
Maximum versions per DC	The most versions kept for each domain controller, however recent they are. Which versions are kept is decided by **Retention mode**. 0 means unlimited.	2	0-10,000	Can delete data, Cannot weaken the machine's floor
Retention mode	Which versions the	`newest`	`newest`, `thinned`	Can delete data

Setting	Description	Default	Range	Flags
	count ceiling and a full disk delete. `newest` keeps the most recent versions and, under space pressure, deletes only versions older than the floor. `thinned` keeps each DC's newest version and its deepest inside the floor and deletes from between them, and under space pressure can delete versions inside the floor, from the middle only.			
Free space reserve	Free space kept on the vault volume beyond what an incoming backup needs. When free space falls below the reserve, the vault refuses every backup. Fractional values are accepted.	10 GB	0–1,048,576	—

Setting	Description	Default	Range	Flags
Sweep interval	How often the vault retention sweep runs. 0 is not accepted.	1 hour	1-8,760	–
Dry run	Selects and reports what retention would delete without deleting anything. Use it to preview a new policy. A storage machine that sets dry run locally cannot have it turned off from the console.	Off	On, Off	Cannot weaken the machine's floor

Note: The retention floor and the minimum versions per DC are set in the storage agent's `appsettings.json` on the storage machine, and cannot be set from the Manager or from an installer property. The console can tighten retention but never weaken it.

Configuration backups

The storage machine's own limits on the Manager's configuration backups. The Manager says how many to keep; the storage machine keeps at least its own minimum and decides what is deleted, so that a compromised Manager cannot remove the history. See Manager Configuration Backups.

Setting	Description	Default	Range	Flags
Configuration backups always kept	The storage machine keeps at least this many configuration backups, whatever the Manager asks.	7	1-1,000	Set on the machine, Cannot weaken the machine's floor
Configuration backups never pruned before	No configuration backup younger than this is deleted,	7 days	0-3,650	Set on the machine, Cannot

Setting	Description	Default	Range	Flags
	however many there are. When the folder fills instead, the folder limit refuses the next backup. 0 means backups beyond the count are deleted at any age.			weaken the machine's floor
Configuration backups folder limit	A configuration backup that would take the `manager-backups` folder past this size is refused. Every backup also leaves the vault's **Free space reserve** free.	100 GB	1-100,000	Set on the machine

Set these in the `CygnaDR` section of the storage agent's `appsettings.json` on the storage machine, as `CygnaDR:ConfigBackup:MinKeep`, `CygnaDR:ConfigBackup:MinAgeDays`, and `CygnaDR:ConfigBackup:MaxFolderGb`. Restart the storage agent after changing them.

Vault verification

For what verification checks, see Vault Protection and Integrity.

Setting	Description	Default	Range	Flags
Verify committed backups	Re-reads committed backups and checks them against the hash recorded when they were committed. The check only reads: a failure is reported and the version is left where it is.	On	On, Off	–
Verification interval	How often a verification pass runs. 0 is not accepted; to stop verification, turn **Verify committed backups** off.	24 hours	1-8,760	–
Versions per	How many versions one pass	5	1-10,000	–

Setting	Description	Default	Range	Flags
pass	reads before it waits for the next interval. The least recently verified versions go first. A large vault needs a shorter interval or more versions per pass; the **Repository** screen shows how far behind verification is.			
Re-verify after	How old a passing result may be before the version is read again. A version that failed is read again on every pass.	30 days	1-3,650	–

OS media library

Setting	Description	Default	Range	Flags
Media folder	A folder of Windows Server ISO files that you supply, as a local path or a UNC path. Empty turns the OS media library off. Nothing in the folder is ever written, moved, or deleted.	Empty	Folder path	Restart required
Scan interval	How often the agent scans the whole media folder.	12 hours	1-8,760	–
Watch the folder for changes	Rescans when the folder changes, in addition to the scheduled scan.	On	On, Off	Restart required
Change debounce	How long changes must settle before a watched rescan runs, so copying a large ISO produces one scan.	60 seconds	1-86,400	–

Setting	Description	Default	Range	Flags
Compute content hashes	Hashes every ISO, so a file replaced under the same name is detected. This reads every byte of every ISO on the storage machine; size and modification time catch most changes without it.	Off	On, Off	–
Include subfolders	Scans folders below the media folder as well.	On	On, Off	Restart required
Maximum files per scan	The most files one scan reads. A scan that reaches the limit is reported as partial.	200	1-100,000	–

Recovery images

For setting up recovery images, see Setting Up OS Media and Recovery Images.

Setting	Description	Default	Range	Flags
Recovery image folder	Where recovery images are built and served from. While it is empty, no images are built and no domain controller can use full system recovery; DC state recovery is unaffected. Use a different folder from the media folder.	Empty	Folder path	Restart required
Build working folder	Scratch space for image builds, roughly the size of an image. The build log is written here too. Empty uses a folder beside the image folder.	Empty	Folder path	Restart required
Recovery agent folder	Overrides the recovery agent placed inside each image. Leave empty to use the one installed with this storage	Empty	Folder path	Set on the machine,

Setting	Description	Default	Range	Flags
	agent, which is the normal case. Changing it makes every existing image out of date. Set on the storage machine (`CygnaDR:RecoveryImages:AgentFolder`), never from the Manager: the storage agent runs the program it finds here as SYSTEM, and every recovery image then starts it on each restored domain controller.			Restart required
Driver folder	Drivers to add to every image, such as storage controller or network drivers for physical hardware. Drivers are added even if they are unsigned. Empty adds none. Set on the storage machine (`CygnaDR:RecoveryImages:DriverFolder`), never from the Manager, because a driver in the image runs in the kernel of every machine the image restores.	Empty	Folder path	Set on the machine, Restart required
Build check interval	How often the agent checks whether any required recovery image is missing or out of date, and builds it if so.	24 hours	1–8,760	–

Local to each machine

Setting	Description	Default	Range	Flags
Storage root	The folder that holds staging and the vault, set at installation with `STORAGEROOT`. It cannot be set from the Manager.	Set by the installer	Absolute path	Set on the machine, Restart required
Manager address	The address the agent dials, set at installation with `MANAGERADDRESS`.	Set by the installer	`https://` address	Set on the machine, Restart required

Setting	Description	Default	Range	Flags
Manager CA fingerprint	The SHA-256 fingerprint of the Manager CA that the agent trusts, set at installation with `MANAGERCAFINGERPRINT`.	Set by the installer	Fingerprint	Set on the machine, Restart required
Log folder	Where the storage agent writes its service log. An installed agent writes to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent\logs`. An empty value turns the file log off.	Set by the installer	Folder path	Set on the machine, Restart required
Log retention (days)	How many days of service log files are kept. 0 keeps every file.	30	0 or more	Set on the machine, Restart required

Isolated Recovery Host agent settings

These settings are sent to every Isolated Recovery Host agent, on the **Isolated Recovery Host agents** tab. An installation without a host has nothing to send them to. Each applies as soon as the agent receives it. For what the agent does with them, see Setting Up an Isolated Recovery Host.

Readiness

Setting	Description	Default	Range	Flags
Readiness check interval	How often the host checks and reports whether it can run test restores: Hyper-V, its switches, disk, and memory. The host also checks at every connection.	15 minutes	1-1,440 minutes	–

Windows features

Setting	Description	Default	Range	Flags
Install Hyper-V	Has the agent install Hyper-V and its PowerShell module when the host lacks them. Tried when the agent starts and when this is turned on, once per run of the service. It cannot succeed where the processor offers no virtualization, which on a virtual machine means nested virtualization.	On	On, Off	–
Restart to finish installing	Lets the agent restart the host to finish a Hyper-V install, once per install, five minutes after asking Windows, so `shutdown /a` can cancel it. Never when the storage agent, the Manager, or a domain controller is on the same machine; the agent then reports that a restart is needed.	On	On, Off	–

Verification VMs

Setting	Description	Default	Range	Flags
Folder for verification VMs	Where test restore virtual machines and their restored disks go. Empty, the agent chooses a `CygnaDR\Recovery VMs` folder on the data drive with the most free space. Set, give a full local path, such as `D:\CygnaDR\VMs`; where hosts have different drives, list a folder for each, separated by `;`, and each host takes the first on a drive it has. A host can also be given its own folder as an override on that machine; an override belongs to the host's agent, so a host that replaces it needs its own. Only NTFS and ReFS drives can hold virtual disks.	Empty	Local folder paths	–
Create the recovery network	Has the agent create an internal Hyper-V switch named **CygnaDR Recovery**, give the host the first address of the recovery network prefix on it, and put Windows	On	On, Off	–

Setting	Description	Default	Range	Flags
	NAT behind it. Test restore virtual machines restore over it. The agent never creates an external switch, but it accepts one an operator made.			
Recovery network prefix	The private network behind the recovery switch's NAT. Choose one used nowhere the host or the virtual machines need to reach. Each test restore gives its virtual machine a static address in it. A change re-creates the switch's address and NAT on the next check, never during a test restore.	`172.31.250.0/24`	A private IPv4 network, /16 to /29	—

Local to each machine

Setting	Description	Default	Range	Flags
Allow updates from the manager	Whether this machine's Isolated Recovery Host agent accepts a newer build of itself from the Manager. Set in the machine's update policy, never from the console. See Update policy.	On	On, Off	Set on the machine, Restart required
Manager address	The address the agent dials, set at installation	Set by the installer	`https://` address	Set on the machine,

Setting	Description	Default	Range	Flags
	with `MANAGERADDRESS`.			Restart required
Manager CA fingerprint	The SHA-256 fingerprint of the Manager CA that the agent trusts, set at installation with `MANAGERCAFINGERPRINT`.	Set by the installer	Fingerprint	Set on the machine, Restart required
State file path	Where the agent keeps its state, including its cached copy of this configuration.	`data/host-agent-state.json`	File path	Set on the machine, Restart required
Log folder	Where the agent writes its service log. An installed agent writes to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Isolated Recovery Host Agent\logs`. An empty value turns the file log off.	Set by the installer	Folder path	Set on the machine, Restart required
Log retention (days)	How many days of service log files are kept. 0 keeps every file.	30	0 or more	Set on the machine, Restart required

Licensing

Cygna Forest Recovery is licensed with a Cygna license key, the same kind of key that Cygna issues for its other products. The key names your organization, the license type (subscription, trial, or perpetual), the number of people it licenses, and its end date. A key licenses modules. Cygna Forest Recovery has one, Cygna Forest Recovery for Active Directory, and the type, number of people, and dates are that module's. A single key can license other Cygna products as well; Cygna Forest Recovery reads only its own modules and ignores the rest. A key that does not license Cygna Forest Recovery for Active Directory, such as a Cygna Auditor key, is refused.

The Manager keeps the key in its database, so the database snapshots carry it and a Manager recovered from a snapshot keeps its license.

What is enforced

- **The number of people**—Cygna licenses physical people, and one person often holds several accounts, such as a separate administrator account or a test account. The License page shows the enabled user accounts in every forest beside the licensed count, to help you check your coverage.
- **The end date is enforced**—The Manager warns you as the end date approaches. After the end date, a subscription has a 14-day grace period in which everything keeps working; a trial has none. Then the Manager refuses to start new backups and recovery operations until a current key is applied. A perpetual license does not end.
- **No key is the same as an expired one**—A new Manager has no key, so it refuses backups until you apply one. Apply the key right after you sign in for the first time.

Applying a license key

Applying a key replaces the installed one, if there is one. A key takes effect at once: if backups and recovery operations were stopped, they resume, and scheduled backups that came due while they were stopped start on the scheduler's next pass.

Before you begin

- You need the `config.manage` capability, which the System Admin role and the break-glass account hold. See Managing Operators and Roles.
- Have the license key that Cygna sent you.

To apply a license key in the console:

1. Select **Administration** > **License**.
2. Select **Apply a license key…**, or **Apply a new license key…** when a key is already installed.

   The **Apply a license key** dialog opens.
3. In **License key**, paste the key exactly as Cygna sent it. Line breaks from an email are fine.

   The Manager reads the key and shows what it says, without installing it: the license's standing and its terms. If the key cannot be used, the dialog shows **Refused** and the reason.

4. Check the licensee, type, number of people, and end date, and then select **Apply this license**.

   The key is installed, the License page shows the new standing, and the audit trail records **License installed**.

To apply a key with the command-line tool, run `cdr license apply <KEY>`. To see what a key says first, run `cdr license preview <KEY>`. See Configuration and Exclusion Commands.

The Manager refuses a key in these cases:

- **It is not a key the Manager can read**—The message is **This is not a Cygna version 2 license key.** Paste the whole key again, exactly as sent. If it is still refused, ask Cygna Labs for a new Cygna Forest Recovery key.
- **It does not license this product**—The message is **This key does not license Cygna Forest Recovery for Active Directory. It may be a Cygna Auditor key: ask Cygna for a key that licenses it.**

The License page

The **License** screen, in the **Administration** group, shows the license and the product version installed on this Manager, the enabled user accounts beside the licensed count, a support package for a support case, and, at the foot of the page, whether this Manager can be recovered. Every operator can open it. At the top, a status chip and one sentence give the standing, for example **Licensed for 250 people until 31 March 2027.** When backups and recovery operations are stopped, the page also says what is refused and what keeps working.

Status	Meaning
Licensed	The license is current and ends more than 30 days from now, or does not end.
Ending soon	The license ends within 30 days. Everything works.
Grace period	The end date has passed, and the 14-day grace period is running. Everything works until the date shown under **Operations stop**.
Expired	The grace period is over, or a trial has ended. New backups and recovery operations are refused.

Status	Meaning
Module not licensed	The installed key does not license Cygna Forest Recovery for Active Directory. New backups and recovery operations are refused.
No license	No key is installed. New backups and recovery operations are refused.

The **License** card shows when the key was installed and by whom, and the facts that apply to the whole key:

- **Licensed to**—The organization named in the key.
- **Issued**—The day the key was issued.

Under **Licensed modules**, each module the key licenses has its own block, headed by the module's name, Cygna Forest Recovery for Active Directory, with its terms:

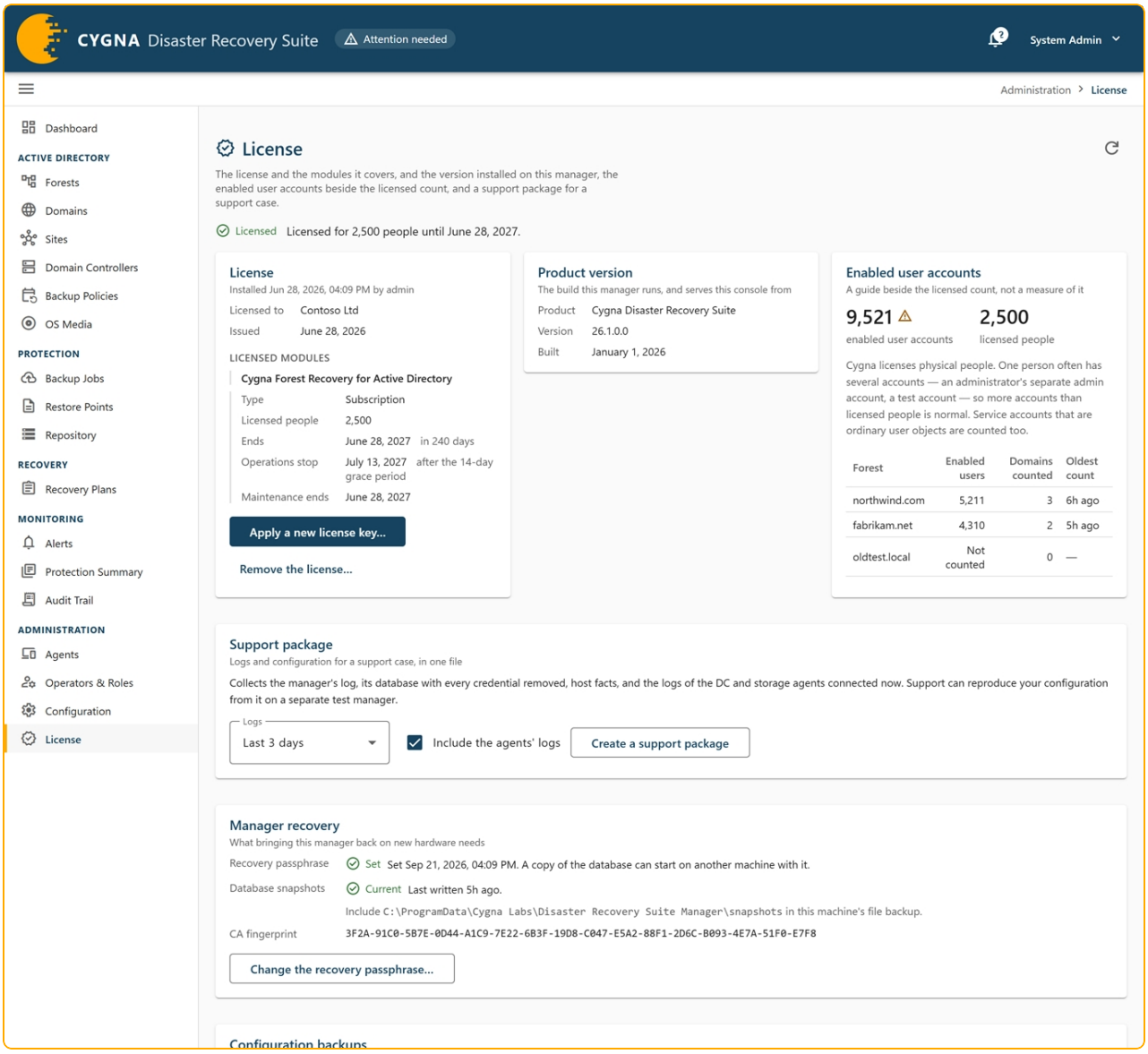
- **Type**—**Subscription**, **Trial**, or **Perpetual**.
- **Licensed people**—The number of people the key licenses.
- **Ends**—The last licensed day, and how far away it is. A perpetual license shows **Never (perpetual)**.
- **Operations stop**—The first day new backups and recovery operations are refused: the day after the 14-day grace period, or the day after a trial ends.
- **Maintenance ends**—The end of the maintenance period named in the key, or **Not stated**.

Operators with the `config.manage` capability see the buttons to apply a key and to remove the license. Other operators see **Applying a license key needs the Configuration capability.**

The **Product version** card shows the build this Manager runs and serves the console from: **Product**, **Version**, and **Built**, the day the build was made. Give the version when you contact Cygna Labs Corp. support. The card is shown even when the license cannot be read.

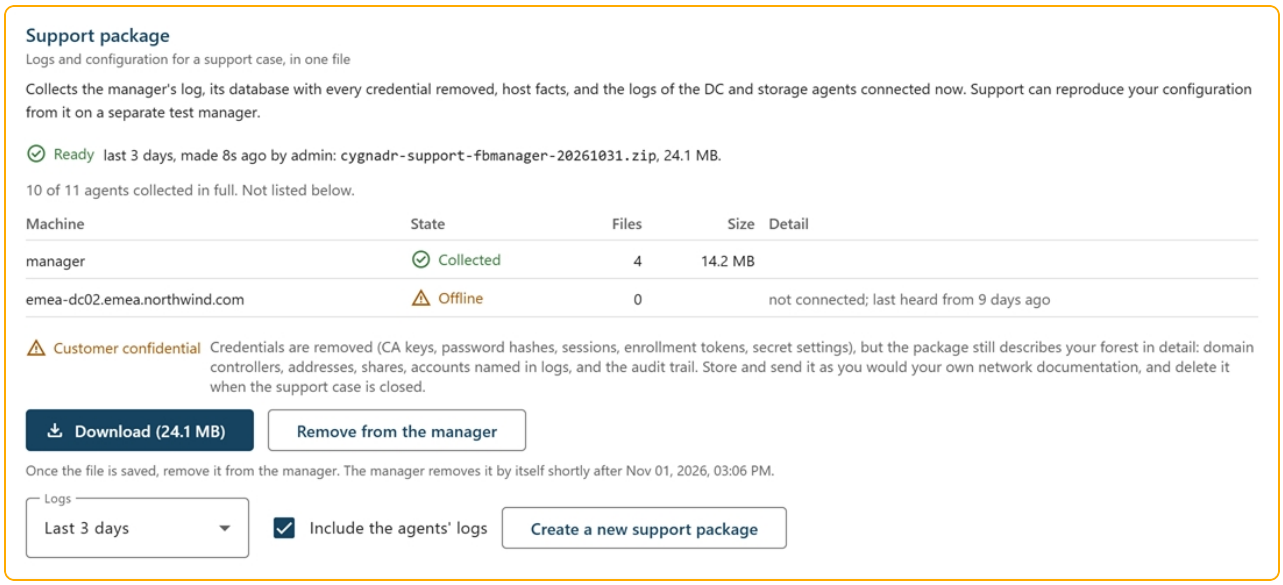
The **Enabled user accounts** card shows the number of enabled user accounts across every forest beside the number of licensed people, and a table with one row per forest: **Forest**, **Enabled users**, **Domains counted**, and **Oldest count**. The figures come from the DC agents, which count the user accounts in their domain. A domain with no DC agent is not counted, so its forest's **Domains counted** shows, for example, **2 of 3**, and the total reads "at least". Domains excluded from alerts are still counted, and service accounts that are ordinary user objects are counted too. More accounts than licensed people is normal, so the page never calls the difference a breach. When the enabled accounts are more than

a fifth above the licensed count, a warning icon appears beside the figure as a prompt to look.



Support package

Below the license cards, the **Support package** card collects what Cygna Labs Corp. support needs for a case into one zip file: the Manager's log, a copy of its database with every credential removed, host facts, and the logs of the agents connected now. The card is shown only to operators with `config.export`, and is shown even when the license cannot be read. For what a package contains and leaves out, see Logs and Diagnostics.



To create a support package:

1. Select **Administration** > **License**.
2. In **Logs**, choose how far back to collect logs: **Last day**, **Last 3 days** (the default), **Last 7 days**, **Last 14 days**, **Last 30 days**, or **Everything retained**.
3. To collect the Manager's part only, clear **Include the agents' logs**. It is selected by default.
4. Select **Create a support package**, or **Create a new support package** if the card already shows one.

   The card shows **Collecting**, with who started it, when, and how many machines have finished.
5. When the card shows **Ready**, select **Download (*size*)** and save the file.
6. Select **Remove from the manager**.

Each agent sends at most 512 MB, newest files first. To choose particular agents or a different limit, use `cdr support package` (see Support Commands).

Machines table

While a package is collected and once it is ready, a table lists each machine with its **State**, **Files**, **Size**, and **Detail**. To keep the table short on a large estate, it lists the Manager, machines still being collected, and machines with a gap; agents collected in full and agents waiting their turn are counted in the caption above it.

State	Meaning
Waiting	Not started yet. The Manager collects from up to four agents at a time.
Collecting	The machine is sending its files.
Collected	Every file in the period was collected.
Partial	Something was left out, because it was over the agent's limit or could not be read. The detail says what.
Offline	The agent is not connected, so nothing was collected. The detail says when it was last heard from. Collect its log folder by hand if support needs it.
Failed	Collection from this machine failed. The detail gives the reason.

Ready, failed, and refused packages

- **Ready**—The card names the file and its size, and shows a **Customer confidential** warning: credentials are removed, but the package still describes your forest in detail. **Download (*size*)** saves the file. Once you have saved it, select **Remove from the manager**; otherwise the Manager removes it 24 hours after it finished, and the card says when.
- **Failed**—The package could not be built, with the reason.
- **Refused**—The Manager did not start the package, with its message: for example, another package is still being collected, or the Manager's data disk has less than 2 GB free.

Under **Also on the manager**, the card lists every other package the Manager still holds, including packages started with `cdr`, each with **Download** and **Remove**. Every package is also deleted when the Manager service starts.

Manager recovery and configuration backups

At the foot of the page, below the **Support package** card, two cards show whether this Manager could be brought back on new hardware. They are shown only to operators with `config.manage`, and are shown even when the license cannot be read. Setting the recovery passphrase and starting a configuration backup also require `config.export`, held by the System Admin role.

The **Manager recovery** card shows:

- **Recovery passphrase**—**Set**, with the date, or **Not set**. Without it, a copy of the Manager's database cannot start the Manager on another machine.

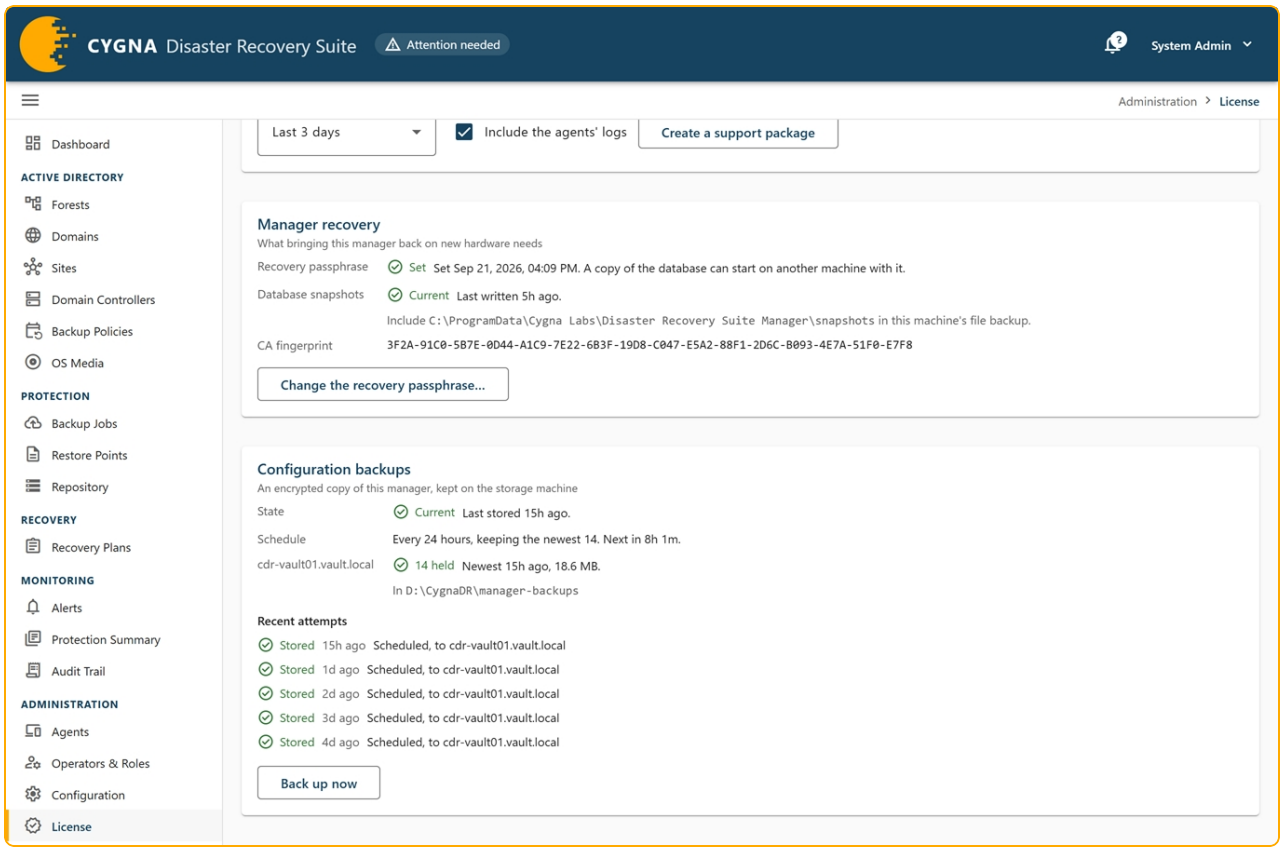
- **Database snapshots**—**Current**, **Stale** (the last snapshot is more than 2 days old), or **None yet**, with the folder to include in the Manager machine's file backup.
- **CA fingerprint**—The fingerprint of the Manager's certificate authority.

With `config.export`, select **Set the recovery passphrase…** or **Change the recovery passphrase…**, enter the **New passphrase** (at least 16 characters) and **Repeat it**, and select **Save the passphrase**. Copies made from then on need the new passphrase; database snapshots and configuration backups written before still open with the previous one. The new passphrase is never stored or shown again, so replace the offline record immediately. Saving it also starts a configuration backup.

Below it, the **Configuration backups** card shows whether the Manager is sending its encrypted configuration backups to the storage machine:

- **State**—**Current**, **Stale** (the newest backup is more than twice the interval old), **None yet**, **Running**, **Last attempt failed**, or **Not running** with the reason.
- **Schedule**—How often a backup is sent, how many the storage machine keeps, and when the next is due.
- **Each storage machine**—How many backups it holds, the newest one's age and size, and the folder they are in.
- **Earlier passphrase**—Shown when backups written under a previous recovery passphrase are still held.
- **Recent attempts**—The last five, with who started each, the storage machine, and why any did not store.

With `config.export`, select **Back up now** to send one at once. For why these matter and how to recover the Manager, see Protecting the Manager, Manager Configuration Backups, and Recovering the Manager.



Warnings before the end date

The Manager checks the license every hour. As the end date approaches, it warns on every surface:

- **License page**–The status changes to **Ending soon** 30 days before the end date.
- **Banner**–From 30 days before the end date, a banner appears on every screen of the console, below the recovery banner, except the License page. It reads **License ending soon**, **License ended** during the grace period, or **Backups and recovery operations are stopped**, followed by the license's sentence and a **View license** link. The banner also appears whenever no usable key is installed.
- **Alert**–The **License ending soon** warning alert, then the **License ended: grace period running** critical alert, then **Backups and recovery operations stopped: no active license**. Only the one that applies is raised. See Alerts Reference.
- **Log and email**–On each warning day, the Manager writes a warning to its log and emails the addresses in the **Expiry warning recipients** setting, through the mail server in the **Email** settings. The audit trail records **License expiry warning emailed**, or **License expiry warning not emailed** with the reason, for example when no recipients are set.

The warning days are 30, 15, 7, 5, 4, 3, 2, and 1 days before the end date; the day the grace period begins, and 7, 5, 4, 3, 2, and 1 days before it is over; and the day backups and recovery operations stop. A trial has no grace period: its warnings run up to the end date, followed by the stop warning the day after.

To receive the emails, set **Expiry warning recipients** to one or more addresses separated by commas, and set up the **Email** settings. See Settings Reference. The alert and the banner appear whether or not anyone is emailed.

When the license stops operations

After the grace period, or the day after a trial ends, and whenever no usable key is installed, the Manager refuses to start:

- New backups, scheduled or started with **Back up now** or `cdr backup start`. A refused backup leaves no failed job; scheduled backups stay due and start once a current key is applied.
- Restore runs. An approved run stays approved, ready to execute once a current key is applied.
- Recovery plans.
- Test restores, scheduled or started by an operator.

The refusal carries the code `license-inactive`, shown in the console as **No active license**, and gives the license's sentence as the reason, for example **No license is installed, so backups and recovery operations are stopped. Apply a license key.** The `cdr` commands exit with code 3 and add a hint to run `cdr license show` and `cdr license apply`.

The license never interrupts recovery work. These keep working:

- Work already under way finishes, including a recovery plan that is already executing and every restore it starts, even one that waits for an operator to execute it.
- Planning and approving restores and recovery plans.
- Every screen of the console and every `cdr` command that only reads.
- Cleanup of test restore virtual machines.



Note: **Important:** Without a current license, your restore points age and no new ones are taken. Renew before the grace period ends, so that you always have a recent backup to recover from.

Removing the license

Removing the license stops new backups and recovery operations at once, exactly as an expired license does. To replace a key, apply the new one instead; you do not need to

remove the old one first.

To remove the license:

1. Select **Administration** > **License**.
2. Select **Remove the license…**.
3. Read the consequences, and then select **Remove the license**.

   The audit trail records **License removed**.

With the command-line tool, run `cdr license remove`.

Maintaining Agents

Once agents are enrolled, most of their upkeep is automatic: they reconnect after an outage, renew their own certificates, and enroll again if the Manager stops recognizing them. The tasks in this topic are the ones that need you: updating agents to a new build, or having the Manager do it automatically, restarting them so saved settings come into force, pointing them at a Manager whose address or CA has changed, and removing an agent from a machine.

The **Agents** screen shows every enrolled DC agent, storage agent, and Isolated Recovery Host agent, with its version, last heartbeat, and connection state. See Agents.

Updating agents

The Manager compares every agent with the package for its own kind in its package folder. By default that is the `redist` folder in the Manager's program folder. It holds the DC agent, storage agent, and Isolated Recovery Host agent packages of the Manager's own build, so upgrading the Manager is what makes newer agents available. The **Agents** screen marks every agent that is behind its kind's package, and the ***n* updates available** chip counts them. To deploy a build other than the one the Manager shipped with, change the **Package folder** setting; see Settings Reference.

The Manager decides for each agent how it is updated:

- **Self-update**—An agent that is connected and can update itself is sent the package over its own connection to the Manager, checks it, and installs it. No credential is needed, and nothing connects into the machine. This applies to agents of every kind.
- **Push install**—A DC agent that cannot update itself, and a domain controller that has no agent, are updated by push installation, with the same prerequisites as a first deployment: an account that is a local administrator on the domain controller, the `ADMIN$` share, the Remote Registry service, and either WinRM or DCOM. See Deploying DC Agents.

- **Hand install**–A storage agent or Isolated Recovery Host agent that cannot update itself is never updated from the Manager. Install its package on the machine; see Updating an agent by hand.

An agent cannot update itself when it is not connected or when its machine's policy switches updates from the Manager off. See Switching off updates from the Manager.

What the agent checks before it installs

A self-updating agent installs a package only when all of these are true:

- Its SHA-256 hash matches the one the Manager offered.
- It carries a valid, timestamped Authenticode signature from Cygna Labs Corp.
- It is the package for this agent's own kind.
- Its version is newer than the build the agent runs. An agent never updates itself to an older build, and never reinstalls the build it already runs.
- The agent's data folder has room for the package plus 256 MB.

The agent then runs the installer from a one-time scheduled task as SYSTEM. Its service stops, and the new build reconnects and reports back. If the installer fails, Windows Installer rolls the change back, and the old build keeps running and reports the failure. The installer log is in the agent's data folder, in the `update` subfolder; a summary of it comes back in the update's transcript.

Work in flight

The Manager does not start an update on a machine that is busy. The target's result is **Refused before starting**, with the reason:

- A domain controller while a backup of it is running, a restore involving it is in progress, or a recovery plan is active in its forest.
- The storage agent while backups are running, a restore run is reading from the vault, or a recovery plan is active.
- An Isolated Recovery Host agent while a test restore is running on its host.

The storage agent also checks for itself, and declines the package while it builds a recovery image or reads a backup to hash it for a commit or a restore. It holds that work back from its check until the installer runs, and builds any recovery image it held back once it is running again, on the new build or the old one. Update the agent when that work has finished.

Agents that share a machine, such as a storage agent and an Isolated Recovery Host agent on one server, are updated one after another, never at once: Windows runs one installation at a time on a machine, and a second started beside it would fail. This holds across separate update runs, including automatic ones.

From the console

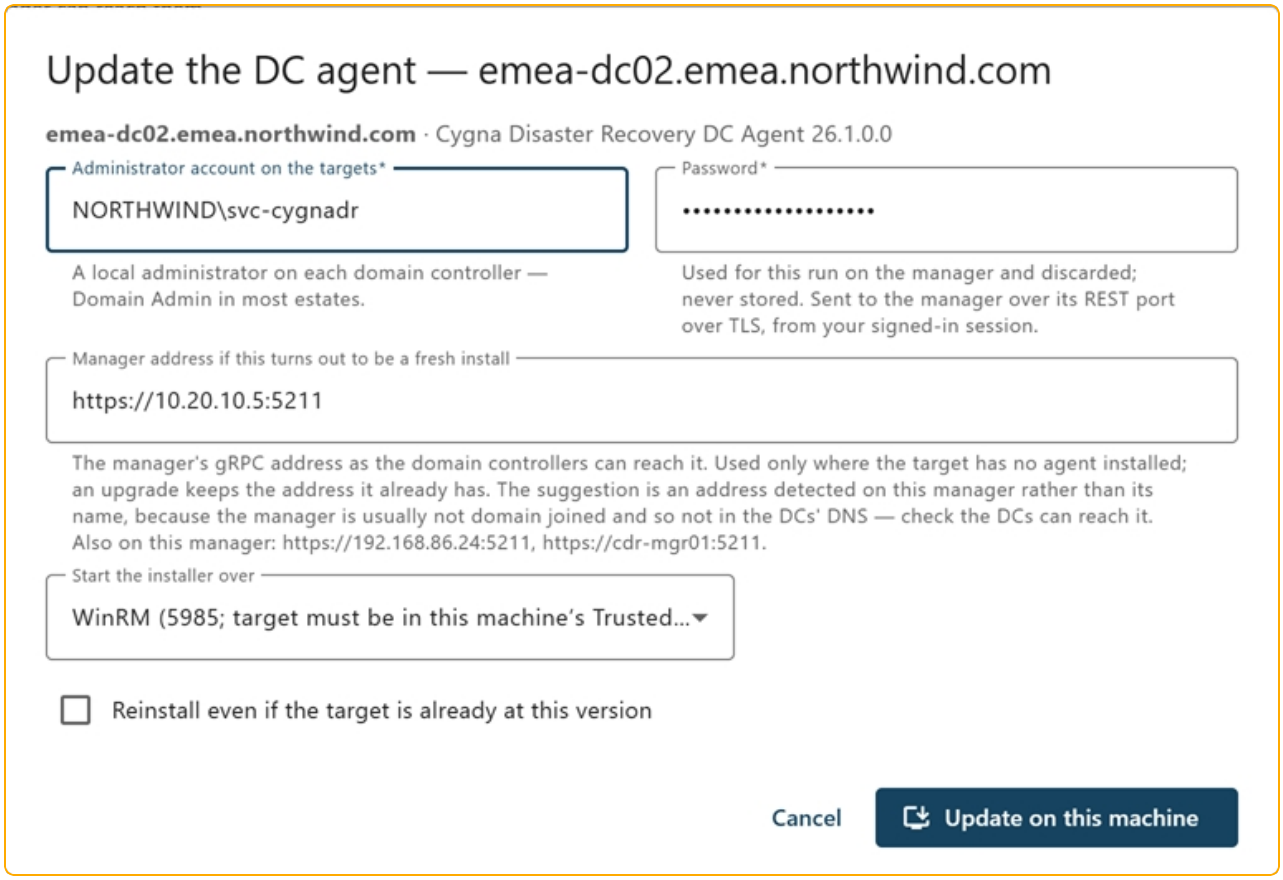
To update agents from the console:

1. Select **Administration** > **Agents**.
2. Choose the agents in one of these ways:
   - To update every outdated agent, select **Update *n* outdated agents…** beside the package line on the **Updates** tab. Agents that must be installed by hand are left out.
   - To update one agent, open its row menu and select **Update the agent on *machine* to *version*…**.
   - To update a selection, select the rows' checkboxes, and then select **Update *n* agents…** (**Update 1 agent…** for one). A row has a checkbox when its agent can update or restart itself, or when it is a domain controller; the button counts only the selected agents that can be updated.
3. If any target needs a push install, enter the administrator account, as `DOMAIN\user` or `user@domain`, and its password.

   The password is used for this run only and is not stored. When every target updates itself, the dialog asks for nothing.
4. Start the update and watch each target's result.

   Each self-updating agent receives the package, checks it, installs it, and reconnects on the new version. A push install copies the package to the domain controller, runs the installer, and checks that the agent service restarted on the new version. If you close the dialog, the update continues on the Manager.

Starting an update needs the `agents.manage` capability. For a push install over WinRM, each target must be in the Manager machine's WinRM TrustedHosts list; otherwise set **Remote management transport** to `dcom`.



From the command line

`cdr agent update` brings agents up to the build the Manager holds: by self-update where they can, and by push install for a domain controller that cannot. Name the agents, by ID or by machine name, or select every outdated agent with `--outdated`:

```
cdr agent update --outdated
cdr agent update --outdated --kind storage
cdr agent update dc01.corp.example.com
cdr agent update --outdated --kind dc --user CORP\admin
```

`--kind` limits `--outdated` to `dc`, `storage`, or `host` agents. `--user` is needed only for a domain controller that must be pushed to; the password is read from the `CYGNADR_DEPLOY_PASSWORD` environment variable or prompted for. Without `--user`, `--outdated` skips those domain controllers, and agents that must be installed by hand, and prints a line for each saying why. The command follows the update for up to 30 minutes (`--wait` changes that) and ends with a table of each agent's route, outcome, and versions. It exits with 0 when every agent was updated or already current, 7 when any was refused, failed, or needs a hand install, and 8 when the update was still running when it stopped following it.

`cdr agent deploy` runs a push installation to domain controllers, including first installs, with a credential you give for that run. Run it on the Manager, in the Manager's program folder, or on any Windows machine you have copied `cdr.exe` to:

```
cdr agent deploy --outdated --user CORP\admin --package "C:\Program Files\Cygna Labs\Disaster Recovery Suite Manager\redist\CygnaDR.DcAgent.msi" --plan
```

`--plan` reads each target's installed version and reports what would happen without changing anything. Run the command again without `--plan` to update. `--domain` limits the selection to one domain, and `--transport dcom` uses DCOM. The command exits with 0 when every target ended up where it should, including targets that were already current, and 7 when at least one did not; the table says which and why. For every option of both commands, see Agent and Trust Commands.

Every update is recorded in the audit trail: one `agent.deployment.requested` entry per target, then `agent.deployment.completed` or `agent.deployment.failed` with the route, the outcome, and the versions before and after.

Updating agents automatically

By default, an operator updates agents. With **Update agents automatically** on, the Manager updates them itself, as each agent can take an update. It updates to whatever is in the package folder, so putting a new build there, or upgrading the Manager, is how a new version reaches every agent. With no package folder, or no package of an agent's kind, nothing is updated.

To turn it on, select **Administration** > **Configuration**, and in the **Agent deployment** group on the **Manager** tab turn on **Update agents automatically**. Two settings beside it set the pace: **Check for agents to update every** (30 minutes by default) and **Agents updated at once** (8 by default). Or use `cdr`:

```
cdr config set CygnaDR:Deployment:AutoUpdate true
cdr config set CygnaDR:Deployment:AutoUpdateIntervalMinutes 60
cdr config set CygnaDR:Deployment:AutoUpdateMaxAtOnce 16
```

Changing them needs the `config.manage` capability and is audited like any setting. A change takes effect within a minute, with no restart. The first check after the Manager starts is three minutes in.

Each check looks at every DC agent, the storage machine, and every approved Isolated Recovery Host that is behind the package for its kind. A candidate or replaced storage agent and a host awaiting approval are never updated. Up to **Agents updated at once** are updated together. When more are ready than that, the rest are queued, and the next starts within a minute of one finishing, without waiting for the interval, until none are left. Raise **Agents updated at once** to work through a large estate faster, at the cost of more package traffic at once over the links to remote sites. Updates an operator starts are not counted against it.

Each update is the same self-update an operator starts: over the agent's own connection, checked by the agent before it installs anything, with no credential. It is recorded in the audit trail as a deployment requested by **Automatic agent update**. Automatic updates never push to a domain controller with a credential and never install anything by hand; those stay an operator's job.

Every outdated agent is in one of these states:

State	Meaning	What happens next
Update started	Connected, able to update itself, and idle. Its update was started.	It reconnects on the new build.
Updating	An update of it is running, started by a check or by an operator.	Nothing until it finishes. Once updated, it is no longer listed or counted.
Queued	Ready, but as many as **Agents updated at once** are updating already.	Started within a minute of one of them finishing.
Waiting: busy	An update would cut short work in flight: a backup or restore on it; for the storage machine, any backup or restore at all; for a host, a test restore; for a domain controller, a recovery plan running in its forest. The storage machine also declines an update itself while it builds a recovery image or reads a backup to check it.	Looked at again on the next check.
Waiting: not connected	Not connected.	Looked at again on the next check.
Waiting: next check	An update an operator started did not complete. Automatic updates hold nothing against it.	Looked at again on the next check, like any other.

State	Meaning	What happens next
Cannot update itself	Its machine's `AllowRemote` policy switches updates from the Manager off. It needs an operator.	Never retried automatically. Update a domain controller from the **Agents** screen with a credential, or install the package by hand.
Last update failed	Its last automatic update did not complete; the reason is shown. Being busy when its turn came is not a failure. It needs an operator.	Tried again after six hours, or at once when a newer package is put in the folder.

To see what automatic updates are doing:

- **Agents screen**—One line says whether automatic updates are on, how many agents are updating, queued, and waiting, and when the Manager last checked. The agents that need an operator are counted beneath it and listed, with why, when you expand the count. **Check now** runs a check at once. See Automatic updates.
- **`cdr agent auto-update`**—The same line, and the agents that need an operator. `--all` lists every outdated agent and why it is waiting, and `--check-now` checks first. It exits with code 4 when an outdated agent needs an operator, so a script or monitor can act on it. See Agent and Trust Commands.
- **The Manager's log**—Says what each check started.

To keep one machine out of automatic updates, set its `AllowRemote` policy to `false`, as described in Switching off updates from the Manager. The agent then refuses every update from the Manager, automatic or not, and is listed as needing an operator.

Keeping DC agents current for recovery

The Manager does not send a recovery plan step to a DC agent whose build is behind the Manager's DC agent package. The step fails with a message that names both builds and how to update the agent: `cdr agent update`, or **Update** on the **Agents** screen; once the agent is updated, retry the step and it runs on the new build. The comparison is by build number. The Manager ships a DC agent package of its own build, so after every Manager upgrade the DC agents need updating. Because of this:

- After every Manager upgrade, update the DC agents at once, with **Update *n* outdated agents…** or `cdr agent update --outdated`.
- Update DC agents before any recovery plan runs. A restore puts back the agent that the restored DC will start, and restore preflight warns with `agent.restored-behind`

when that agent will be behind; update the machine that stages the restore while it is still reachable.

- Avoid upgrading the Manager while a recovery is under way.

A restored DC starts the agent build of the machine that staged the restore, when the recovery image can put it in place, rather than the older build in its backup.

Keeping the storage agent current

Update the storage agent when you upgrade the Manager: recovery images are rebuilt when the recovery agent shipped with the storage agent changes, and only a current recovery image can give a restored DC the agent build of the machine that staged the restore. The storage agent updates itself like any other agent, from the console or with `cdr agent update --outdated --kind storage`, between backups. See Upgrading and Uninstalling.

Restarting agents

Some agent settings are marked **Restart required**: they are saved at once, and come into force when the agent's service next starts. The Configuration screen counts the agents waiting for a restart. You can restart them through the Manager, without signing in to each machine:

- On the **Configuration** screen, select **Restart them…** beside the count of agents waiting for a restart. When only some of them can be restarted from the console, the button reads **Restart *n* that can be restarted from here…**.
- On the **Agents** screen, select **Restart the agent on *machine*…** in a row's menu, or select rows' checkboxes and then **Restart *n* agents…**. See Agents.
- With `cdr agent restart`, naming the agents or giving `--restart-pending` for every connected agent waiting for a restart:

```
cdr agent restart --restart-pending
cdr agent restart --restart-pending --kind dc
cdr agent restart dc01.corp.example.com storage01
```

Each agent's service stops and starts again, and the agent reconnects within a few seconds. The Manager refuses to restart an agent that is not connected, one that is not running as its service, and one whose work a restart would cut short, and says which and why. Work that holds a restart back: for a DC agent, a backup of the domain controller, a restore involving it, or any running recovery plan; for the storage agent, any running backup or any restore reading from the vault; for an Isolated Recovery Host agent, a test restore on its host. An agent that is not connected starts on the saved settings whenever its service next starts. Restarting needs the `agents.manage` capability, and each request is recorded in the audit trail.

Updating an agent by hand

A storage agent or Isolated Recovery Host agent that cannot update itself is installed by hand on its machine, and so is any agent when you prefer not to update it from the Manager. The **Agents** screen says so in the agent's row menu. Get the package in one of these ways:

- On the **Agents** screen, select **Install agent** in the page header, and then select the download button beside the agent's install line. Its tooltip gives the version and SHA-256 hash. Downloading needs the `agents.manage` capability.
- With `cdr agent package download storage` (or `dc`, or `host`), which checks the file against the Manager's SHA-256 hash.
- From the `redist` folder in the Manager's program folder.

Then run it on the machine in an elevated command prompt:

```
msiexec /i CygnaDR.StorageAgent.msi /qn
msiexec /i CygnaDR.HostAgent.msi /qn
```

The installer keeps the Manager address, CA fingerprint, enrollment token, and, for the storage agent, the storage root from the existing installation.

Every Cygna Forest Recovery package is signed by Cygna Labs Corp. To check a copy before you install it, run `Get-AuthenticodeSignature` on it in PowerShell and expect the status `Valid` with `O=Cygna Labs Corp.` in the signer, or open the file's **Properties** > **Digital Signatures** tab.

Switching off updates from the Manager

Whether an agent accepts updates from the Manager is set on its machine, never from the Manager. Set it in the registry, directly or through Group Policy, under this key, where `<agent>` is `DC Agent`, `Storage Agent`, or `Isolated Recovery Host Agent`:

```
HKLM\SOFTWARE\Policies\Cygna Labs\Disaster Recovery Suite\<agent>\CygnaDR\Update
```

- **`AllowRemote`**—`true` by default. Set it to `false` and the agent stops offering to update itself; the console then offers a push install for a domain controller, or a hand install for any other agent. Automatic updates leave the agent alone and list it as needing an operator.

Use a `REG_SZ` value of `true` or `false`, or a `REG_DWORD` (0 is false, anything else is true). Restart the agent service after you change a value. No installer creates or deletes anything under this key, so it survives upgrades and uninstalls. The **Configuration** screen shows the setting, read-only, as **Allow updates from the manager** in each agent scope's **Local to each machine** group.

Re-pointing agents at a moved Manager

Agents do not look the Manager up. Each one dials the address it was installed with and trusts only the CA fingerprint it was installed with. If the Manager's address changes, or its CA is replaced, every agent has to be told.

Note: The console's install and update actions do not re-point agents. An agent that is already current is left alone by them. Use `cdr agent deploy --repoint` for DC agents, and reinstall the storage agent and any Isolated Recovery Host agent.

After the Manager's address changes

The CA has not changed, so agents keep their fingerprint and only need the new address. From a machine where `cdr` trusts the Manager:

```
cdr agent deploy dc01.corp.example.com dc02.corp.example.com --repoint --force --manager-address https://10.0.0.20:5211 --user CORP\admin --package CygnaDR.DcAgent.msi
```

`--repoint` writes the new address to targets that already have an agent; `--force` reinstalls targets that are already at the package's version, so the agent restarts with the new address. Use the Manager's IP address, not its name. Name the domain controllers, or select them with `--all`, which also installs the agent on any domain controller that has none.

For the storage agent, and for any agent you installed by hand, reinstall with the new address on that machine. A property given when you reinstall replaces the stored value:

```
msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.0.0.20:5211
```

After the Manager CA is re-keyed

A re-keyed Manager has a new CA, and every agent refuses it until it is given the new fingerprint. Run `cdr agent deploy --repoint --force` as above, from a machine where `cdr` trusts the re-keyed Manager: the deployment pins the fingerprint of the Manager that `cdr` trusts. Reinstall the storage agent, and any Isolated Recovery Host agent, with `MANAGERCAFINGERPRINT` set to the new fingerprint, and `MANAGERADDRESS` as well if it changed. The full procedure is in Re-Keying the Manager CA.

Re-enrollment

An agent enrolls again on its own when the Manager no longer accepts its certificate: for example, after the Manager is restored from a database snapshot taken before the agent enrolled, after a re-key, or if its certificate expired. The agent drops the certificate and enrolls again with the enrollment token it kept from installation. Nothing needs to be done on the machine, and the audit trail records the re-enrollment. The Manager refuses to enroll

a machine onto the record of an agent that is connected under that name at that moment, and audits the refusal; the machine retries once the other session has ended.

Re-enrollment needs the token the Manager was installed with. If you install a Manager with a new enrollment token, which you do after a compromise, agents holding the old token cannot enroll. For DC agents, `cdr agent deploy --repoint` writes the token you give it to every target: set `CYGNADR_ENROLLMENT_TOKEN` to the new token, or enter it when prompted. Reinstall the storage agent, any Isolated Recovery Host agent, and any agent you installed by hand on its machine with `ENROLLMENTTOKEN` set:

```
msiexec /i CygnaDR.StorageAgent.msi /qn ENROLLMENTTOKEN=<new token>
```

Note: The enrollment token is a credential that can enroll new agents. Windows Installer masks it in installer logs, but do not leave it in scripts or command history.

Certificates

Agent certificates need no maintenance:

- Each agent's certificate is valid for 365 days and is renewed automatically once two-thirds of its lifetime has passed. A failed renewal is retried, and the current certificate stays in use meanwhile.
- An agent whose certificate has expired, or was issued by a CA it no longer trusts, enrolls again with its token.
- The Manager's own server certificate is also valid for 365 days and is reissued automatically.

For how agents trust the Manager, see Certificates and Trust.

Removing an agent

Before you remove a DC agent, remove the machine's backup policy on the **Backup Policies** screen, so the Manager stops scheduling backups for it.

The storage machine cannot be uninstalled or forgotten from the Manager: replace it with another storage agent first, and then remove it. Before you remove an Isolated Recovery Host agent, drain the host so that no test restore is cut short. See Moving the Storage Machine or an Isolated Recovery Host.

A connected agent can uninstall itself when the Manager asks, so you do not need to sign in to its machine. On the **Agents** screen, open the agent's row menu, select **Uninstall the agent from *machine*...**, enter the machine's name, and select **Uninstall the agent**. With the command-line tool, run `cdr agent uninstall`, which only says what it would do until you give the agent ID again with `--confirm`:

```
cdr agent uninstall dc05.corp.example.com
cdr agent uninstall 0847c3af-ec9c-4389-bcb1-7b706c5dc164 --confirm 0847c3af-ec9c-4389-bcb1-7b706c5dc164
```

The Manager refuses to uninstall the agent of a domain controller that is neither retired nor excluded, so that no domain controller loses its protection unnoticed: retire it, or exclude it with a reason, first (see Excluding and Retiring Objects). It also refuses while the agent has work in flight, and the agent refuses when it was not installed by its own package or its machine's `AllowRemote` policy is `false`. The agent runs the uninstall from a one-time scheduled task as SYSTEM, and writes the installer log to the Windows temporary folder on its machine. `cdr agent uninstall` prints the log's path, and the audit trail entry records it. Uninstalling needs the `agents.manage` capability, and each request is recorded in the audit trail.

To uninstall an agent on its machine instead, run its package with `/x` in an elevated command prompt:

```
msiexec /x CygnaDR.DcAgent.msi /qn
msiexec /x CygnaDR.StorageAgent.msi /qn
msiexec /x CygnaDR.HostAgent.msi /qn
```

Uninstalling removes the service, the program folder, and the registry configuration. It leaves the agent's data under `%ProgramData%\Cygna Labs\`, including job transcripts. Backups in the vault are never touched and stay under the vault retention rules.

However you uninstall it, the Manager keeps the agent on the **Agents** screen as not connected, and raises **Enrolled agents that are not connected** for it. What to do next depends on why you removed it:

- **The domain controller is being decommissioned**—Demote it and remove it from the directory as usual. Once every agent with a complete view of the directory stops seeing it, discovery records it as removed and it is no longer counted. See Excluding and Retiring Objects.
- **The domain controller stays, but should not be protected**—Exclude it, with a reason, so it is left out of the counts and alerts.
- **You are reinstalling the agent**—Install it again with the usual properties. It enrolls under the same machine name and reuses its existing identity.
- **The machine is gone for good**—Delete the agent's record from the Manager, as described below.

Deleting an agent's record

The Manager keeps one agent record for each machine name and agent kind. When a recovery rebuilds a domain controller onto another machine, the Manager removes the replaced machine's record itself. To delete the record of an agent whose machine is gone

for some other reason, select **Forget this agent record…** in its row menu on the **Agents** screen, which is offered only when nothing stops the deletion, or use `cdr agent forget` with the agent ID that `cdr agent list --wide` shows. Without `--confirm`, the command only says what it would delete; give the ID again with `--confirm` to delete it:

```
cdr agent forget 0847c3af-ec9c-4389-bcb1-7b706c5dc164
cdr agent forget 0847c3af-ec9c-4389-bcb1-7b706c5dc164 --confirm 0847c3af-ec9c-4389-bcb1-7b706c5dc164
```

The Manager refuses while the agent is connected, while a restore run that has not finished names it, while an approved, running, or halted recovery plan has bound it as a candidate, while a test restore runs on it, or when it is the storage machine. The record's session history and backup jobs are kept, and the audit trail keeps a copy of the record. If the machine is still running, its agent enrolls again under its own name on its next connection. Deleting a record needs the `agents.manage` capability. For the options, see Agent and Trust Commands.

Moving the Storage Machine or an Isolated Recovery Host

You may need to move the storage machine or an Isolated Recovery Host, for example onto new hardware or from a pilot setup into production. The two moves work differently:

	Storage machine	Isolated Recovery Host
What is lost	Every restore point. Backups start again from zero on the new machine.	Nothing. The test restore history stays.
Who can do it	An operator holding both `agents.manage` and `backup.manage`, or two operators: one asks, and one holding the other capability approves.	An operator holding `agents.manage`.
Old and new side by side	The new storage agent can be installed beside the old one, but it does no work until the storage machine is replaced with it.	Yes. Any number of approved hosts share the test restores.

The same machine at a new IP address is not a move. Change the address, keep it fixed, and restart the agent on that machine.

One storage machine, and candidates

An installation has one storage machine. The first storage agent to enroll becomes it. Any storage agent that enrolls after that is a *candidate*: it is listed on the **Agents** screen as **Candidate, given no work**, it is sent no backups, restores, or configuration backups, and it raises the alert **Storage agents that are not the storage machine** until it is used or removed. The storage machine is the only one the backups, the configuration backups, the storage alerts, and the **Repository** screen read.

That alert is also what an unexpected storage enrollment looks like. If nobody installed the candidate, uninstall its agent on that machine, delete its record with `cdr agent forget`, and treat the enrollment token as known to whoever installed it.

The storage machine itself cannot be uninstalled or forgotten from the Manager; the Manager refuses with **It is the storage machine**. Without a storage machine, the next storage agent to enroll would become it with nobody deciding. Replace it, and then remove it.

Replacing the storage machine

What a replacement does

- **Restore points are not moved.** Every restore point on the current storage machine stops being restorable when the replacement is carried out. The backup history stays, in backup jobs, reports, and the audit trail; only the ability to restore from those backups is lost. Restore points ended this way show **Storage machine replaced** as the rule that removed them. The confirmation gives the number, for example *412 restore points across 37 DCs will no longer be restorable; backups start again from zero on the new machine.*
- **Backups that were never committed are given up.** Their content is on the old machine.
- **Every domain controller with an enabled backup policy is backed up at once** to the new machine, within the usual limit on concurrent backups, instead of at its next scheduled time. Until its backup finishes, a domain controller has no restore point.
- **The configuration backup** is sent to the new machine straight away. See Manager Configuration Backups.
- **OS media and recovery images**—The old machine's OS media and recovery images leave the library. The new machine builds recovery images from its own media folder, so copy the installation ISOs there. Images that had passed a test restore are tested again.
- **Retention floors are not carried over.** The retention floor and the minimum versions per DC are set in each storage machine's own `appsettings.json` and cannot be set

over the network. The confirmation shows both machines' values side by side and marks the new machine's as **Weaker** where it protects less. Set them on the new machine before you replace. See Settings Reference.

- **The old machine still holds every backup it took,** each domain controller's directory database included. Nothing in Cygna Forest Recovery can delete it, because no network path deletes vault content. Uninstall its agent and wipe its storage root afterward.

Before you begin

The replacement is refused, and a request for it stays pending, while any of these is true:

- A restore, a recovery onto a new VM, or a test restore has not finished.
- A recovery plan is approved, running, or halted.
- A backup is running. Wait for it, or cancel it on the **Backup Jobs** screen or with `cdr jobs cancel`.
- The new storage agent is not connected.

To replace the storage machine

1. Prepare the new machine as you would any storage machine: not joined to a domain, with a fixed IP address, a volume for the storage root, and the staging share. If it is to hold OS media, copy the ISOs into its media folder. If the old machine's retention floor or minimum versions per DC were changed from the defaults, set them in the new machine's `appsettings.json`. See Setting Up the Storage Machine.
2. Install the storage agent on it with the same Manager address, CA fingerprint, and enrollment token as any agent:

   ```
   msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERADDRESS=https://10.0.0.10:5211
   MANAGERCAFINGERPRINT=<fingerprint> STORAGEROOT=E:\CygnaDR
   ENROLLMENTTOKEN=<token>
   ```

   It enrolls as a candidate and is given no work.
3. Look at what the replacement would do. On the **Agents** screen, open the candidate's row menu and select **Replace the storage machine with *machine*…**, or run:

   ```
   cdr storage replace <candidate agent id>
   ```

   Nothing changes yet. Check the number of restore points that end, the comparison of the two machines' local settings, and anything that blocks the replacement.

4. Ask for it. In the dialog, enter the candidate's agent ID and select **Replace the storage machine** or **Ask for the replacement**, or run:

   ```
   cdr storage replace <candidate agent id> --confirm <candidate agent id> --note "Moving to production"
   ```

   - If you hold both `agents.manage` and `backup.manage`, as the break-glass account does, the storage machine is replaced in this one step.
   - Otherwise this creates a request that waits for an operator holding the capability you lack. A System Admin's request waits for a Backup Operator, and a Backup Operator's for a System Admin. The alert **Requests waiting for a second operator** is raised, and emailed with the other alerts, and the request appears on the **Agents** and **Repository** screens.
5. The second operator approves it. On the **Repository** screen, select **Review and approve…** on the request, or run:

   ```
   cdr request show <request id>
   cdr request approve <request id> --confirm <candidate agent id>
   ```

   Approving shows what the replacement would do now, which can differ from when it was asked for, and asks for the candidate's agent ID again. Select **Approve and replace the storage machine**. It is carried out at once.
6. Watch the first backups on the **Backup Jobs** screen. A domain controller whose backup could not start is named in the result and in the Manager's log.
7. Retire the old machine. Uninstall its agent from the **Agents** screen or with `cdr agent uninstall`, delete its record with `cdr agent forget`, and then wipe its storage root and remove its staging share. Until it is wiped, it holds a copy of every domain controller's directory database.

A replaced machine is listed on the **Agents** screen as **Replaced** and is given no work. To use it again, delete its record with `cdr agent forget` and let its agent enroll again; it comes back as a candidate.

Requests

An action that needs two capabilities, such as replacing the storage machine, runs as a request when the operator who asks holds only one of them. Nothing happens until it is approved.

- A request lapses after 24 hours if nobody decides it. It is withdrawn if the candidate's record is deleted or another storage agent becomes the storage machine first.

- The requester cannot approve or decline their own request, and can cancel it. An operator who could approve it can decline it, with a note. Any operator holding one of the two capabilities can cancel it.
- One storage replacement can be pending at a time.
- Pending requests, and those decided in the last week, are shown on the **Repository** screen. The **Agents** screen shows the pending ones. `cdr request list` lists them, and `--all` lists every request. The command exits with code 4 while one is pending.
- The audit trail records each step: **Request made, waiting for a second operator**, **Request approved**, **Request declined**, **Request cancelled**, **Request lapsed**, **Request withdrawn**, and **Storage machine replaced** with the counts.

For the commands, see Storage and Media Commands.

Moving an Isolated Recovery Host

Nothing is lost when a host moves. Each test restore keeps the name of the host it ran on, so its history outlives the host, and test restores go to whichever approved host can run one. A move is therefore: add the new host, drain the old one, and remove it.

Approving and draining hosts

A host is given test restores only once an operator approves it. A newly installed host enrolls **Awaiting approval**, however ready it is, and raises the alert **Isolated Recovery Hosts awaiting approval**. Approve only a host you installed: any machine holding the enrollment token can enroll one.

Draining a host stops new test restores going to it. The test restore running on it finishes. A test restore that was planned for it but has not started is refused there; plan it again and it goes to another host. Once nothing runs on it, the host reads **Drained** and can be removed. You can put a drained host back in use.

Approving, draining, and putting a host back in use need `agents.manage`, and each is recorded in the audit trail. Do them in the host's panel on the **Recovery hosts** tab of the **Agents** screen with **Approve…**, **Drain**, and **Use it again**, or with `cdr recovery-host approve`, `drain`, and `undrain`. Backups show as not verified only while no approved host is ready.

To move an Isolated Recovery Host

1. Prepare the new host and install the Isolated Recovery Host agent on it. See Setting Up an Isolated Recovery Host.
2. Give it its own settings, if it needs any. The folder for verification VMs and the recovery network prefix apply to every host unless a host has its own value as an override. An override belongs to the old host's agent and does not carry over: read the old host's

overrides on the **Configuration** screen, and set the ones the new host needs with **Override a setting on this machine…** or `cdr config set --agent`.

3. When the new host reads **Ready**, approve it: select **Approve…** in its panel on the **Agents** screen, or run `cdr recovery-host approve <host agent id>`.
4. Drain the old host: select **Drain** in its panel, or run `cdr recovery-host drain <old host agent id>`. Wait until its panel, or `cdr recovery-host list`, says it is drained.
5. Remove the old host: uninstall its agent with `cdr agent uninstall`, and then delete its record with `cdr agent forget`. Deleting the record is refused while a test restore still names the host. If the machine stays in service for something else, remove the **CygnaDR Recovery** switch, its NAT, and the folder for verification VMs.

Related topics

- Agents
- Maintaining Agents
- Managing Operators and Roles
- Test Restores

Alerts Reference

This topic lists every alert the Manager raises, what raises it, and what to do about it. Alerts appear on the **Alerts** screen and in the notifications menu in the top bar, and `cdr alert list` returns them. For the screen, see Alerts.

The Manager raises one alert per condition and names up to ten of the affected objects. Domain controllers recorded as removed from the directory appear in no alert, and neither do domain controllers that are rebuilt on recovery: those with no enabled backup policy and nothing in the vault. Objects that an operator has excluded raise no alerts; they are counted separately on the **Alerts** screen. The exceptions are the alerts about the backups themselves and the staging share, which are raised whether or not the domain controller is excluded: **Vault content whose hash does not match**, **Committed vault content changed outside retention**, and **Unexpected SMB connections to the staging share**. The code shown under each title is what `cdr alert list` and its `--json` output use.

Critical alerts

A critical alert means your ability to recover is impaired now.

Alert	Raised when	What to do
Backups and recovery operations stopped: no active license `license.stopped`	No license is installed, the installed key does not license Cygna Forest Recovery for Active Directory, or the license ended and its grace period is over. New backups, restores, recovery plans, and test restores are refused; work already under way finishes. The alert names the license's standing in one sentence.	Apply a current license key on the **License** screen or with `cdr license apply`. Contact Cygna for a renewal. See Licensing.
License ended: grace period running `license.grace`	The license's end date has passed. Everything still works until the grace period is over, then backups and recovery operations stop.	Apply a renewed license key on the **License** screen or with `cdr license apply` before the date named.
Domains that could not be recovered `domain.unrecoverable`	No writable domain controller in the domain holds a restorable backup, so a recovery has nothing to restore its first DC from.	Back up at least one writable domain controller in each domain with a full system backup. A read-only DC cannot start a recovery.
Forests that could not be recovered `forest.unrecoverable`	The forest root domain cannot be recovered, or a domain under it cannot.	Open the **Forests** screen, which says for each forest whether the root or a child domain is the problem, and fix the domains named.
Domains past the tombstone lifetime `domain.beyond-tombstone`	The newest restorable backup of every writable domain controller in the domain is older than the tombstone lifetime. Restore preflight refuses such backups.	Take a fresh backup of a writable DC in each domain, then find out why backups stopped.

Alert	Raised when	What to do
Domains replicating SYSVOL by FRS `domain.sysvol-frs`	The newest backup reports that SYSVOL replicates with FRS rather than DFSR. Restore preflight refuses a domain on FRS.	Migrate SYSVOL replication to DFSR, then take fresh backups. Backups taken before the migration stay unusable.
Vault refusing backups `vault.full`	Free space on the vault volume is below the free space reserve, so the storage agent refuses every new backup.	Add capacity or reduce retention. Backups are failing now, and recovery capability degrades for as long as this lasts.
Domain controllers whose latest test restore failed `restore.verification-failed`	The newest test restore of the domain controller's backups, on an Isolated Recovery Host, did not come up as a working first DC: the restore failed, a check did not pass, or it never reported. Test restores that ended without a verdict are ignored.	Open the test restore on the **Recovery Plans** screen: its verdict names the step and the reason. Then test a newer backup. See Running a Test Restore.
Vault content whose hash does not match `vault.hash-mismatch`	At its last scheduled verification, a committed backup no longer hashed to the value recorded when it was committed: corruption or tampering. A version that could not be read, or has no recorded hash, is not in this alert, because the check was not made. Raised for as long as the version's latest verification result is a failure.	Do not restore from the versions named. Check the health of the vault volume and who has had write access to the storage machine, and take a fresh backup of each domain controller named. Each version's failing checks are on the **Backup verification** tab of the **Repository** screen; the content is left in place as evidence. See Vault

Alert	Raised when	What to do
		Protection and Integrity.
Unexpected SMB connections to the staging share `storage.unexpected-smb`	The storage agent saw a use of staging that does not fit a backup job: a job's account used from an address that is not its domain controller's, before its credential was issued or after it was removed, or an account name of that form with no job; or any other account reaching the staging share, another share over the staging folder, or a file under the staging folder through any share, an administrative share such as `D$` included. Raised for seven days after it was last seen. Each finding names the account, the address, and what was wrong.	Treat the job's credential as in the wrong hands until shown otherwise. Find the machine at each address named, check who else has had access to the storage machine, and take a fresh backup of the domain controller whose job it was. The storage agent reports only: it closes no session and removes no account early. The audit trail records the first sighting of each finding as **Unexpected use of the staging share**; **View audit trail** opens it.
Committed vault content changed outside retention `storage.mass-delete`	Something other than retention deleted, renamed, wrote to, truncated, or changed the permissions or attributes of a committed backup. A single file is enough. Each version is named with what was done, how many paths, and over how long; `cdr repository changes` also says who did it, where the storage machine's audit policy recorded it. Raised until someone records a review.	Treat the storage machine as compromised until shown otherwise: keep it isolated and find out who has write access to the vault volume. Do not restore from the versions named. List the changes with `cdr repository changes --paths`, and once a version's changes are understood, record the review with `cdr repository changes`

Alert	Raised when	What to do
		`review <version> --note <text>`, which clears the alert for that version. Nothing was blocked or reverted. See Vault Protection and Integrity.
Domain controllers silent past the tombstone lifetime `dc.replication-beyond-tombstone`	No DC agent's domain controller has replicated from this domain controller for longer than the tombstone lifetime, taken as 180 days, the Active Directory default. Its copy of the directory is too old to rejoin: reconnected, it brings back objects the rest of the forest has deleted, as lingering objects. The alert is raised only for a domain controller that some DC agent's domain controller replicates from; a machine nothing observes is not called silent. The **Alerts** screen shows when the longest-silent one last replicated.	Keep it off the network. If it is gone, remove it from the directory with `cdr cleanup dc <name>`, or `cdr cleanup domain` when it was its domain's last DC. If it is still needed, rebuild it—forced demotion, metadata cleanup, then promote it again—rather than repairing its replication.

Warning alerts

A warning means you can recover today, but for a reason that may not hold tomorrow.

Alert	Raised when	What to do
Mass removals from the directory held for review `topology.removal-held`	More than a quarter of a forest's domain controllers or domains, or of a domain's trusts, disappeared from the directory at once. Discovery records removals on its own, but not this many: it is what an attack, a mistaken cleanup, or a permissions change that hides objects from every agent looks like.	Confirm in the directory that each named object really is gone. If it is, record each one as removed with `cdr cleanup retire <name> --kind <kind> --reason <text>`, pasting the name as the alert lists it. If it is not, find out why the agents stopped seeing it.
Objects removed from the directory `topology.removed`	Discovery recorded a domain controller, domain, or trust as removed, because every agent with a current, complete view of the	Confirm that each removal was intended. An unexpected one is a security finding: the object is no longer counted, although recovery plans can still use restore points taken before it went.

Alert	Raised when	What to do
	directory stopped seeing it. Raised for seven days after the removal.	
Backup targets with nothing to restore from `dc.unprotected`	A domain controller with an enabled backup policy has no restorable backup: the first backup has not completed, every backup failed, or every version was pruned. A domain controller with no enabled policy and nothing in the vault is not a backup target and is in no alert, because recovery rebuilds it. Whether a missing backup costs the domain its recovery is decided by	Open the domain controller's backup jobs to see why nothing is landing, then run a backup. See Backup Targets and Rebuilt Domain Controllers.

Alert	Raised when	What to do
	Domains that could not be recovered, which is raised on its own.	
Global catalogs with no agent `dc.no-agent`	A global catalog has no enrolled agent, and its forest holds another domain this installation would recover. Recovering any other domain in the forest reconciles every global catalog through its agent, so that recovery's plan refuses to start until this one has an agent or stops being a global catalog. Domain controllers that are not global catalogs need no agent unless they	Install the DC agent, or remove the machine's global catalog role.

Alert	Raised when	What to do
	are backed up.	
Domain controllers whose last backup failed `backup.failing`	The most recent backup of a domain controller failed. An older restore point still exists, so no data is lost yet. A backup refused because the domain controller had no enrolled agent yet does not count as a failure.	Open the job on the **Backup Jobs** screen for the recorded error and transcript. See Backup Job Errors.
Backups past their freshness window `backup.stale`	A domain controller's newest restore point is older than twice its policy interval.	Check that the agent is connected, then run a backup.
Domain controllers with no backup schedule `backup.unscheduled`	A restore point exists but no enabled policy keeps it current, so it only gets older.	Give each one an enabled policy on the **Backup Policies** screen.

Alert	Raised when	What to do
Domains with no full system recovery option `domain.mode-a-only`	Every restorable backup of a writable domain controller in the domain is a DC state backup, which cannot recover a machine whose operating system is gone.	Change the policy of at least one writable DC in each domain to **Full system**.
Topology stores that are stale `topology.stale`	No DC agent in the forest has reported the directory for longer than twice the topology report interval, so inventory figures are only as current as the last report.	Check that a DC agent is running and connected in each forest named.
Vaults with no room for another version `vault.tight`	The vault has free space, but not enough for another copy of the largest version it holds, so the largest	Add capacity or reduce retention before the next backup of the largest DC.

Alert	Raised when	What to do
	domain controller's next backup will not fit.	
Domain controllers with no recent verified backup `restore.unverified`	The domain controller's newest backup has not passed a test restore, and its backups have gone untested for longer than the **Test each DC's new backups within** period. Raised only while an Isolated Recovery Host is enrolled, and only for domain controllers with full system backups. Setting the period to 0 turns this alert off.	The schedule starts each test in time to pass before this deadline, so a domain controller named here was not reached in time: a host that is not ready, backups that keep it waiting, failing tests, or more domain controllers than the hosts can test in a period. Add a host or lengthen the period if it keeps happening; check the host's panel on the **Recovery hosts** tab of the **Agents** screen. You can also start one yourself from the **Recovery Plans** screen. See Running a Test Restore and Test Restores.
Exclusions due for review, or voided `exclusion.laps`	An exclusion reached its review date,	Review each one: extend it with a fresh reason, or clear it and deal with what it was hiding.

Alert	Raised when	What to do
`ed`	or the object it was written about changed identity. An exclusion on a domain controller or domain since recorded as removed is left out.	
Staging that could not be created `staging.mint-failed`	The storage agent could not give a domain controller's backup a staging area or a per-job write credential, and no later backup of it has been given one. Raised until a later backup of the domain controller is. Not raised for a domain controller that is excluded or no longer in the directory. A backup refused for lack of space	Open the failed jobs on the **Backup Jobs** screen; each records the storage agent's reason. The cause is on the storage machine—the storage agent's rights, its password policy, the Cygna Forest Recovery local groups, or the staging volume—so retrying the backup alone does not clear it. See Backup Job Errors.

Alert	Raised when	What to do
	raises **Vault refusing backups** instead.	
krbtgt second reset due `recovery.krbtgt-second-reset`	A domain in a running or halted recovery plan has had its first krbtgt password reset and not its second. Tickets forged with the old key stay usable until the second reset. Raised for the whole wait.	Let the plan run its second reset, or, on a halted plan, resolve the halt so that it can. The plan does not reconnect to production until the second reset is done. If a plan is aborted between the two resets, its recovery report lists the second reset as a follow-up.
Recovery protection override active `recovery.protection-override`	In a running or halted recovery plan, seizing the operations master roles set `Repl Perform Initial Synchronizations` to 0 on a domain controller, and the plan's step that sets it back to 1 has not run	Let the plan's cleanup step run. If the plan will not reach it, set `HKLM\SYSTEM\CurrentControlSet\Services\NTDS\Parameters\Repl Perform Initial Synchronizations` back to 1 on each domain controller named, once its replication partners are healthy.

Alert	Raised when	What to do
	yet. The alert is read from the plan, not from the domain controller: a value changed by hand is not seen, and an aborted plan is not followed.	
Storage agents that are not the storage machine `storage.candidate`	A storage agent enrolled with the enrollment token while another is the storage machine. It is given no work: no backups, no restores, and no configuration backups.	If it is meant to replace the storage machine, replace it from its row menu on the **Agents** screen or with `cdr storage replace`; that ends every restore point and needs both `agents.manage` and `backup.manage`. If nobody installed it, uninstall the agent on that machine, delete its record with `cdr agent forget`, and treat the enrollment token as known to whoever did. See Moving the Storage Machine or an Isolated Recovery Host.
Requests waiting for a second operator `request.pending`	An action that needs two capabilities, such as replacing the storage machine, was asked for by an operator	An operator holding the capability named approves or declines it on the **Repository** screen, or with `cdr request approve` or `cdr request decline`. The requester can cancel it.

Alert	Raised when	What to do
	who holds only one of them. Nothing happens until an operator holding the other approves it, and it lapses after 24 hours.	
License ending soon `license.expiring`	The license ends within 30 days. A subscription then has a 14-day grace period; a trial stops the day after it ends. The alert names the end date.	Contact Cygna for a renewal and apply the new key on the **License** screen or with `cdr license apply`. See Licensing.

Could not find out

These alerts report something the Manager could not confirm. They are an absence of information, not a fault, and are counted as neither.

Alert	Raised when	What to do
Enrolled agents that are not connected `dc.agent-disconnected`	An agent has enrolled but is not connected now, so its figures are the last it reported.	Check that the agent service is running on each machine and that it can reach the Manager. See Agent Enrollment and Connectivity.
Forests never	No DC agent has ever reported	Install a DC agent in

Alert	Raised when	What to do
enumerated `topology.never-enumerated`	the forest's directory, so its inventory is only what machines claim.	the forest. It reports the directory when it connects.
Vault volumes that could not be measured `vault.unreadable`	The storage agent could not read the free space of its vault volume, or has never reported it. An unreadable volume is not a full one.	Check that the volume is present and that the storage agent can read it.

Alert	Raised when	What to do
Storage machines not fully watching the staging share `storage.smb-watch-unavailable`	The storage agent's watch on its staging share is off or partial, or it has not reported on it for 30 minutes. The reasons: the Security log cannot be read; the audit policy does not record logons, share connections, or file access through shares; the poll of sessions and open files fails; or running jobs have no known domain controller address to check against. Each machine is named with its reason. While a machine is in this alert, the absence of **Unexpected SMB connections to the staging share** from it is not evidence that there were none.	Act on the reason given. Cygna Forest Recovery never changes the audit policy: for every connection to be seen, however short, turn on success auditing for **Logon**, **File Share**, and **Detailed File Share** in the storage machine's Advanced Audit Policy. See Turning on auditing on the storage machine below.
Vaults not being watched for changes `storage.vault-watch-unavailable`	A storage machine cannot tell whether its committed backups are being changed: the change journal of the vault volume cannot be read (a file system without one, a network share, or a storage agent that is not running elevated), the storage agent has not reported recently, or the journal lost records in the last seven days, so that stretch of time was checked by comparing the vault with its commit records rather than watched. Each machine is named with its reason.	Act on the reason given. Keep the storage root on a local NTFS or ReFS volume, where the storage agent can always read or create a change journal, and keep the storage agent service running as LocalSystem. For a journal gap, `cdr repository changes` lists what the comparison found. Scheduled hash verification continues either way.

Notices

A notice tells you about something that waits on an operator but does not affect recovery. Notices are listed under **Notice** on the **Alerts** screen and are not counted in the bell's

badge.

Alert	Raised when	What to do
Isolated Recovery Hosts awaiting approval `recovery-host.awaiting-approval`	An Isolated Recovery Host agent enrolled and no operator has approved it, so test restores are not sent to it however ready it is.	If it is yours, approve it on the **Agents** screen or with `cdr recovery-host approve`. If it is not, uninstall the agent and delete its record. See Moving an Isolated Recovery Host.

Turning on auditing on the storage machine

Storage machines not fully watching the staging share names each machine with its reason, for example **Audit File Share is off in this machine's audit policy**. Cygna Forest Recovery never changes the audit policy, so turn the auditing on yourself. For full coverage of the staging share and the vault, turn on success auditing for Logon, File Share, Detailed File Share, and File System.

The storage machine is not joined to a domain, so set this in its local policy. You can use a command prompt or Local Security Policy.

To turn on the auditing from a command prompt:

1. On the storage machine, open a command prompt as an administrator.
2. Turn on success auditing for the four subcategories. The subcategories are given by their GUIDs, which work in every Windows language:

   ```
   auditpol /set /subcategory:"{0CCE9215-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE9224-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE9244-69AE-11D9-BED3-505054503030}" /success:enable
   auditpol /set /subcategory:"{0CCE921D-69AE-11D9-BED3-505054503030}" /success:enable
   ```

   In order, these are Logon, File Share, Detailed File Share, and File System.
3. Check the result:

   ```
   auditpol /get /subcategory:"{0CCE9215-69AE-11D9-BED3-505054503030},{0CCE9224-69AE-11D9-BED3-505054503030},{0CCE9244-69AE-11D9-BED3-505054503030},{0CCE921D-69AE-11D9-BED3-505054503030}"
   ```

   Each of the four lines should show **Success** (or **Success and Failure**).

To turn on the auditing in Local Security Policy:

1. On the storage machine, run `secpol.msc`.
2. Go to **Security Settings** > **Advanced Audit Policy Configuration** > **System Audit Policies - Local Group Policy Object**.
3. Under **Logon/Logoff**, open **Audit Logon**, select **Configure the following audit events** and **Success**, and select **OK**.
4. Under **Object Access**, do the same for **Audit File Share**, **Audit Detailed File Share**, and **Audit File System**.

If the machine also has basic audit policies set under **Local Policies** > **Audit Policy**, check that **Local Policies** > **Security Options** > **Audit: Force audit policy subcategory settings (Windows Vista or later) to override audit policy category settings** is **Enabled**, which is the default. Otherwise Windows ignores the subcategory settings.

You do not need to add audit entries to any folder: the storage agent adds the one that File System auditing needs on each committed backup. Detailed File Share logs every access to every share on the machine, so increase the maximum size of the Security log to keep enough history.

The storage agent reads the audit policy every 10 minutes, so **Storage machines not fully watching the staging share** clears without a restart once it sees the change, unless the alert gives another reason. To have it read the policy at once, restart the storage agent service, `CygnaDRStorageAgent`, while no backup is running.

For what each subcategory adds, see Audit policy on the storage machine.

Alerts by email

The Manager emails alerts to the **Recipients** in the **Alert notifications** settings, through the mail server in the **Email** settings. With no recipients, alerts are not emailed. The email reads the same alerts as the **Alerts** screen, so it never names an alert the screen does not show.

- **Per object**–Each object an alert names is reported on its own. A domain controller that joins an alert already raised for others is reported as raised; one that no longer matches is reported as cleared while the alert stays raised for the rest.
- **Raised, cleared, still raised**–The Manager checks at the **Check interval**. An object is raised when a check first finds it and cleared when two checks in a row no longer find it. One email per check lists everything raised and cleared, and below that what is still raised. Nothing is sent when nothing changed and no reminder is due.
- **Reminders**–An object still raised is reported again once the **Reminder interval** has passed since the last email that listed it.

- **Minimum severity**–Alerts below the **Minimum severity** are not emailed. Raising the minimum stops reports about the alerts it leaves out without reporting them cleared; lowering it reports the alerts it brings in as raised.
- **Leaving an alert's scope**–An object an alert no longer watches is dropped from email, not reported as cleared: a domain controller that stops being a backup target leaves **Backup targets with nothing to restore from**, and one that no recovery needs an agent on leaves **Global catalogs with no agent**. Leaving scope is not the same as being fixed.
- **After a restart**–The first check waits five minutes, or the check interval if that is longer, so agents can reconnect first.
- **Failed sends**–An email that could not be sent is held and sent with the next check, so a raised alert is not lost while the mail server is down. The **Alerts** screen then shows **Email not working**.

The subject line names the Manager machine, the worst severity listed, and the counts, for example `Cygna Disaster Recovery alerts on <machine> (critical): 1 raised, 2 still raised`. The email has HTML and plain-text parts and links to the **Alerts** screen. Times are in the Manager machine's local time. The email names that time zone once, beside when it was checked, with the offset in effect then, for example `Times are the manager's local time: Eastern Daylight Time (UTC-04:00).` The link uses the Manager machine's DNS name and HTTPS port, or the **Console address in emails** setting when recipients reach the console by another name. Each object is listed with the alert's title and, for raised objects, what to do; an alert lists at most 100 objects per section, then a count.

Every email sent is recorded in the audit trail as **Alert email sent**, and a failure as **Alert email not sent** when it starts and when its reason changes. For the settings and their defaults, see Settings Reference. To check that email works, use **Send test email** on the **Alerts** screen (see Alerts) or `cdr alert notify test`.

Monitoring alerts from other tools

To include Cygna Forest Recovery in your own monitoring, run `cdr alert list` on a schedule. It exits with code 4 when any alert is raised, and `--json` returns the alerts in machine-readable form. `--critical` limits the list to critical alerts. `cdr repository changes` also exits with code 4 while any change to a committed backup is open or any storage machine is not watching its vault. See Alert, Audit, and Report Commands.

Auditing Operator Activity

The Manager writes an audit entry for every gate result and every state change: an operator signing in, a policy or setting changing, a backup starting or failing, a restore being approved, a role changing hands. The audit trail shows who did what, from where, and on

whose authority. You read it on the **Audit Trail** screen, which every role can see, or with `cdr audit list`, and you can export it as CSV, JSON, HTML, or PDF.

For a description of the screen, see Audit Trail.

What is audited

- **Operators and sign-in**–Setup of the Manager and the break-glass confirmation, sign-ins, failed sign-ins, sign-outs, lockouts and unlocks, accounts created, changed, disabled, and enabled, password resets and changes, and multi-factor authentication being turned on, turned off, or reset, with each recovery code used.
- **Configuration**–Every settings change, retention changes separately, policy changes, the recovery passphrase being set, and changes to the Manager CA, including a re-key.
- **Agents**–Enrollment and re-enrollment, certificate renewal, refused messages, console-started deployments, and restart and uninstall requests made through the Manager.
- **Backups and the vault**–Each backup job's creation, completion, failure, interruption by a DC agent restart, and cancellation, refused backups, retention sweeps and deletions, failed verifications, and the hourly check of backups that did not reach the vault. The storage agent's findings are recorded too, each once: **Unexpected use of the staging share** (`staging.smb.unexpected`), **Changes detected in vault content** (`vault.change.detected`), a change to a committed backup outside retention, recorded once per backup version; **Vault changes reviewed** (`vault.change.reviewed`), a review of such a change with its note; and **Vault change watch lost its place** (`vault.watch.position-lost`), a vault watch that restarted without its saved position, so changes in that window went unwatched.
- **Configuration backups**–Each attempt to send a configuration backup to a storage machine and how it ended (`manager.config-backup.stored`, `manager.config-backup.failed`, `manager.config-backup.refused`), and each import with `CygnaDR.Manager.exe bundle import` (`manager.config-backup.imported`), which names the Windows account that ran it.
- **Support packages**–Each support package created (from the console, `cdr`, or the Manager's local `support package` command), each download, and each removal from the Manager (**Support package created**, **Support package downloaded**, **Support package deleted**).
- **Alert email**–Each alert email sent, with how many objects it listed as raised, cleared, and still raised (**Alert email sent**, `alert.email.sent`); a failure to send, when it starts and when its reason changes (**Alert email not sent**, `alert.email.not-sent`); and each test email, with the operator who asked (**Test alert email requested**, `alert.email.test`).

- **The audit trail itself**–Each export of the audit trail (**Audit trail exported**).
- **Topology**–Domain controllers discovered, objects recorded as removed or held for review, exclusions added and cleared, and operations master roles moving.
- **Recovery**–Every restore run and recovery plan action: planned, approved, approval withdrawn, executed, cancelled, discarded, abandoned, and its outcome, and for plans, each point-of-no-return acknowledgment, attestation, rejection, and overridden wait. Vault read leases issued and revoked for each restore are recorded too, and for a DC state recovery, each `krbtgt` reset.

Domain credentials given to a recovery plan or to a DC state recovery's `krbtgt` reset, and passwords used for push installation, are never recorded. Changes that `cdr agent deploy` and `cdr cleanup` make in the directory use your own Windows credential and are recorded in the domain controller's own security log.

What an entry records

Field	Description
When	When the entry was recorded. The detail sheet shows it as **Recorded (local)** and **Recorded (UTC)**.
Actor	Who caused it. For an operator's action this is the operator's username. Work the product does on its own is recorded as **System**, **Manager**, or **Storage agent**. A failed sign-in is recorded as **Unauthenticated caller**, because nobody had signed in.
Action	What happened, as a label and an action code such as `restore.run.approved`. A code the console has no label for is shown as the code.
Subject	What it happened to, usually a machine, domain, or object name.
Fields	The entry's detail. Every entry caused by a network request, from the console, `cdr`, or an agent, carries `remoteAddress`, the address the request came from, and `remoteHost`, the name reverse DNS gave for that address when one was found. Rely on the address: the host name is only as trustworthy as the reverse DNS zone. An entry with neither field came from the Manager's own scheduled work, not from the network.

An entry written by the Manager's local maintenance commands, such as `CygnaDR.Manager.exe ca recover`, names the Windows account that ran the command.

Finding entries

On the **Audit Trail** screen, the filters are applied by the Manager:

- **Actor**–One actor, such as an operator's username.
- **Action**–One action, listed with its label and code.
- **Subject contains**–Part of a machine or object name.
- **From** and **To**–A date range.

The Manager returns the newest 200 matching entries and the total count. The search box narrows the entries already returned; to reach older entries, narrow the filters. Select an entry to open its detail sheet, which shows every field and the detail exactly as stored.

From a script, `cdr audit list` takes the same filters:

```
cdr audit list --actor jsmith --from 2026-10-01 --detail
cdr audit list --action restore.run.approved --json
```

`--from` and `--to` take UTC times, as `yyyy-MM-dd` or a full timestamp. `--actors` lists every actor and action code in the trail. See Alert, Audit, and Report Commands.

Entries worth reviewing regularly

Action (code)	Why
Operator created (`operator.created`), **Operator roles or name changed** (`operator.updated`), `operator.removed`, `operator.restored`, `operator.mfa.reset`	Who was given which capabilities, who was removed or restored, whose MFA was reset, and by whom. A removal records the roles the account held; a restore records the roles it was given.
Sign-in failed (`operator.signin.failed`), **Account locked after failed sign-ins** (`operator.locked`), `operator.signin.mfa-challenged`, `operator.mfa.locked`	Password and MFA code guessing, and from which address. Each refused sign-in is recorded with its reason, such as a wrong password, a locked or disabled account, or too many failures from the address. A username that no account holds is recorded as `(unknown username)` with a fingerprint, never as typed, because it may be a password. A code request with no sign-in after it, or a lock on MFA codes, means someone has the operator's password.
Signed in (`operator.signin`)	The break-glass account should be used only in an

Action (code)	Why
by `breakglass`	emergency.
Retention changed (`config.retention-changed`), **Configuration changed** (`config.changed`)	Changes that can reduce what is kept.
Restore approved (`restore.run.approved`), **Restore execution requested** (`restore.run.execute-requested`)	Who approved and who executed each restore.
`restore.krbtgt.reset-requested`, `restore.krbtgt.reset`, `restore.krbtgt.reset-failed`	Who started a DC state recovery's `krbtgt` reset again, and each reset the domain controller's agent applied or failed. The credentials are never recorded.
`manager.config-backup.failed`, `manager.config-backup.refused`, `manager.config-backup.imported`	A configuration backup did not reach a storage machine, or a Manager was rebuilt from one.
`manager.ca.rekeyed`, `manager.ca.recovered`	The Manager's CA was replaced or recovered.
Storage machine replaced (`storage.replaced`), **Second storage agent enrolled as a candidate** (`storage.candidate-enrolled`), and the `change-request.*` entries	Who replaced the storage machine and how many restore points it ended, who asked for and decided each request, and when a storage agent enrolled that nobody may have meant to install.
Isolated Recovery Host approved (`recovery-host.approved`), `recovery-host.draining`, `recovery-host.undrained`	Who allowed a host to read backups for test restores, and who took one out of use.

For every multi-factor authentication action, see Multi-Factor Authentication.

Exporting entries

To give an auditor a record, or to archive the trail before you shorten its retention, export it. Select **Export** on the **Audit Trail** screen, or run `cdr audit export`. Exporting needs the `audit.view` capability, which every role includes.

- **What is exported**—Every entry that matches the filters and, in the console, the search box, across the whole trail. The 200-entry limit of the screen does not apply.
- **Formats**—CSV and JSON hold up to 100,000 entries; HTML and PDF, documents for a person to read, hold up to 5,000. When more entries match, the export is refused with the reason, never cut short. Narrow the dates or the filter, or export in parts.
- **Wording and times**—A console export uses the console's wording for actors and actions, with the codes beside them, and gives times in UTC and in your browser's time zone. `cdr audit export` writes the codes, and times in UTC and the time zone of the machine it runs on.
- **Spreadsheet safety**—In a CSV file, a value that starts with `=`, `+`, `-`, `@`, a tab, or a carriage return gets a leading apostrophe, so a spreadsheet does not run it as a formula. This matters because the subject of a failed sign-in is whatever an unauthenticated caller typed.
- **File name**—`cygnadr-audit-<yyyyMMdd-HHmmss>.<format>`, from the UTC time of the export.
- **Recorded**—Each export is itself an audit entry, **Audit trail exported** (`audit.exported`), written before the data leaves the Manager. Its subject is the file name, and its fields record the format, the number of entries, the filter, and the search.

For the formats and the export menu, see Audit Trail.

Retention of the audit trail

By default the audit trail is kept forever. The **Audit trail** setting under **Database retention** on the **Configuration** screen sets how many days entries are kept; 0 means forever. Shortening it deletes entries for good, including entries that record changes to settings, so saving it needs typed confirmation. Set a window only where your own data-protection rules require one. See Settings Reference.

The audit trail lives in the Manager database. If the Manager is restored from a backup or a database snapshot, entries written after that point are not in the restored trail. See Recovering the Manager.

Logs and Diagnostics

Each Cygna Forest Recovery service writes its own log file and also sends warnings and errors to the Windows Event Log. The DC agent keeps a transcript of every backup and

restore on the domain controller. Diagnose most problems from the console first: the **Backup Jobs** detail sheet shows a backup's error and transcript, and a restore run's page shows what the machine reported. Turn to the files in this topic when the console does not say enough, or when a service cannot reach the Manager at all.

Service logs

Component	Folder	File name
Manager	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\logs`	`manager-yyyyMMdd.log`
DC agent	`%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent\logs`	`dc-agent-yyyyMMdd.log`
Storage agent	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent\logs`	`storage-agent-yyyyMMdd.log`
Isolated Recovery Host agent	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Isolated Recovery Host Agent\logs`	`host-agent-yyyyMMdd.log`

The folders are readable by SYSTEM and Administrators only. A component installed on the same machine as another writes to its own folder.

- **Rotation**–A new file is started each day, and when the current file reaches 32 MB.
- **Retention**–Files older than 30 days are deleted. The **Log retention (days)** setting changes this for each machine; 0 keeps every file.
- **Format**–One line per entry: a UTC timestamp, the level (`TRACE`, `DEBUG`, `INFO`, `WARN`, `ERROR`, or `CRIT`), the source, and the message. For example: `2026-09-03 18:55:01.234Z INFO Firebreak.DcAgent.AgentWorker Dialing the manager at https://10.0.0.10:5211`
- **Level**–Information and above by default. To change it, set `Logging:LogLevel:Default` in the component's `appsettings.json`, or `Logging:CygnaDRFile:LogLevel:Default` to change the file alone, and restart the service.
- **Location**–The **Log folder** setting moves the log for one machine. An empty value turns the file log off; that is the only way to turn it off.

To follow a log while the service runs, use `Get-Content -Wait` in PowerShell. Some editors ask for exclusive access and refuse to open a file the service holds open.

At startup each service logs the configuration it came up with, the Manager address it dials (for agents), and the CA fingerprint it trusts. These lines are the first thing to check when an agent does not appear on the **Agents** screen.

Windows Event Log

Each service also writes warnings, errors, and critical entries to the Windows Application log. The Manager adds one informational entry at every start, giving its CA fingerprint:

> Cygna DR manager CA fingerprint: <fingerprint>. Agents are installed with this value, and cdr asks you to confirm it the first time it connects.

Otherwise the Event Log holds warnings and errors only; use the log files for anything below a warning.

Job transcripts on domain controllers

The DC agent keeps the wbadmin transcript of each backup in `%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent\job-logs` on the domain controller. It keeps the newest 50; the **Job transcripts kept** setting changes the number. The transcript is also sent to the Manager, so you can usually read it in the **Transcript** section of the job's detail sheet on the **Backup Jobs** screen.

A restore keeps its own record and transcript on the domain controller, separate from the job transcripts. The restore run's page in the console shows the end of the transcript and gives the path of the full transcript on the machine. When a restore stops and needs an operator, the agent also leaves a note on the machine's disk.

After a full system recovery, if the restored system cannot apply its planned network settings, the DC agent writes what went wrong to `POST-RECOVERY-README.txt` on the machine and reports it once it connects.

Recovery environment output

The recovery agent runs inside the recovery image, from memory, so it has no log file of its own on the machine. It shows its progress on the machine's console and reports each phase to the Manager, where the restore run's page shows it. If a full system recovery fails, the machine stays at the recovery prompt with the output on screen, so you can read it there before you restart. If the Manager does not confirm the result, restart the machine by hand with `wpeutil reboot`.

Push installation logs

When a push installation of the DC agent fails, the Windows Installer log stays in `%SystemRoot%\Temp` on the target, and the deployment's result names its path. After a successful installation the log is removed. `cdr agent deploy --json` includes each target's full transcript.

Recovery image builds

The storage agent writes the DISM log of each recovery image build to the **Build working folder**, where you can collect the log of a failed build.

What to collect for support

When you contact Cygna Labs Corp. support, create a support package. It is one zip file with what support needs for a case, collected over the agents' own connections, so nobody has to sign in to each domain controller to gather files.

Create it in one of these ways:

- In the console, on the **Support package** card of the **License** screen. See Support package.
- With `cdr support package`. See Support Commands.
- If the Manager service does not start, with `CygnaDR.Manager.exe support package` on the Manager. See When the Manager service does not start below.

Creating, downloading, and removing a support package needs the `config.export` capability, held by the System Admin role and the break-glass account. Each is recorded in the audit trail.

What a support package contains

- **Manager service log**—The log files last written within the period you choose: by default the last 3 days, or everything still retained.
- **Manager database copy**—A copy of the Manager database with every credential removed (see below).
- **Host and configuration facts**—The Manager's build, operating system, time zone, CA fingerprint and whether the CA key has an escrow copy, database and snapshot details, free disk space, and the agents connected at the time with their versions. Also the Manager's own configuration values; a value whose name refers to a token, password, passphrase, secret, credential, or key is shown only as set or not set.
- **DC agent files**—From each connected DC agent: its service log, its backup job transcripts, and its state file.
- **Storage agent files**—From each connected storage agent: its service log and its state file. Never the vault, the staging area, or the OS media library.
- **Isolated Recovery Host agent files**—From each connected Isolated Recovery Host agent: its service log and its state file.

- **A description of the package**–A `README.txt`, and a `package.json` that records what each machine sent, what was left out and why, and what was removed from the database copy.

Each agent sends its newest files first, up to a limit per agent (512 MB unless you set another with `cdr support package --max-agent-mb`); anything over the limit or unreadable is left out and listed. A DC agent never sends the folder a restore uses on the domain controller, because it can hold a protected credential during a restore.

An agent that is not connected when you create the package is listed as offline. Collect that machine's log folder by hand, or create the package again once the agent is connected.

What the database copy leaves out

The Manager database holds the operator accounts and the Manager CA, so the package never carries the live database. It carries a copy with these removed:

- The Manager CA's private key and its escrow copy. The CA fingerprint is kept.
- Operator password hashes, and every session.
- Recovery enrollment tokens.
- Every secret setting, such as the SMTP password.
- The typed name of a failed sign-in that matches no operator, because it is often a password typed in the username field.

The copy is compacted after the removal, so the removed values are not left in its free space, and checked before the package is written. If the check fails, no package is written. Because the copy has no CA key, a support package cannot restore a Manager; use the database snapshots and the recovery passphrase for that (see Recovering the Manager).

Note: Important: A support package is not encrypted, and it still describes your forest in detail: domain controllers, addresses, shares, accounts named in logs, and the audit trail. Store and send it as you would your own network documentation, and delete it when the support case is closed.

Where packages are kept on the Manager

The Manager builds each package in the `support` folder beside its database, by default `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\support`, which is readable by SYSTEM and Administrators only. The file is named `cygnadr-support-<manager>-<yyyyMMdd-HHmmss>.zip`, with the time in UTC.

- Only one package can be collected at a time. Starting another while one is being collected is refused.

- A package is refused when the Manager's data disk has less than 2 GB free. Below 1 GB free, agents not yet started are skipped.
- A finished package is kept on the Manager for 24 hours, then deleted. Every package is also deleted when the Manager service starts.
- `cdr support package` removes the package from the Manager once it has saved it. After a download from the console, remove it yourself with **Remove from the manager**.

When the Manager service does not start

Run this command on the Manager, in an elevated command prompt in the Manager's program folder. It builds the Manager's part of a support package—its logs, the database copy with every credential removed, and host facts—without the service:

```
CygnaDR.Manager.exe support package [path] [--days N | --all-logs]
```

- `path`—The file to write. Default: the standard file name in the current folder. An existing file is not overwritten.
- `--days N`—Logs last written within this many days, from 1 to 366. Default: 3.
- `--all-logs`—Every log file still retained, whatever its age.

The package has no agent files, because there are no agent connections without the service. Collect the agents' log folders by hand if support needs them. The command is recorded in the audit trail against the Windows account that ran it. When the service is running, use `cdr support package` instead, which adds every agent's files.

Also include

With the support package, or instead of it if you cannot create one, send:

- The Cygna Forest Recovery version, shown in the account menu in the console and by `cdr --version`.
- If you cannot create a support package, the service log files from the Manager and from each machine involved, covering the time of the problem.
- For a backup problem, the job's error code and transcript from the **Backup Jobs** detail sheet, or the transcript from the DC's `job-logs` folder.
- For a restore problem, the restore run's page, and the restore transcript from the domain controller.
- The output of relevant commands with `--json`, such as `cdr agent list --json`, `cdr config show --agents --json`, `cdr alert list --json`, and `cdr restore show <run> --json`.
- Warnings and errors from the Application log on each machine involved.

Note: **Important:** Do not send the Manager database (`cygnadr.db`) itself or its snapshots unless support asks for them: they hold the operator accounts, their password hashes, and the Manager CA. The database copy in a support package has these removed, and is the copy to send. Do not send an installer log from an installation that was given `ENROLLMENTTOKEN`, because the log can contain the token.

Protecting the Manager

The Manager is what you use to run a recovery, so it has to survive whatever takes your directory down, or be easy to bring back. Your backups are safe on the storage machine whatever happens to the Manager, but without a working Manager nothing can plan or drive a restore from them. A little preparation on the day you install it decides whether bringing it back takes an hour or means visiting every domain controller.

What makes a Manager

Reinstalling the Manager and copying its database is not enough on its own. Four things identify a Manager, and a replacement needs all four:

What	Where it lives	Why it matters
The certificate authority	In the Manager database, with its private key protected by Windows data protection for that machine	Every agent was installed with this CA's fingerprint and trusts nothing else. A Manager with a different CA is refused by every agent.
The address	In each agent's installed configuration	Agents do not look the Manager up. They dial the address they were installed with.
The database	`%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\cygnadr.db`	Inventory, restore points, policies, plans, settings, operator accounts (the break-glass account included), and the audit trail.
The enrollment token	In the Manager's installed configuration in the registry, not in the database. A configuration backup carries it.	An agent the Manager does not recognize enrolls again with the token it kept. A Manager installed with a different token refuses them all.

The CA is the hard part. Because its key is protected for one machine, a copy of the database on new hardware carries a CA that cannot be used there. The Manager then refuses to start rather than create a new CA that every agent would refuse. The way through is the **recovery passphrase**, which you set at the first sign-in. The Manager keeps a second copy of the CA key encrypted with it, and on a new machine `CygnaDR.Manager.exe ca recover` uses that copy to make the CA usable again.

The Manager also sends a **configuration backup** to the storage machine every day: its database together with the enrollment token and the secret settings, such as the SMTP password, that a database copy cannot carry to another machine. It is encrypted, and only the recovery passphrase opens it. With a configuration backup and the passphrase, a new Manager starts with the same CA, and agents reconnect without being touched. See Manager Configuration Backups.



Ways back

You have	What to do	Effect on agents
A backup of the whole Manager machine	Restore the machine	None; they reconnect on their own.
A configuration backup and the recovery passphrase	Build a new Manager and import the backup	None, if the new Manager has the old address.
A database snapshot and the recovery passphrase	Build a new Manager from the snapshot	None, if the new Manager has the old address.
A database snapshot	Build from the	Every agent must be re-pointed at the

You have	What to do	Effect on agents
but no recovery passphrase	snapshot and re-key the CA	new CA.
Nothing	Install a new Manager	Every agent must be re-pointed. History and the restore-point index are lost; the backups remain on the storage machine.
A Manager that was compromised	Replace it with a re-keyed Manager and a new enrollment token	Every agent must be re-pointed and given the new token.

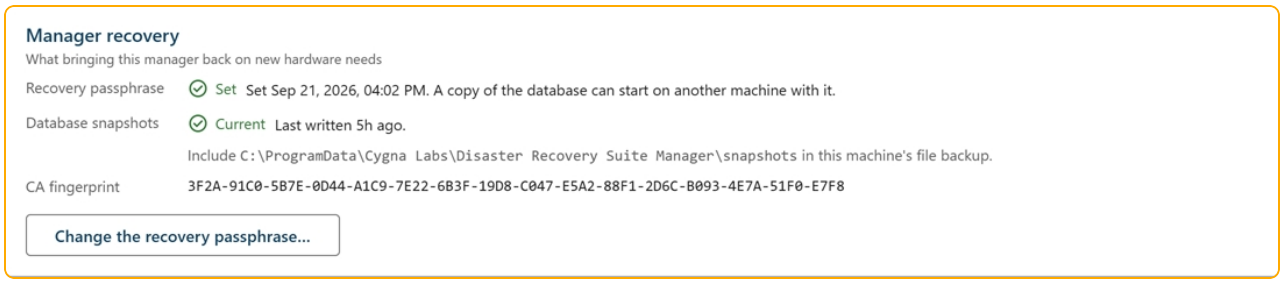
Whichever way you go, anything that happened after the backup or snapshot you restore from is missing from the restored Manager, including backups taken since. Afterwards, take a fresh backup of every domain controller you back up.

A support package is not a way back. Its copy of the Manager database has the CA key, password hashes, and other credentials removed, so it cannot start or restore a Manager. Use a file-level backup, a configuration backup, or a database snapshot with the recovery passphrase.

Watching the Manager's readiness

The **Manager recovery** card, at the foot of the **License** screen (**Administration** > **License**), shows whether the recovery passphrase is set, when the last database snapshot was written (it is marked **Stale** after two days), and the CA fingerprint. Check it after installation and after any change to the Manager.

Below it, the **Configuration backups** card shows whether configuration backups are reaching the storage machine, how many each storage machine holds, and the folder they are in. It is marked **Stale** when the newest backup is more than twice the backup interval old.



In this chapter

- Manager Day-One Checklist
- Manager Configuration Backups
- Recovering the Manager
- Re-Keying the Manager CA

Manager Day-One Checklist

Do these on the day you install the Manager, and check them again after any change to it. Each item is something that cannot be recovered from the Manager once it is gone. For why each one matters, see Protecting the Manager.

Store these offline

Keep these together, offline, somewhere that does not depend on your directory, such as a safe:

Item	Where to get it
The break-glass username and password	Shown once at the first sign-in. The username is `breakglass`. If you change the password later, update the record in the same sitting. If you turn on multi-factor authentication for the account, store its recovery codes with it. See Multi-Factor Authentication.
The recovery passphrase	Set at the first sign-in. It opens the escrow copy of the CA key and every configuration backup. If you change it later on the **Manager recovery** card, copies made from then on need the new one, and copies made before still open with the old one. Replace the record in the same sitting.
The enrollment token	The value you gave `ENROLLMENTTOKEN` when you installed the Manager. It is not in the database, so no snapshot brings it back, and a Manager rebuilt from a snapshot must be installed with the same token for agents to enroll again. A configuration backup carries it.
The Manager's IP address	The address every agent was installed with.
The storage volume's BitLocker recovery key	Saved when you turn on BitLocker for the storage volume. The storage machine is not domain-joined, so the key is not in Active Directory. See Encrypting the vault volume.

Item	Where to get it
The license key	The key Cygna sent you. The Manager keeps it in its database, so a Manager recovered from a snapshot keeps its license. A Manager rebuilt without a snapshot refuses backups and recovery operations until the key is applied again. See Licensing.
The SMTP password, if the Manager sends email	The password of the **SMTP user** in the **Email** settings. The Manager never shows it again once saved, and a database moved to another machine cannot read it, so you enter it again after recovering the Manager from a snapshot. A configuration backup carries it. Until it is entered, recovery plans send no notifications and no report email.

Keep with your recovery notes

The Manager CA fingerprint is public, so it does not belong in the safe. Keep a copy with your recovery notes, such as your runbook. After you recover the Manager, you compare its fingerprint with a copy that did not come from the recovered Manager itself.

The fingerprint is shown on the **Agents** screen and the **Manager recovery** card, by `cdr trust list`, and in the Application log at every Manager start. Every agent also keeps the fingerprint it was installed with, so a surviving agent machine gives you a second copy. See The fingerprint to compare against.

Checklist

1. **Store the items above offline.**
2. **Confirm the license is applied.** On **Administration** > **License**, the status should be **Licensed**, or **Ending soon** for a short trial. Without a license, the Manager refuses every backup. To be told by email before it ends, set **Expiry warning recipients**. See Licensing.
3. **Turn on alert email.** Set **Recipients** in the **Alert notifications** settings, then select **Send test email** on the **Alerts** screen. See Alerts Reference.
4. **Confirm the recovery passphrase is set.** At the foot of the **License** screen (**Administration** > **License**), the **Manager recovery** card shows **Set** with a date. If it shows **Not set**, a copy of the database cannot bring the Manager back on another machine. Select **Set the recovery passphrase…**, which needs the System Admin role. The passphrase must be at least 16 characters; a few unrelated words are easier to keep than a short complex one.

5. **Confirm configuration backups reach the storage machine.** Once the passphrase is set, the Manager sends an encrypted configuration backup to the storage machine every day, the first a few minutes after the service starts. On the **License** screen, the **Configuration backups** card should show **Current** and the folder on the storage machine that holds them. Note the folder in your recovery notes. See Manager Configuration Backups.
6. **Back up the database snapshots.** The Manager writes a consistent copy of its database every day to `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager\snapshots` and keeps the newest seven. The first is written a few minutes after the service starts. Include this folder in the Manager machine's file backup, and protect the backup as you would the database: a snapshot holds the operator accounts and, with the passphrase, the CA.
7. **Do not back up the live database file.** `cygnadr.db` is written continuously, so a file copy of it can be inconsistent or miss recent changes. Back up the snapshots instead.
8. **Back up the whole Manager machine, if your platform can.** A virtual machine backup or snapshot, or a bare-metal backup, is the simplest way back.
9. **Keep the Manager's address stable.** Give it a static IP address or a reserved DHCP lease. A replacement Manager that can take the same address needs nothing done to any agent.
10. **Create named operators.** Keep the break-glass account for emergencies. See Managing Operators and Roles.

Taking a snapshot on demand

Take an extra snapshot before an upgrade or any other change to the Manager. In an elevated command prompt on the Manager, in its program folder, run:

```
CygnaDR.Manager.exe db snapshot
```

The snapshot is written to the snapshots folder, and the command prints its path. To write it somewhere else, give a path: `CygnaDR.Manager.exe db snapshot D:\ManagerBackup\before-upgrade.db`. This command is safe to run while the service is running.

Manager Configuration Backups

The Manager sends an encrypted copy of itself to the storage machine every day. With one of these configuration backups and the recovery passphrase, you can build a new Manager on new hardware, and every agent reconnects to it without being touched. When the Manager machine is gone and you have no backup of the whole machine, it is the simplest way back.

What a configuration backup holds

- **The Manager database**—Inventory, the restore-point index, policies, recovery plans, settings, operator accounts and roles, the audit trail, and the certificate authority with its escrow copy.
- **The enrollment token**—The token the Manager was installed with. It lives in the registry, not in the database, so a database snapshot cannot bring it back.
- **Secret settings**—Such as the SMTP password. They are protected for the machine that saved them, so a copy of the database cannot carry them to another machine.
- **Multi-factor authentication enrollments**—Each operator's enrollment, so operators keep using their authenticator app on a Manager built from the backup.

The backup is encrypted, and only the recovery passphrase that was current when it was written opens it. The storage machine keeps the file but cannot read it. Each backup is roughly the size of the Manager's database, compressed, and is named `cygnadr-config-<yyyyMMdd-HHmmss>.cdrbundle`, with the time it was written in UTC.

The Manager builds each backup in a work folder beside its database and removes it once it has been sent, so no copy stays on the Manager.

Where backups are kept

The storage machine keeps the copies, in `<StorageRoot>\manager-backups` beside the vault. The storage agent creates the folder when it starts and gives it the vault's permissions. It is never shared and is not under the staging folder that domain controllers write to. To open it, sign in to the storage machine with an account in the CygnaDR Administrators group.

Only the storage machine receives configuration backups. A candidate storage agent, which anyone holding the enrollment token could have enrolled, is sent none, because a configuration backup carries the Manager's CA key and every operator account. When the storage machine is replaced, a configuration backup is sent to the new one straight away. A replaced machine keeps the copies it already holds until you wipe it. See Moving the Storage Machine or an Isolated Recovery Host.

Schedule and retention

- **Schedule**—The Manager sends a backup every 24 hours by default, set by **Back up this manager's configuration every**. Set it to 0 to turn the schedule off; **Back up now** still works.
- **Start**—Backups start once the recovery passphrase is set. The first is sent a few minutes after the Manager service starts.

- **Retries**–A scheduled backup that fails is tried again an hour later. A storage machine that was offline receives one soon after it reconnects.
- **New passphrase**–Setting a new recovery passphrase starts a backup straight away, so that a copy under the new passphrase exists at once.
- **How many are kept**–**Configuration backups to keep** tells the storage machine how many to keep, 14 by default. The oldest is removed when a new one is stored.

For the defaults and ranges, see Settings Reference.

The storage machine decides what is deleted

The history of configuration backups exists for the day the Manager is compromised, so the Manager cannot delete it. The Manager only says how many to keep. The storage machine applies its own limits, which are set on that machine and cannot be changed from the console:

- **Configuration backups always kept**–It never keeps fewer than this many, whatever the Manager asks.
- **Configuration backups never pruned before**–It never deletes a backup younger than this.
- **Configuration backups folder limit**–It refuses a backup that would take the folder past this size, and any backup that would leave less than the vault's **Free space reserve** free on the volume.

The storage machine removes the oldest backups by its own file times, never overwrites a backup it holds, and never touches a file in the folder that is not named like a configuration backup. To delete configuration backups yourself, delete the files on the storage machine.

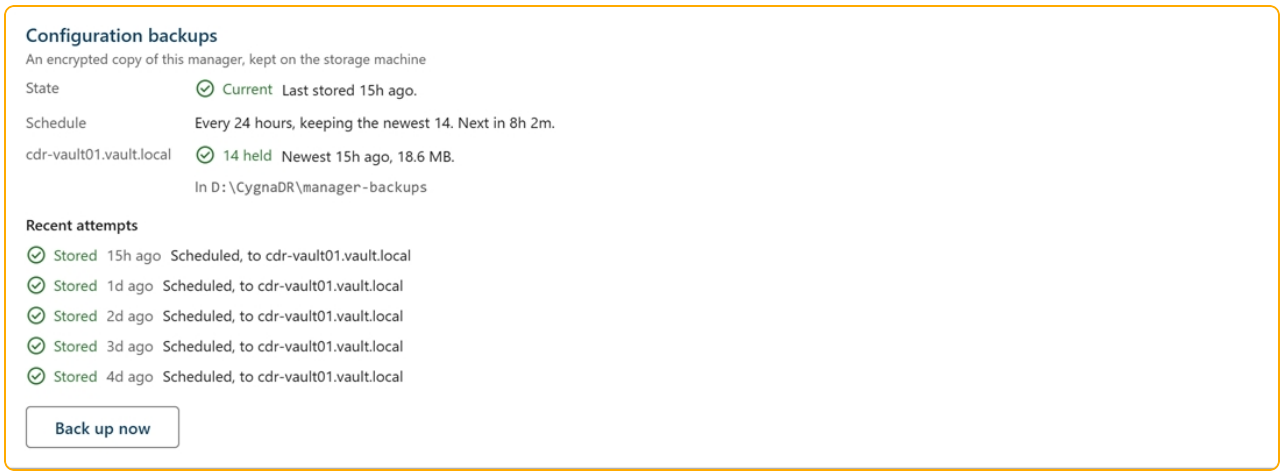
Checking configuration backups

At the foot of the **License** screen (**Administration** > **License**), the **Configuration backups** card, below the **Manager recovery** card, shows whether the Manager is being backed up and where the copies are. `cdr config backup` shows the same; see Configuration and Exclusion Commands.

Row	Shows
State	**Current** with when the last backup was stored; **Stale** when the newest is more than twice the interval old; **None yet**; **Running**; **Last attempt failed**; or **Not running**, with the reason, for example that the recovery passphrase is not set.
Schedule	How often a backup is sent, how many are kept, and when the next is due.

Row	Shows
Each storage machine, by name	How many backups it holds, the newest one's age and size, and the folder they are in. **None yet** if it holds none, **Offline** if it is not connected.
Earlier passphrase	Shown when the storage machine holds backups written before the recovery passphrase was last set. They still open with the previous passphrase. See After you change the recovery passphrase.

Under **Recent attempts**, the card lists the last five attempts: **Stored**, **Running**, **Refused**, or **Failed**, when, whether it was scheduled or started by an operator, the storage machine, and why any attempt did not store. Every attempt is also recorded in the audit trail.



Check the card on the day you install the Manager, and after any change to the Manager or the storage machine.

Taking a backup now

Take one before an upgrade, or after a change you want captured at once. You need the `config.export` capability, held by the System Admin role and the break-glass account.

To take a configuration backup now:

1. Select **Administration** > **License**.
2. On the **Configuration backups** card, select **Back up now**.

   The button shows **Backing up…** while the Manager writes the backup and sends it, and the card updates when the storage machine has answered. Its outcome appears under **Recent attempts**. The button is unavailable until the recovery passphrase is set, and while a backup is running.

From the command line, run `cdr config backup run`, which waits until the storage machine has stored the backup.

After you change the recovery passphrase

A new recovery passphrase protects only what is written from then on. Every configuration backup and database snapshot written before the change still opens with the previous passphrase, and each holds the CA key; a configuration backup also holds the enrollment token and the secret settings. The **Earlier passphrase** row on the card counts the configuration backups that open only with an earlier passphrase.

If the previous passphrase may have been seen by someone who can also reach those copies, delete them:

- On each storage machine, the files in `manager-backups` written before the change.
- On the Manager, the older files in the `snapshots` folder, and any copies in your file backups.

Deleting is done on the storage machine on purpose: the Manager cannot delete its own backups there, so a compromised Manager cannot either. If the CA key itself may have been exposed, a new passphrase is not enough; replace the Manager. See Re-Keying the Manager CA.

When a backup is not stored

The card and `cdr config backup` give the reason in full. The most common reasons are:

- **The recovery passphrase is not set**–Without it nobody could ever open a backup. Set it on the **Manager recovery** card.
- **No storage agent is connected**–A backup is sent when it reconnects.
- **The Manager's data volume is short of space**–Building a backup needs about twice the database's size plus 1 GB free on the volume that holds the database. Free some space there.
- **The storage machine refused it**–The folder limit or the vault's free space reserve would be passed, or the `manager-backups` folder is missing. Raise the limit on the storage machine or let older backups age out, free space on the volume, or restart the storage agent so that it creates the folder with the right permissions.
- **The database was moved here by hand**–The key that encrypts backups was protected for the old machine. Run `CygnaDR.Manager.exe ca recover`, or set the recovery passphrase again.

To rebuild a Manager from a configuration backup, see Building a new Manager from a configuration backup.

Recovering the Manager

Use this topic when the Manager machine has failed, been encrypted, or been rebuilt, and you need a working Manager again. If the Manager was compromised, do not follow it: a compromised Manager is replaced with a new CA, not recovered. See Re-Keying the Manager CA.

Before you begin

- Have the offline record from the Manager Day-One Checklist: the recovery passphrase, the enrollment token, and the break-glass credential.
- Have the CA fingerprint the agents trust, from your recovery notes or from a surviving agent. See The fingerprint to compare against.
- Choose the route from what you have. A backup of the whole Manager machine is simplest. Without one, build a new Manager from a configuration backup on the storage machine, which brings back everything a database snapshot does, plus the enrollment token and the secret settings. A configuration backup and a database snapshot both need the recovery passphrase; without it, re-key the CA instead.

Note: Important: Sign in as soon as the Manager service starts on a new installation. A Manager that starts on an empty database offers its one-time setup to whoever reaches the console first.

The fingerprint to compare against

A recovered Manager must have the same CA as before, or every agent refuses it. To check, you compare the recovered Manager's fingerprint with one that did not come from the recovered Manager: the copy in your recovery notes, or the fingerprint a surviving agent was installed with.

Each agent keeps the fingerprint it trusts in the `ManagerCaFingerprint` value under its registry key:

- **Storage agent**—`HKLM\SOFTWARE\Cygna Labs\Disaster Recovery Suite\Storage Agent\Configuration\CygnaDR`
- **DC agent**—`HKLM\SOFTWARE\Cygna Labs\Disaster Recovery Suite\DC Agent\Configuration\CygnaDR`
- **Isolated Recovery Host agent**—`HKLM\SOFTWARE\Cygna Labs\Disaster Recovery Suite\Isolated Recovery Host Agent\Configuration\CygnaDR`

The storage machine is usually the best source, because it is outside the domain and is the machine most likely to survive. For example, on the storage machine:

```
Get-ItemPropertyValue -Path 'HKLM:\SOFTWARE\Cygna Labs\Disaster Recovery Suite\Storage Agent\Configuration\CygnaDR' -Name ManagerCaFingerprint
```

The stored value is written as it was given at installation, so it may use colons or spaces instead of hyphens. Compare fingerprints ignoring letter case and the separators between the groups of digits.

Restoring the whole machine

To recover the Manager from a whole-machine backup:

1. Restore the Manager machine or virtual machine from its backup, and start it.

   The CA, the address, the database, and the enrollment token all come back with the machine, so the Manager starts as it was.

2. Sign in to the console, and select **Administration** > **Agents**.

   Agents reconnect on their own within a few minutes.

3. Start a backup of every domain controller you back up.

   See After any Manager recovery.

Building a new Manager from a configuration backup

You need a configuration backup from the storage machine, the recovery passphrase that was current when it was written, and the Manager package of the same version as the old Manager or newer. For what a configuration backup holds, see Manager Configuration Backups.

To build a new Manager from a configuration backup:

1. On the storage machine, sign in with an account in the CygnaDR Administrators group, and copy the newest backup from `<StorageRoot>\manager-backups`.

   Backups are named `cygnadr-config-<yyyyMMdd-HHmmss>.cdrbundle`, with the time they were written in UTC. If the storage root is not where you expect, use the folder that the old Manager's **Configuration backups** card and `cdr config backup` showed, if you recorded it. If you have changed the recovery passphrase since, choose a backup written under the passphrase you have.

2. Build the machine and give it the old Manager's IP address.

   If it cannot have the old address, carry on, and re-point the agents afterwards as described in If the new Manager has a different address.

3. Install the Manager:

```
msiexec /i CygnaDR.Manager.msi /qn ENROLLMENTTOKEN=<token>
```

Use the old Manager's enrollment token if you have it. If you do not, give any token: the import puts the old one back. If the old Manager used values other than the defaults for `RESTPORT`, `GRPCPORT`, or `LISTENADDRESS`, give the same values again. Do not open the console.

4. Stop the Manager service:

```
Stop-Service CygnaDRManager
```

5. Copy the backup file to the new Manager, open an elevated command prompt in the Manager's program folder, `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager`, and check the file:

```
CygnaDR.Manager.exe bundle inspect <file.cdrbundle>
```

The command needs no passphrase. It shows when the backup was written, by which Manager and version, and its CA fingerprint, which must match the one the agents trust. See The fingerprint to compare against.

6. Import the backup:

```
CygnaDR.Manager.exe bundle import <file.cdrbundle>
```

The command shows the backup again and the database it is about to replace. It refuses a backup written by a newer Manager version; install that version or newer first.

7. Enter the recovery passphrase when asked.

   You have three tries, and a wrong passphrase changes nothing.

8. Type `IMPORT` to continue.

   The import keeps the new installation's database beside it as `cygnadr.db.before-import-<time>`, puts the backup's database in place, protects the CA key for this machine, restores the secret settings, writes the enrollment token back if this installation's differs, and records the import in the audit trail with the Windows account that ran it. It then prints what it did.

   If it reports that the CA could not be unlocked with this passphrase, the database's escrow copy was set under a different passphrase. Run `CygnaDR.Manager.exe ca recover` with that passphrase before you start the service. If it lists settings that were not in the backup, enter them again on the **Configuration** screen after you sign in.

9. Start the Manager service:

```
Start-Service CygnaDRManager
```

The Application log records the CA fingerprint at every start. It must match the one in step 5.

10. Sign in with a named account or the break-glass account.

    Accounts, roles, settings, and backup schedules are as they were when the backup was written. The SMTP password came back with the backup, so you do not need to enter it again. Operators who use multi-factor authentication sign in with their authenticator app as before: the backup carries their MFA enrollment.

11. Select **Administration** > **Agents**, and wait for the agents to reconnect.

    Agents reconnect within a few minutes with no change on their side. An agent that enrolled after the backup enrolls again with its kept token and appears after one more reconnect.

12. Start a backup of every domain controller you back up.

13. On the **License** screen, check that the **Configuration backups** card shows a new backup stored within the day.

Building a new Manager from a snapshot

Use this route when you have a database snapshot but no configuration backup. You need a database snapshot file (`cygnadr-snapshot-*.db`, from the snapshots folder in your file backup), the recovery passphrase, the enrollment token, and the Manager package of the same version as the old Manager or newer.

To build a new Manager from a database snapshot:

1. Build the machine and give it the old Manager's IP address.

   If it cannot have the old address, carry on, and re-point the agents afterwards as described in If the new Manager has a different address.

2. Install the Manager with the old Manager's enrollment token:

   ```
   msiexec /i CygnaDR.Manager.msi /qn ENROLLMENTTOKEN=<token>
   ```

   A new installation uses the default `RESTPORT`, `GRPCPORT`, and `LISTENADDRESS`. If the old Manager used other values, give the same values again. The service starts on an empty database and creates a CA of its own, which you are about to replace. Do not open the console.

3. Stop the Manager service:

```
Stop-Service CygnaDRManager
```

4. In `%ProgramData%\Cygna Labs\Disaster Recovery Suite Manager`, delete `cygnadr.db`, `cygnadr.db-wal`, and `cygnadr.db-shm`.

   No `-wal` or `-shm` file from the new installation may remain beside the snapshot.

5. Copy the snapshot into the same folder and rename it `cygnadr.db`.

6. Open an elevated command prompt in the Manager's program folder, `%ProgramFiles%\Cygna Labs\Disaster Recovery Suite Manager`, and check the CA:

   ```
   CygnaDR.Manager.exe ca status
   ```

   Expect `Usable here: NO` and `Escrow copy: yes`. The fingerprint shown must match the one the agents trust. If `Escrow copy` is `NO`, the recovery passphrase was never set for this database: re-key instead.

7. Recover the CA:

   ```
   CygnaDR.Manager.exe ca recover
   ```

   Enter the recovery passphrase when asked. You have three tries per run, and a wrong passphrase changes nothing. On success the CA key is protected for this machine, and an audit entry names the Windows account that ran the command.

8. Start the Manager service:

   ```
   Start-Service CygnaDRManager
   ```

   The Application log records the CA fingerprint at every start. It must match the one in step 6.

9. Sign in with a named account or the break-glass account.

   Accounts, roles, and passwords are as they were when the snapshot was taken.

   Multi-factor authentication enrollments are protected for the old machine and cannot be read here. An operator who used MFA signs in with one of their recovery codes, then turns MFA off and sets it up again; without a recovery code, an administrator resets their MFA. The break-glass account signs in with its password alone, and MFA is turned off for it, with an entry in the audit trail. Turn it on again from the account menu. See Multi-Factor Authentication.

10. If the Manager sends email, enter the **SMTP password** again: select **Administration** > **Configuration**, and on the **Manager** tab, in the **Email** group, enter the password and

save. Or use `cdr config set CygnaDR:Email:SmtpPassword`, which prompts for it.

Passwords kept in settings are protected for the old machine, and `ca recover` carries over only the CA key. On the new machine the password reads as not set, and the Manager logs that it needs entering again. Until you enter it, recovery plans send no notifications and no report email.

11. Select **Administration** > **Agents**, and wait for the agents to reconnect.

    Agents reconnect within a few minutes with no change on their side. An agent that enrolled after the snapshot is unknown to the restored database: the Manager refuses its session, and the agent enrolls again with its kept token and appears after one more reconnect. Check the **Repository** screen to confirm the storage agent has reconnected. If you use an Isolated Recovery Host, check that its panel on the **Agents** screen shows it connected and **Ready**; scheduled test restores resume once it is.

12. Start a backup of every domain controller you back up.

13. If there is any doubt about who has seen the recovery passphrase, set a new one on the **Manager recovery** card on the **License** screen, and replace the offline record in the same sitting.

    Copies written before the change still open with the old passphrase. See After you change the recovery passphrase.

If the new Manager has a different address

Agents trust the CA, not the address; the address is only where they look. A Manager on a new address with the recovered CA is trusted as soon as agents can reach it, so they need only the new address:

- **DC agents**–Run `cdr agent deploy` with `--repoint --force --manager-address https://<new address>:5211`. This keeps their fingerprint. See Maintaining Agents.
- **The storage agent, the Isolated Recovery Host agent, and agents installed by hand**–Reinstall with the new `MANAGERADDRESS`.
- **Operators**–Use the new address in the browser and with `cdr --manager`. `cdr` asks you to confirm the fingerprint for the new address; it should be the one the agents trust.

After any Manager recovery

- **Take a fresh backup of every domain controller you back up.** Backups that completed after the machine backup, configuration backup, or snapshot are still in the vault, but the restored Manager does not know about them and cannot restore from them. Fresh backups give it current restore points.

- **Check the Manager recovery card.** It should show the passphrase as set and a snapshot written within the day. On a new machine the first snapshot is written a few minutes after the service starts.
- **Check the Configuration backups card.** It should show a backup stored on the storage machine. On a new machine the first is sent a few minutes after the service starts.
- **Review the audit trail.** Nothing that happened between the backup or snapshot and the failure is in it.
- **Review anything else changed since the backup or snapshot.** Policies, settings, operators, and recovery plans are as they were when it was taken.
- **Update your records.** Update the offline record if the address or the passphrase changed, and your recovery notes if the fingerprint changed.

Re-Keying the Manager CA

Re-keying gives the Manager a new certificate authority. Every agent in your estate trusts the old CA's fingerprint, so after a re-key every agent refuses the Manager until it is re-pointed at the new fingerprint. That makes re-keying the last resort for a lost Manager, and the required step for a compromised one.

When to re-key

- **You have a database snapshot but not the recovery passphrase**, so the CA in the snapshot cannot be recovered on a new machine. A configuration backup cannot be opened at all without the passphrase.
- **You have no database at all**, and are installing a new Manager from scratch. A fresh installation creates its own CA, which amounts to a re-key.
- **The Manager was compromised**: an attacker had control of it, or its CA key, database, or enrollment token may have been read. Re-key even if you have the recovery passphrase, because recovering the old CA would put a key the attacker may hold back into service.

If you have a configuration backup or a snapshot and the passphrase, and the Manager was not compromised, recover the Manager instead; agents then need nothing. See Recovering the Manager.

Re-keying a Manager database

To re-key the CA in a Manager database:

1. Build the Manager machine from a snapshot, following steps 1 to 5 of building a new Manager in Recovering the Manager.

   Leave the service stopped.

2. In an elevated command prompt in the Manager's program folder, run:

   ```
   CygnaDR.Manager.exe ca rekey
   ```

   The command warns you if the current CA is still usable on this machine or the database holds an escrow copy. In that case you probably want `ca recover` instead.

3. Type `REKEY` to continue.

   The CA is replaced, and an audit entry records both fingerprints.

4. Enter a new recovery passphrase, twice, when asked.

   Store it offline with the break-glass credential. Configuration backups resume under the new passphrase once the service starts.

5. Start the Manager service, and sign in.

6. If the Manager sends email, enter the **SMTP password** again in the **Email** group on the **Manager** tab of the **Configuration** screen.

   A password kept in settings cannot be read on a machine other than the one that saved it. Until you enter it, recovery plans send no notifications and no report email.

7. Select **Administration** > **Agents**, and read the new CA fingerprint.

   Replace the copy in your recovery notes.

8. Re-point every agent, as described below.

With no database at all, install the Manager, sign in at once, and complete the setup. The new Manager has a new CA, no inventory, no restore points, and no history; the backups are still on the storage machine, but the Manager does not know about them. Then re-point every agent.

Re-pointing the fleet

Until an agent is re-pointed, it refuses the re-keyed Manager. When it is re-pointed, its old certificate is refused too, and it enrolls again with its enrollment token; you do not need to delete certificates by hand.

- **DC agents**–From a machine where `cdr` trusts the re-keyed Manager, run `cdr agent deploy` with `--repoint --force` and `--manager-address` set to the Manager's agent

address (TCP 5211 by default). The deployment pins the fingerprint of the Manager that `cdr` trusts. If the Manager has a new enrollment token, give it in `CYGNADR_ENROLLMENT_TOKEN` or when prompted, and the deployment writes it to every agent too. Name the domain controllers, or use `--all`, which also installs the agent on any domain controller that has none:

```
cdr agent deploy --all --repoint --force --manager-address https://10.0.0.10:5211 --user CORP\admin --package CygnaDR.DcAgent.msi
```

- **The storage agent, and agents installed by hand**–Reinstall on the machine with the new fingerprint, and the new address and enrollment token if they changed:

```
msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERCAFINGERPRINT=<new fingerprint>
```

- **cdr users**–Each operator runs `cdr trust remove --manager <address>`, connects again, and confirms the new fingerprint.
- **Browsers**–Operators trust the new Manager CA on their workstations. See Trusting the Manager Certificate.

The console's install and update actions do not re-point agents, so they are not a substitute for these steps. For details, see Maintaining Agents.

Replacing a compromised Manager

A compromised Manager's identity can no longer be trusted, so it is replaced, never recovered. Anyone holding a copy of its CA key could pose as the Manager to every agent.

To replace a compromised Manager:

1. Take the old Manager off the network, and keep its disk for investigation.

   Do not start it on the network again, even to copy something off it.

2. Choose the database to start from.

   The best choice is a snapshot from before the compromise began. A snapshot taken during the compromise brings back whatever the attacker changed, including operator accounts, roles, settings, exclusions, and plans. If no snapshot can be trusted, start from a fresh installation.

3. Build the new Manager machine. Giving it the old address saves re-addressing the agents.

4. Install the Manager with a new enrollment token, and store the new token offline in place of the old one.

5. If you are using a snapshot, put it in place as for any snapshot recovery, then run `CygnaDR.Manager.exe ca rekey`, not `ca recover`, and set the new recovery

passphrase it asks for.

6. Start the service and sign in with the break-glass account.
7. On the **Operators & Roles** screen, reset the password of every named operator, and disable any account you cannot account for.
8. Change the break-glass password from the account menu, and update the offline record.
9. Read the new CA fingerprint on the **Agents** screen, and replace the copy in your recovery notes.
10. Re-point every agent at the new fingerprint, as described in Re-pointing the fleet, and give every agent the new enrollment token.

    The re-pointing deployment writes the new token to every DC agent. Reinstall the storage agent, any Isolated Recovery Host agent, and any agent installed by hand on its machine with the new token, for example `msiexec /i CygnaDR.StorageAgent.msi /qn MANAGERCAFINGERPRINT=<new fingerprint> ENROLLMENTTOKEN=<new token>`. Until an agent is re-pointed, it still trusts the old CA and does not work with the new Manager, so re-point every agent, and do it before the backups in the next step.
11. Start a backup of every domain controller you back up, once its agent is re-pointed.

    These are the first backups whose agent trusts the new Manager.
12. Have every `cdr` user remove the old trust and confirm the new fingerprint.
13. Once the **Configuration backups** card on the **License** screen shows a backup from the new Manager, delete the older configuration backups on the storage machine.

    The files in `<StorageRoot>\manager-backups` written before the replacement hold the old CA key, the old enrollment token, and the old secret settings, and they open with the old recovery passphrase. The Manager cannot delete them, so this is done on the storage machine.

Restoring backups taken before a re-key

After a compromise, the backups worth restoring are usually ones taken before the re-key, and the DC agent inside each of them still trusts the old CA. Preflight reports this with `agent.ca-changed`:

- It does not block an ordinary restore. The directory comes back, but the restored agent refuses the new Manager until you re-point it. Re-point it with `cdr agent deploy <DC> --repoint --force --manager-address <address>` as soon as the machine is on the network. The run waits in validation until the agent connects. The restored agent is then asked to check its domain controller, and the run completes or fails on its answer.
- It blocks approval when the restore plans network settings for the restored system, because the restored agent would refuse the settings the new Manager signs. Plan the

restored system's network as unchanged, and set the address by hand before you re-point the agent.

Note: **Important:** A DC restored from a backup taken before the re-key trusts the old CA from the moment it starts until you re-point it. Keep restored domain controllers on the isolated recovery network until they are re-pointed.

Troubleshooting

Cygna Forest Recovery is built to say what went wrong and what to do about it. Every refusal, failed job, and preflight check carries a code and a sentence that names the cause, and most of them name the fix. Start with what the console tells you, and go to log files only when it does not say enough.

General approach

1. **Read the message in full.** A backup job's **What went wrong**, a preflight check's remediation, and an alert's advice are written to be acted on. The code beside each one is stable and is what this chapter is organized by.
2. **Check that every component is connected.** Most failures follow from an agent that is not connected or a storage agent that cannot be reached. The **Agents** screen shows the connection state and last heartbeat of every agent.
3. **Look at the transcript.** Backups and restores keep transcripts, which the console shows: on the **Backup Jobs** detail sheet, and under **What the machine reported** on a restore run's page.
4. **Read the service log** of the component involved, around the time of the problem. See Logs and Diagnostics.
5. **Check the audit trail** for what changed: a setting, a policy, a role. See Auditing Operator Activity.
6. **Before you contact support, create a support package.** See Collecting a support package below.

Where to look

Problem	Start with
A backup failed or did not run	The job on the **Backup Jobs** screen, and Backup Job Errors
A restore or recovery plan cannot be approved	The **Preflight findings** card on a recovery plan's page, or the **Preflight** card on a restore run's page, and Preflight Check Reference
A restore stopped or failed	The outcome and **What the machine reported** on the restore run's page, then the restore transcript on the domain controller

Problem	Start with
An agent does not appear, or shows as not connected	The agent's service log, and Agent Enrollment and Connectivity
A push installation failed	The deployment's per-target result, and Agent Enrollment and Connectivity
An installer refused to install	Installer Messages
An alert is raised	The alert's advice on the **Alerts** screen, and Alerts Reference
Full system recovery is unavailable for a domain controller	The **Recovery image** column on the **OS Media** screen
The Manager is gone or will not start	Recovering the Manager

Using the command line

The `cdr` command-line tool shows the same information as the console, and adds detail that is useful when diagnosing. `cdr jobs list --failed --detail` adds the staging path, the agent log path, and the full error to each failed job. `cdr agent list --offline` and `--silent-for <minutes>` find agents that have gone quiet. `cdr config show --agents --behind` lists machines that are not running the current configuration. `--json` on any command gives output you can save or send to support.

`cdr` also uses exit codes you can act on: 2 means the Manager could not be reached, 9 that its CA is not trusted, 10 that you are not signed in, and 3 that the Manager refused the request, which includes an operator lacking the capability. See Command-Line Reference.

Collecting a support package

When you contact Cygna Labs Corp. support, send a support package. It collects the Manager's log, a copy of the Manager database with every credential removed, host and configuration facts, and the logs, job transcripts, and state files of every connected agent, into one zip file. You need the `config.export` capability (System Admin).

- In the console, use the **Support package** card on the **License** screen. See Support package.
- From a command prompt, run `cdr support package`. See Support Commands.

- If the Manager service does not start, run `CygnaDR.Manager.exe support package` on the Manager as an administrator. It collects the Manager's part only.

The package is customer confidential and not encrypted. For what it contains and leaves out, how long the Manager keeps it, and what else to send, see Logs and Diagnostics.

In this chapter

- Preflight Check Reference
- Backup Job Errors
- Agent Enrollment and Connectivity
- Installer Messages

Preflight Check Reference

Before a restore or recovery plan can be approved, the Manager checks it against the backup, the target machine, and the directory. Each finding is a gate with a code, a title, the machine or object it is about, and what to do. This topic lists every gate by code.

Each gate is one of three kinds:

- **Blocks approval**–The Manager refuses approval while the gate stands. Fix the cause, or choose another restore point or target, and the gate clears.
- **Needs your acceptance**–The Manager refuses approval until the approver accepts the gate by its code, or the run is cancelled. These gates report something the Manager could not prove from where it looked, which the restore may still get past from where it runs: for example, a name the machine cannot resolve now that the recovery environment's network may resolve. A DC state recovery also raises two gates of this kind that are not about the environment but about what its authoritative rollback does across the domain. Accept them only if that is what you mean to do. The acceptance is recorded on the run and in the audit trail, and holds until that run is executed. In the console, select the gate's acceptance check box on the run's page before you approve. With `cdr`, pass `--accept <GATE>` to `cdr restore approve`. Only restore runs raise these gates. When a forest or domain plan starts a restore, the restore run raises them, and the plan waits until someone accepts them on that run or cancels it.
- **Advisory**–Read, not cleared. The restore is possible; the gate says what it costs or what to watch for.

Gates are evaluated when a restore is planned, and again when it is approved and executed. For a full system recovery, the Manager also asks the target machine to describe itself again at each of those points. A forest or domain plan has its own gates, listed at the end of this topic, and each restore it starts is checked again against the restore gates. For how preflight fits into a recovery, see Recovery Lifecycle.

Some checks run only at execution, and do not appear as gates: the backup's hash is checked against the vault before the read credential is issued, the version is confirmed present at the target with `wbadmin get versions`, the machine's boot configuration is proven settable before it is set for real, the recovery environment checks its own firmware and disks against the backup before it writes, and secure channel and replication health are checked after the restore.

Executing a restore run or a recovery plan also needs an active license. This is not a gate: you can plan and approve without one, but execution is refused with the code `license-inactive`, shown as **No active license**, until a current license key is applied. A recovery plan that is already executing is not stopped, and neither are the restores it starts. See Licensing.

Restore runs: all restore methods

Code	Kind	Meaning	What to do
`rodc.restore`	Blocks approval	The target is a read-only domain controller. Read-only DCs are never restored.	Rebuild the machine instead.
`fsmo.seized-after-backup`	Blocks approval	An operations master role this DC held at backup time has since been seized by another DC.	Rebuild the machine. A DC whose roles were seized must never be restored.
`version.identifier-uncaptured`	Blocks approval	The backup has no wbadmin version identifier, which is what a restore addresses.	Choose another restore point.
`version.pruned`	Blocks approval	Retention has already deleted this version from the vault.	Choose a version that is still in the vault.
`version.no-manifest`	Blocks approval	The version has no manifest, so there is nothing to check it	Choose a version taken by Cygna

Code	Kind	Meaning	What to do
		against, and no evidence the agent was on the machine to finish the restore.	Forest Recovery.
`version.beyond-tombstone`	Blocks approval	The restore point is older than the domain's tombstone lifetime. Restoring it would bring back deleted objects as lingering objects.	Choose a newer version, or rebuild the machine.
`version.age-unknown`	Blocks approval	When the version was taken was not recorded, so the tombstone check cannot be made.	Choose a version whose time is known.
`version.unknown`	Blocks approval	The version the run addresses is no longer in the job history.	Plan the restore again against a version the Manager still knows.
`sysvol.frs`	Blocks approval	The backup records that SYSVOL replicates with FRS, which is not supported for recovery.	Migrate the domain to DFSR, then take fresh backups.
`sysvol.unknown`	Blocks approval	The SYSVOL replication engine was not recorded in the backup.	Take a fresh backup, so the manifest records it.
`agent.not-connected`	Blocks approval	No agent is connected for the machine. A DC state recovery is driven by the agent on the machine itself.	Install or reconnect the agent. If the operating system is

Code	Kind	Meaning	What to do
			gone, use full system recovery instead.
`backup.in-flight`	Blocks approval	A backup is running on the machine, and the restore would restart it mid-backup.	Let the backup finish, or cancel it, then approve the restore again.
`machine-password.stale`	Advisory	The machine account password has changed since the backup, so the restored DC's secure channel will fail until it is reset.	Nothing; the reset is part of the restore's validation.
`colocated.manual-followup`	Advisory	Other roles on the machine, such as AD CS or DHCP, are not recovered by Cygna Forest Recovery.	Plan for them separately. The recovery report lists them as manual follow-ups.
`agent.ca-changed`	Blocks approval when network settings for the restored system are planned, and always for a test restore; otherwise advisory	The agent in the backup trusts a different Manager CA, because the Manager was re-keyed after the backup.	Re-point the agent as soon as the machine is up; the run waits until it connects. With planned network settings, plan the restored network as unchanged and set it by

Code	Kind	Meaning	What to do
			hand, or restore a backup taken after the agent was re-pointed. See Re-Keying the Manager CA.
`version.taken-during-recovery`	Advisory	The backup was taken while a recovery of this domain controller's forest was under way. It holds the directory part-way through that recovery: operations master roles just seized, domain controllers removed and not yet rebuilt, and initial synchronization turned off. The gate names the plan and the last step it had completed.	Restore it only if that is the state you mean to return to. Otherwise choose a backup taken before the recovery began, or after it completed. Forest and domain plans raise this gate too.

Restore runs: DC state recovery

Code	Kind	Meaning	What to do
`build.ubr-mismatch`	Blocks approval	The machine has been patched since the backup. DC state cannot be put back across a Windows update revision change.	Take a fresh backup and restore that.
`build.build-mismatch`	Blocks	The machine	Rebuild the

Code	Kind	Meaning	What to do
	approval	runs a different Windows release from the backup.	machine instead.
`build.no-target`	Blocks approval	Nothing is running under the machine's name to compare the build against.	A DC state recovery needs the machine running. If the operating system is gone, use full system recovery.
`build.manifest-uncollected`	Blocks approval	The backup did not record the Windows build or revision.	Take a fresh backup, so the manifest records the build.
`build.target-uncollected`	Blocks approval	The machine has not reported its Windows build or revision.	Check that its agent is connected and current.
`target.insufficient-space`	Blocks approval	The machine does not have enough free space to stage the version locally. It needs about 1.15 times the backup's size.	Free space, or set the **Restore staging volume** to a volume that has it.
`target.free-space-unknown`	Advisory	The machine did not report its free space.	Nothing now. The agent checks again before it copies, and refuses if the

Code	Kind	Meaning	What to do
			copy will not fit.
`authoritative.agent-too-old`	Blocks approval	The domain controller's agent cannot mark the rollback authoritative: it is older than the Manager, or it is current but not running as its Windows service, which it must be to hold back the restart wbadmin starts.	Update the DC agent from the **Agents** screen, or check that its service is running, then approve.
`authoritative.overwrites`	Needs your acceptance	The rollback overwrites the named subtrees and objects on every domain controller: every change made to them since the backup is undone, including attributes, password resets, and group memberships. Objects created since the backup are left as they are.	Check the list. Accept it to roll them back, or cancel the run and plan again with different entries.
`authoritative.krbtgt`	Needs your acceptance	A named subtree holds `krbtgt`, so the domain's	Accept it to go ahead. Without the

Code	Kind	Meaning	What to do
		Kerberos keys and every user's password in it go back to the backup on every domain controller. Tickets issued since then stop working. The title says whether `krbtgt` is then reset twice.	reset, reset `krbtgt` twice yourself afterwards, or plan again with the reset chosen.
`authoritative.krbtgt-agent-too-old`	Blocks approval	The rollback resets `krbtgt`, and the domain controller's agent cannot wait for the first reset to reach every writable domain controller before the second.	Update the DC agent from the **Agents** screen, then approve.
`authoritative.group-policy`	Advisory	The rollback reaches Group Policy objects. They roll back in the directory, but SYSVOL is restored non-authoritatively, so their files come back from the other domain controllers as they are now.	Once the rollback has replicated, restore the policies from a Group Policy backup, for example with `Restore-GPO`.

Restore runs: full system recovery image and backup

Code	Kind	Meaning	What to do
`version.systemstate-only`	Blocks approval	The version is a DC state backup, or its type was not recorded. Full system recovery restores every critical volume, which needs a full system backup.	Choose a full system version, or use DC state recovery if the operating system still starts.
`image.none`	Blocks approval	No recovery image is held for the backup's operating system.	Depends on the cause given; see the table below.
`image.failed`	Blocks approval	The only recovery image for this operating system failed a boot test.	Rebuild the image. A failed image is not staged.
`image.untested`	Advisory	The recovery image has never been proven by a restore. The gate clears once a full system recovery or a test restore with the image completes its restore, which marks the image proven.	Nothing. If the image does not start, the machine returns to its own Windows at the next start, because the recovery boot is one-shot, and the run reports the missed boot. Onto a new VM, the VM boots the image from the run's recovery ISO. If it does not boot, nothing enrolls and the run fails when its media expires. Nothing

Code	Kind	Meaning	What to do
			will have been written, and the VM can be deleted.
`image.agent-cannot-apply-network`	Blocks approval	The image was built with an older recovery agent that cannot apply the restore's planned network settings or post-recovery directives. The gate names the missing abilities.	Upgrade the storage agent and wait for it to rebuild its images, or plan the network as unchanged.
`disks.layout-changed`	Blocks approval	The machine's disks no longer match the backup's. An in-place recovery cannot repartition.	Choose a version taken since the change, or recover onto a candidate machine.
`disks.layout-unknown`	Advisory	Whether the disks still match the backup could not be checked.	If a disk was added, resized, or replaced since the backup, wbadmin will refuse after the machine restarts into the recovery image, without writing anything.
`agent.restored-behind`	Advisory	The DC agent the restored system will start is older than the DC agent package the Manager holds. A	Update the agent on the machine that stages the restore first, with `cdr agent deploy`, while it is

Code	Kind	Meaning	What to do
		restored DC starts the agent build of the machine that stages the restore when the recovery image can put it in place, and the build in its backup otherwise. The Manager does not send recovery plan steps to an agent that is behind its package.	still reachable. If the gate says the recovery image cannot replace the restored DC's agent, update the storage agent so that it rebuilds its recovery images. Otherwise every plan step on the restored DC is refused until its agent is updated by hand. A standalone restore runs no plan steps and is not affected.
`image.agent-cannot-boot-media`	Blocks approval	The recovery image's agent cannot run from recovery media. A recovery onto a new VM, and a test restore, boot a per-run ISO whose job carries no vault credential; the recovery agent asks the Manager for it after enrolling.	Upgrade the storage agent and wait for its images to rebuild, or recover onto a candidate machine instead.

The remediation for `image.none` depends on why no image exists. Every variant notes that a DC state recovery needs no image if the operating system still starts.

Cause	What to do
No recovery image folder is set	Set **Recovery image folder** for the storage agent on the **Configuration** screen, and restart the storage agent. The **OS Media** screen shows the image as it is built.
No installation media to build from	Add a Windows Server ISO matching the backup's build to the OS media folder. The storage agent then builds the image.
Media is held, but the image is not built yet	Wait: the storage agent builds it on its next pass.
No storage agent has reported	Check that the storage agent is connected.

Restore runs: full system recovery target machine

These checks use the target machine's own description of itself, collected when the restore is planned, approved, and executed. Several have separate wording for a recovery in place and onto a candidate machine.

A recovery onto a new VM, and a test restore, have no machine to describe until the recovery ISO boots, so most of these checks are not made for them. The recovery agent checks the VM's firmware and disks against the backup before it writes. A recovery onto a new VM raises `target.original-running` and `target.new-vm` instead. A test restore raises neither, because the original keeps running. It raises the test restore gates listed below.

Forest and domain recovery plans also evaluate the firmware and disk checks for each domain controller restored onto a bound candidate machine. See Forest and domain recovery plans.

Code	Kind	Meaning	What to do
`target.survey-missing`	Blocks approval on a candidate; advisory in place	The machine has not described its hardware, or its description is too old.	Select **Ask the machine again** on the run's page. An agent older than the Manager does not answer; update it. In place, the disk check falls back to the machine's newest backup.
`target.clock-skew`	Advisory	The machine's clock differs from the Manager's by more than 5 minutes.	Correct the time, or on a virtual machine the host's time, before recovering. Kerberos refuses beyond five minutes.
`target.firmware-mismatch`	Blocks approval	The machine boots under different firmware (UEFI or BIOS) from the backup. The restored system would not start.	On a candidate, switch its firmware (on Hyper-V, generation 2 is UEFI and generation 1 is BIOS) or choose another candidate: a candidate is restored as it is. To check a backup without touching a machine, run a test restore on an Isolated Recovery Host instead, which builds a virtual machine of the backup's own generation. In place, choose a version taken since the change.
`target.firmware-unknown`	Advisory	Whether the	If the backup

Code	Kind	Meaning	What to do
		firmware matches could not be checked.	recorded its firmware, the recovery environment checks it before writing. If not, nothing can check it; confirm by hand.
`target.uefi-unproven`	Advisory	A UEFI recovery onto a platform other than a Hyper-V generation 2 virtual machine running Windows Server 2019 or Windows Server 2022. The gate is not raised for those.	Nothing. If the image does not start, the machine returns to its own Windows at the next start, because the recovery boot is one-shot, and the run reports the missed boot. Nothing is written.
`target.disks-unknown`	Blocks approval on a candidate; advisory in place	Whether the machine's disks can hold the backup could not be checked, because disk sizes were not reported or the backup did not record them.	Ask the machine again, or take a fresh backup. Otherwise check by hand that each disk is at least as large as the backup's.
`target.disks-too-small`	Blocks approval	A disk the backup needs has no disk on the machine at least as large. wbadmin would refuse after the	Give a candidate larger disks, or choose another. In place, choose a version taken since the change.

Code	Kind	Meaning	What to do
		machine restarts.	
`target.disks-unverified`	Advisory	Some of the backup's disks have no room on the machine and may be needed.	If any holds the directory database, its logs, or SYSVOL, wbadmin will refuse after the restart. A backup from a current agent records which disks are critical.
`target.partition-style-changed`	Blocks approval	The machine's boot disk has been converted between MBR and GPT since the backup.	Choose a version taken since the conversion.
`target.memory-insufficient`	Blocks approval	The machine has less memory than the recovery image needs to start from a RAM disk: the image's size plus 1 GB.	Give the machine more memory.
`target.memory-unknown`	Advisory	Whether the image fits in memory could not be checked.	Nothing now. The agent checks again before it arms the boot.
`target.less-memory`	Advisory	The candidate has less memory than the original.	The DC will run, possibly slowly. Add memory after recovery if it matters.
`target.fewer-processors`	Advisory	The candidate has fewer	Nothing.

Code	Kind	Meaning	What to do
		processors than the original.	
`target.staging-space`	Blocks approval	Not enough free space to download and stage the recovery image on the system volume: about 1.1 times the image's size.	Free space on the system volume.
`target.staging-space-unknown`	Advisory	Free space for the image was not reported.	Nothing now. If the image does not fit, staging fails before anything is armed.
`target.vault-unresolved`	Needs your acceptance	The storage agent advertises its shares by name, and the machine cannot resolve that name. The recovery image is fetched and the vault is read through that name for the whole restore. If the image fetch cannot resolve it either, the restore refuses before anything is changed. If the recovery environment cannot, it refuses and	Accept the gate only if you know the name will resolve to the storage machine from where the restore runs, for example through the recovery environment's own DNS server. Otherwise cancel the run. A name is also a redirection point: whatever answers for it receives the vault read credential and supplies what is restored onto the machine, so accept it only if that DNS is trustworthy. To remove the dependency, set

Code	Kind	Meaning	What to do
		restarts back into Windows, which costs a reboot cycle and, in place, keeps the domain controller down for its length.	**Advertised host** on the storage agent to the IP address the domain controllers reach it on, and restart the storage agent.
`target.vault-unreachable`	Blocks approval	The machine cannot connect over SMB to the storage machine, at the address the storage agent advertises its shares on. That is the address the recovery environment uses.	Open TCP 445 from the machine to the storage machine: firewalls, routing, and isolation rules.
`target.vault-not-in-dns`	Needs your acceptance	The storage agent advertises its shares by name, and that name resolves on the machine only through a hosts file, LLMNR, or NetBIOS. The recovery environment resolves names through DNS and nothing else. If its DNS cannot answer	Accept the gate only if you know the recovery environment's DNS will answer for the name with the storage machine's address. Otherwise cancel the run. Accept it only if that DNS is trustworthy, for the same reason as `target.vault-unresolved`. To remove the dependency, set **Advertised host** on

Code	Kind	Meaning	What to do
		for the name, it starts, fails to find the vault, refuses, and restarts back into Windows.	the storage agent to its IP address, and restart the storage agent.
`target.vault-by-name`	Advisory	The storage agent advertises its shares by name, so the recovery depends on DNS. The name resolves through DNS from the machine now, but in a forest recovery the DNS that answers for it may be the directory being recovered.	Set **Advertised host** on the storage agent to the IP address the domain controllers reach it on, and restart the storage agent. A name is also a redirection point for the vault read credential.
`target.vault-unchecked`	Advisory	Whether the machine can reach the storage machine was not checked: no storage agent was connected, none has reported the share it advertises and no backup has been staged, or the machine ran out of time first.	Connect the storage agent, then ask the machine to describe itself again. The recovery image is fetched from the storage machine and the vault is read from it for the whole restore.

Code	Kind	Meaning	What to do
`target.original-unchecked`	Advisory	The candidate ran out of time before checking every address the original DC used.	Make sure the original is off or isolated, then ask the candidate again.
`target.original-answering`	Blocks approval	The original domain controller still answers on the network. Two machines with one identity damage a domain.	Power the original off or isolate it, then ask the candidate again.
`target.original-address-in-use`	Advisory	Another directory server answers at the original's address.	Resolve the address conflict before the restored DC starts.
`target.storage-controller-changed`	Advisory	The candidate has a different platform or boot storage controller from the original.	The restored Windows needs a driver for the new controller, or it stops at INACCESSIBLE_BOOT_DEVICE. Built-in drivers cover common virtual controllers; any other boot controller may need its driver installed on the original before the backup.
`target.bitlocker`	Advisory	Volumes in the backup were protected by	Have the recovery keys ready.

Code	Kind	Meaning	What to do
		BitLocker, whose TPM protector belongs to the original machine.	
`target.static-address-not-carried`	Advisory	The original had static addresses, which are unlikely to follow it onto a candidate's different network adapter.	Plan the restored system's network as the backup's own static address, and its agent puts the address back at first start. When you do not choose a network for the restored system, the Manager plans this for you wherever the backup's agent can apply it and the address is unambiguous. Otherwise set the addresses by hand afterwards.
`target.recovery-needs-dhcp`	Advisory	Every adapter on the machine is statically addressed, and the recovery environment takes its address from DHCP unless the plan gives it one.	Give the recovery environment the machine's own address or another in the restore's network options, or make sure a DHCP scope serves the network.
`target.original-running`	Blocks approval	Recovery onto a	Power the original off, or disconnect it,

Code	Kind	Meaning	What to do
		new VM only: the original domain controller is still running; its agent is connected to the Manager. The new VM comes up as this machine, with the same name and the same SID. Two of them on one network damage the domain rather than recover it.	then approve.
`target.new-vm`	Advisory	Recovery onto a new VM: the VM is checked by the recovery environment, not here. Nothing runs on the VM before the recovery ISO boots, so there is no description of it.	Build the VM with `cdr dc rebuild --restore-run`, which shapes it from this backup's manifest and attaches the ISO with the disk first in the boot order. Before wbadmin writes, the recovery agent checks the VM's firmware and disks against the backup and stops, with nothing written, if they do not fit. See Full System Recovery onto a New VM.

Restore runs: test restores

A test restore restores a version onto a new VM on an Isolated Recovery Host, isolates it, checks it, and deletes it. It raises these gates in addition to the image and backup gates above. Because the original keeps running, `target.original-running` is not raised. For what a test restore does, see Test Restores.

Code	Kind	Meaning	What to do
`verification.host`	Blocks approval	The Isolated Recovery Host cannot run this test restore. The gate names the host and what stops it. A test restore runs on a host that is connected, ready, and not running another one.	The Agents screen shows each host's state and what stops it. Wait for the other test restore to finish, or fix the host's readiness finding.
`verification.vm-shape`	Blocks approval	A VM cannot be shaped from this backup. The host builds the VM from the backup's manifest, as `cdr dc rebuild` does.	Test a backup whose manifest records the machine's firmware, disks, and processors.
`verification.isolated`	Advisory	Always shown: nothing in production changes. The restore is the real one, from this run's recovery ISO with the vault read under a lease. The VM is then powered off, moved to a private switch of its own, started there as the first DC of a recovered domain,	Nothing. The original keeps running throughout, and the restored copy never reaches its network.

Code	Kind	Meaning	What to do
		checked, and deleted.	

Restore runs: recovery environment network

These gates appear when the restore plans an address for the recovery environment instead of DHCP.

Code	Kind	Meaning	What to do
`network.recovery-planned`	Advisory	The recovery environment takes the machine's own address, or a planned one, instead of DHCP.	Nothing. If the address cannot be applied, the run refuses before writing and the machine restarts into its own Windows. It never falls back to DHCP.
`network.recovery-adapter-unverified`	Advisory	Which adapter takes the address could not be checked.	Ask the machine to describe itself again, or name the adapter by its MAC address.
`network.recovery-no-dns`	Advisory	The planned address has no DNS server, and the storage agent advertises its shares by name, or whether it advertises a name or an address is not known. The recovery environment resolves names through DNS alone, so without a DNS	Add the DNS servers for the network, or set **Advertised host** on the storage agent to its IP address. If the advertised address is not

Code	Kind	Meaning	What to do
		server it cannot find the vault, and refuses. When the storage agent advertises an IP address, the recovery environment needs no DNS server and this gate does not appear.	known, ask the machine to describe itself again.
`network.recovery-original-unknown`	Blocks approval	The machine has not described its adapters, so its own address is not known.	Ask the machine again, or plan the address as an override.
`network.recovery-original-not-static`	Blocks approval	No adapter on the machine has a static IPv4 address to carry over.	Plan the network as unchanged (DHCP), or plan an override address.
`network.recovery-adapter-ambiguous`	Blocks approval	The machine has several adapters and the plan does not say which one takes the address.	Name the adapter by its MAC address in the restore's network options.
`network.recovery-adapter-missing`	Blocks approval	The adapter the plan names is not on the machine.	Name one of the adapters the machine described, or ask it again if one was just added.
`network.recovery-unresolved`	Blocks approval	The recovery environment's address cannot be worked out.	Change the restore's network options.

Restore runs: restored system network

These gates appear when the restore plans the restored system's address at its first start.

Code	Kind	Meaning	What to do
`network.restored-planned`	Advisory	The restored system takes its backed-up address, or a planned one, at its first start.	Nothing. The restored DC's agent applies it before contacting the Manager and, if the address changes, re-registers the DC in DNS. A failure is written to `POST-RECOVERY-README.txt` on the machine.
`network.restored-agent-unknown`	Blocks approval	The backup's manifest could not be read, so whether its agent can apply network settings is not known.	Restore a backup whose manifest the Manager holds, or plan the restored network as unchanged.
`network.restored-adapter-unverified`	Advisory	Which adapter takes the address could not be checked. Not raised for a recovery onto a new VM, which has a single adapter.	Ask the machine to describe itself again.
`network.restored-no-dns`	Advisory	The planned address has no DNS server. A DC with none cannot find its partners or register itself.	Point it at itself, or at another DC on the recovery network.
`network.restored-new-subnet`	Advisory	The restored DC moves to a different subnet.	Afterwards, add a subnet for the new

Code	Kind	Meaning	What to do
			range mapped to the DC's site, remove old A records, update delegation and reverse zones, and update anything that reaches the DC by its old address, such as DNS forwarders, firewall rules, and DHCP authorization.
`network.restored-original-unknown`	Blocks approval	The backup did not record the address to put back.	Plan the restored network as an override, with the address it should have.
`network.restored-original-not-static`	Blocks approval	The backed-up DC took its address from DHCP.	Plan the network as unchanged, or plan an override address.
`network.restored-original-ambiguous`	Blocks approval	The backed-up DC had several statically addressed adapters, and only one is carried.	Plan the adapter that matters as an override, and set the others by hand afterwards.
`network.restored-adapter-ambiguous`	Blocks approval	The target has several adapters and the plan does not say which takes the address.	Name the adapter by its MAC address.

Code	Kind	Meaning	What to do
`network.restored-adapter-missing`	Blocks approval	The adapter the plan names is not on the target.	Name one of the adapters the machine described.
`network.restored-unresolved`	Blocks approval	The restored system's address cannot be worked out.	Change the restore's network options.

Forest and domain recovery plans

These gates are evaluated when a forest or domain recovery plan is built, and decide whether it can be approved.

For each domain controller the plan restores by full system recovery onto a bound candidate machine, the plan also compares the candidate's newest description of itself with the chosen backup, using the target machine checks above: `target.survey-missing`, `target.firmware-mismatch`, `target.firmware-unknown`, `target.uefi-unproven`, `target.disks-unknown`, `target.disks-too-small`, `target.disks-unverified`, `target.partition-style-changed`, `target.storage-controller-changed`, and `target.bitlocker`. They have the same kinds as on a restore run onto a candidate, except that `target.survey-missing` is advisory in a plan: the restore asks the machine again when its step runs, and refuses then. These checks are made again from the machines as they are when the plan is approved, and approval is refused while any that blocks stands. Domain controllers restored in place are not checked at plan level; their restore run checks them. Memory, staging space, vault reachability, and whether the original is still running are checked by the restore run, not the plan.

Code	Kind	Meaning	What to do
`scope.no-root`	Blocks approval	Forest plans: no domain in the plan is known to be the forest root.	Refresh the topology, so the root is known.
`input.ntp`	Blocks approval	Forest plans: no external time source is given for the forest root's PDC emulator.	Give at least one time server reachable from the isolated recovery network.
`input.gc-designation`	Blocks	Forest plans: the DC	Choose a restored

Code	Kind	Meaning	What to do
	approval	chosen to have the global catalog added back is not a restored forest-root DC.	DC in the root domain.
`domain.no-restore`	Blocks approval	No domain controller in a domain is marked to be restored.	Mark one writable DC with a usable version as the restore.
`domain.multiple-restores`	Blocks approval	More than one DC in a domain is being restored from backup. A recovery restores exactly one DC per domain.	Restore one DC and rebuild the rest. Each rebuilt DC is promoted fresh from the restored one and brings the whole directory with it, so users, groups, memberships, and permissions lose nothing. What a rebuild does not bring back is what was on the machine itself: its roles are reinstalled from its last backup's record, but their configuration and its installed programs are not. The plan names both for each DC.
`restore.rodc`	Blocks approval	A read-only DC is marked to be restored.	Mark it rebuild or exclude.

Code	Kind	Meaning	What to do
`restore.no-version`	Blocks approval	No restore point is chosen for a DC being restored.	Choose one. After ransomware, choose one from before the compromise.
`restore.no-target`	Blocks approval	A DC being restored has no agent, and no candidate machine is bound to receive it.	Bind a candidate machine with an enrolled agent to the DC.
`target.not-candidate`	Blocks approval	The machine bound to a DC is not a candidate machine.	Bind a machine with an enrolled DC agent that is not itself a domain controller. `cdr recovery candidates` lists them.
`target.shared`	Blocks approval	One machine is bound to more than one DC.	Each restored or rebuilt DC needs a machine of its own. Bind a different candidate to each.
`restore.mode-a-elsewhere`	Blocks approval	A DC state recovery is planned onto another machine. DC state recovery only works in place.	Use full system recovery for a candidate machine.
`rebuild.no-dns-server`	Blocks approval	No address is known for the restored DC that a rebuilt DC takes its DNS from. Before promoting a rebuilt machine, the plan points its DNS at the restored DC, using	Give the restored DC an isolated address, or choose a backup taken while it had a static address.

Code	Kind	Meaning	What to do
		the static IPv4 address recorded in the restored DC's chosen backup, or the address the directory reported. The backup recorded none (the DC used DHCP), and the directory reported none.	
`rebuild.no-target`	Advisory	No machine is bound to be promoted as a DC being rebuilt.	The plan waits for an agent to connect under the DC's name: reinstall the machine clean under that name and enroll the agent, or bind another machine before approval.
`version.spread`	Advisory	The chosen restore points are more than 6 hours apart, which is a source of lingering objects.	Align them where the available versions allow.
`network.duplicate-address`	Blocks approval	Two DCs are planned onto the same isolated address.	Give each restored DC its own address.
`input.rid-increase`	Blocks approval	The RID pool increase is not positive.	Enter a positive number. The default is 100,000.
`trust.external-manual`	Advisory	The other side of an external trust must be reset by hand. A	When the trust is reset, the partner's administrator

Code	Kind	Meaning	What to do
		domain recovery always resets the domain's trusts. A forest recovery resets trusts only when it is planned as a security breach, so a forest plan shows this gate only then.	resets their side with the password the console shows when the step runs.
`trust.external-unreset`	Advisory	Forest plans without the breach flag: an external trust is not reset, so this forest keeps the trust password its backup holds. Windows changes that password every 30 days and keeps only the current and the previous one, so if the partner forest has changed it twice since the backup, the trust does not authenticate.	After the recovery, reset both sides of the trust by hand, or plan the recovery as a security breach.
`scope.domain-unknown`	Blocks approval	Domain plans: the domain is not in the forest's topology.	Refresh the topology, or choose a domain it knows.
`scope.single-domain-forest`	Blocks approval	Domain plans: the forest has only one domain.	Plan a forest recovery instead.
`scope.forest-root`	Blocks approval	Domain plans: the forest root domain cannot be recovered on its own, because	Plan a forest recovery instead.

Code	Kind	Meaning	What to do
		it holds the forest-wide roles.	
`input.gc-strategy`	Blocks approval	Domain plans: the global catalog strategy is not recognized.	Choose lingering-object removal (the default) or rehosting.
`gc.no-agent`	Blocks approval	Domain plans: a global catalog outside the domain has no agent, and the plan reconciles every global catalog through its agent.	Install the DC agent on it, or remove its global catalog role, before approving.
`gc.none-outside`	Advisory	Domain plans: no global catalog outside the domain is known.	If the topology may be stale, refresh it before approving.
`trust.partner-manual`	Advisory	Domain plans: the live side of a trust reset has no agent to run it.	An administrator of the partner domain resets it with the password the console shows.
`dns.no-root-server`	Advisory	Domain plans: no live forest-root DC is known to act as the restored DC's DNS server.	Set the restored DC's DNS server by hand at the DNS repair step.

Code	Kind	Meaning	What to do
`scope.outside-effects`	Advisory	Domain plans: always shown. The plan changes machines outside the domain: global catalogs in other domains, and trusts.	Read the list of affected machines before approving.
`trust.restore-points-apart`	Advisory	Forest plans not planned as a security breach: the two sides of a trust inside the forest are restored from points 30 or more days apart. Without the breach flag no trust is reset, and each side keeps the trust password its own backup holds. Windows changes it every 30 days and each side keeps only the current and the previous one, so the two may no longer match and the trust would not authenticate.	Choose closer restore points, or plan the recovery as a security breach, which resets every trust.

Backup Job Errors

When a backup does not produce a restore point, the Manager records why as an error code and a sentence. The code is shown in the **Cause** column on the **Backup Jobs** screen, the sentence under **What went wrong** in the job's detail sheet, and both in `cdr jobs list --failed --detail`. Reports count failures by code. For the screen, see Backup Jobs.

Backups that do not start

Before it starts a backup, scheduled or on demand, the Manager checks five things. A backup that fails a check does not start:

Code	Meaning and what to do
`backup-already-in-flight`	A backup of the machine is already queued or running. Wait for it to finish. No failed job is recorded.
`dc-agent-unavailable`	No DC agent is connected for the machine, so no backup can be taken. A failed job is recorded. Check the agent; see Agent Enrollment and Connectivity. A refusal made before the machine's current agent enrolled, or while it has no agent, still appears on the Backup Jobs screen but does not make the domain controller's last backup count as failed or raise the last-backup-failed alert.
`license-inactive`	Shown as **No active license**. No license key is installed, the installed key does not license Cygna Forest Recovery for Active Directory, or the license has ended and its grace period is over. The Manager checks this first. No failed job is recorded: the **Backups and recovery operations stopped: no active license** alert and the banner on every screen say why backups have stopped. A manual backup is refused with the license's sentence as the reason. Scheduled backups stay due and start by themselves once a current key is applied. See Licensing.
`recovery-in-progress`	Shown as **A recovery is working on this DC**. A recovery plan is running over the machine, a restore run on it is in progress, or a plan recovered it in the last 2 hours and its DC agent has not connected yet. No failed job is recorded. A manual backup is refused. A scheduled backup waits and starts by itself when the recovery is done. Test restores do not hold backups.
`storage-agent-unavailable`	No storage agent is connected. A failed job is recorded. Check that the storage agent service is running and connected.

Job error codes

Code	Shown as	Cause and what to do
`staging-insufficient-capacity`	Insufficient staging capacity	The storage agent refused to accept the backup because the vault volume does not have room for its estimated size plus the free space reserve, and retention found nothing it was allowed to delete. Add capacity or reduce retention; see Retention. The **Repository** screen shows capacity and the space reserved for running backups.
`staging-credential-mint-failed`	Staging account could not be created	The storage agent could not create the local account the domain controller writes the backup with. Read the storage agent's log for the reason. The storage agent must run as LocalSystem on a machine where it is a local administrator.
`staging-mint-failed`	Staging credential could not be issued	The storage agent could not prepare staging for the backup for another reason. Read the storage agent's log at the time of the job.
`backup-timeout`	Backup not reported in time	The domain controller did not report the backup within the **Backup job timeout**. (If the DC agent restarted during the backup, the job is closed as `agent-interrupted` instead, as soon as the agent reconnects.) The backup was not cancelled: wbadmin is most likely still running, and if it finishes, the agent reports it and the backup is still committed. Wait, then check the job again. If large DCs regularly take longer, raise **Backup job timeout**, keeping the storage agents' **Reservation timeout** at least 30 minutes longer.
`agent-interrupted`	DC agent restarted during the backup	The DC agent stopped while the backup was running—during an agent upgrade, a crash, or a restart of the machine—and came back without it, so the outcome was not captured. The job's status is **Interrupted**. wbadmin is left running when the agent stops and may have finished: check the Microsoft-Windows-

Code	Shown as	Cause and what to do
		Backup event log on the domain controller. Nothing from this run was committed to the vault, even if wbadmin finished, so take another backup. To avoid this, upgrade DC agents outside backup windows.
`agent-reported-failure`	Agent reported a failure	The DC agent or wbadmin reported that the backup failed. The detail sheet gives the agent's error and the transcript. See Common causes.
`vault-commit-failed`	Vault commit failed	The backup finished on the domain controller, but the storage agent could not hash it and move it into the vault. Read the error in the detail sheet and the storage agent's log.
`vault-commit-timeout`	Vault commit not confirmed in time	The storage agent did not report the commit within the **Vault commit timeout**. The commit was not cancelled and is most likely still running, so look for the backup in the vault before looking in staging. If it happens often, raise **Vault commit timeout**.
`storage-agent-unavailable`	Storage agent unreachable	The storage agent was not connected when the job needed it.
`dc-agent-unavailable`	DC agent unreachable	The DC agent was not connected when the job needed it.
`backup-already-in-flight`	A backup is already running	Another backup of the machine was already queued or running.
`operator-cancelled`	Cancelled by an operator	An operator cancelled the backup. The job's status is **Cancelled**.
`orchestrator-fault`	Internal manager error	An unexpected error in the Manager. Read the Manager's log at the time of the job, and contact support if it recurs.

Two codes can appear on a job that completed:

Code	Shown as	Meaning and what to do
`version-id-uncaptured`	Backup version identifier not captured	The backup completed and is in the vault, but the wbadmin version identifier was not recorded, so it cannot be chosen as a restore point. Take another backup of the machine, and read the DC agent's transcript for why the identifier was missing.
`vault-commit-reconciled`	Vault commit confirmed late	The commit was confirmed after the Manager had stopped waiting for it, and the job was corrected when the result arrived or when the Manager next checked the storage machine. The restore point is good. The error says why the Manager stopped waiting. Raise **Vault commit timeout** only when it says the commit took longer than that setting; a commit reported across a Manager restart needs nothing.

Job statuses that need attention

- **Completed (unreconciled)**—The Manager could not confirm the job against the vault. Treat the restore point as uncertain, and take another backup.
- **Completed, not committed**—The domain controller produced the backup, but the Manager did not see its content reach the vault. It is not yet a restore point. The Manager checks the storage machine for these every hour and whenever the storage agent connects, well within the time after which **Delete abandoned staging folders** removes an idle folder. Content still in staging is committed, and content already in the vault is recorded, so the job becomes **Completed**. To check now, select **Check now** under **Backups not in the vault** on the **Repository** screen, or run `cdr repository uncommitted --check`. A job whose content is found in neither place stays as a record and is not a restore point. Take another backup.
- **Interrupted**—The job stopped before it finished and its outcome was not captured: the DC agent restarted during the backup (for example during an agent upgrade, with the code `agent-interrupted`), or the Manager restarted and the agent did not report the job's outcome afterward. A Manager restart during the vault commit does not interrupt the job: the storage agent finishes the commit, and the job completes. Nothing from an interrupted run was committed to the vault. Take another backup. An interrupted backup counts as a failed attempt in alerts and reports.

Common causes

Symptom	Cause and fix
Every backup of a DC fails, and the transcript says the system cannot find the file specified	The Windows Server Backup feature is not installed on the domain controller. The DC agent installs it when **Install the Windows Server Backup feature** is on, which is the default. If the setting has been turned off for that DC, install the feature by hand or turn the setting back on. The DC agent's log says at startup whether the feature is present. If the setting is on and the agent's install attempt failed, a restart still pending on the domain controller (after a promotion, for example) is the usual cause: restart the domain controller, and the agent tries again at startup. The job's error message says which of these applies.
Backups from every DC fail to connect to staging	No share publishes the staging folder on the storage machine. Create an SMB share over `<storage root>\staging`; see Setting Up the Storage Machine.
Backups fail with a logon failure to staging	The staging share's permissions do not admit the staging writers group, so every per-job account is refused. Leave **Manage the staging share's permissions** on, or grant the group Change permission on the share. A deny entry is reported as an error and must be removed by hand.
Backups fail from some DCs but not others	The storage machine has several addresses and advertises one the failing DCs cannot reach, or advertises its name. Set **Advertised host** to the address the DCs reach; see Agent Enrollment and Connectivity.
Backups are refused for capacity	The vault is full or tight. See the **Vault refusing backups** and **Vaults with no room for another version** alerts in Alerts Reference.
Timeouts when many DCs back up at once	Backups and commits share the storage machine's disks. Raise **Backup job timeout** and **Vault commit timeout**, or lower **Maximum concurrent backups**.

For running and cancelling backups, see Monitoring and Cancelling Backup Jobs.

Agent Enrollment and Connectivity

An agent that does not appear on the **Agents** screen, or shows as not connected, has almost always said why in its own service log. Agents dial out to the Manager; nothing

connects to them. So start on the agent's machine, with the log in `%ProgramData%\Cygna Labs\Disaster Recovery Suite DC Agent\logs` or `%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent\logs`. At startup the agent logs the address it dials and the CA fingerprint it trusts. See Logs and Diagnostics.

How an agent connects

1. It fetches the Manager's CA from the address it was installed with, and keeps it only if the CA's fingerprint matches the one it was installed with.
2. It enrolls with the enrollment token, and receives its own certificate.
3. It opens a session with that certificate, and keeps it alive with heartbeats.

Each step logs what went wrong when it fails. The sections below follow the same order.

The agent cannot connect to any Manager

If the agent's settings cannot work at all, it logs one critical entry, `This agent cannot connect to a manager: …`, and then waits until it is stopped, rather than restarting over and over. The rest of the entry says why:

Reason in the log	Fix
The manager address is not https	The Manager serves HTTPS only. Reinstall the agent with `MANAGERADDRESS=https://<manager IP>:5211`.
No manager CA fingerprint is configured	Reinstall with `MANAGERCAFINGERPRINT` set to the fingerprint on the **Agents** screen, or push-install the DC agent from the console or with `cdr agent deploy`, which writes the fingerprint for you.
The configured fingerprint is not a SHA-256 fingerprint	The fingerprint is 64 hexadecimal digits, which may be grouped. Copy it from the **Agents** screen and reinstall.

Reinstalling with `msiexec /i <package> /qn` and the corrected property replaces the stored value.

The fingerprint does not match

The agent logs that the Manager at its address has one CA fingerprint while the agent is pinned to another, and that nothing was sent. Either the address points at a different Manager, or the agent was installed with the wrong fingerprint. The agent retries, so it connects once the cause is fixed.

- Compare the fingerprint in the log with the one on the **Agents** screen or in your recovery notes.
- If the agent has the wrong fingerprint, reinstall it with the right `MANAGERCAFINGERPRINT`.
- If the Manager was re-keyed, re-point the agent. See Maintaining Agents.
- If the address reaches a different machine, check the Manager's address and the agent's `MANAGERADDRESS`.

The enrollment token is refused

The Manager logs `Enrollment rejected for <machine> from <address>: invalid token`, and the agent never appears on the **Agents** screen. The token the agent holds does not match the one the Manager was installed with, or the agent was installed with none. Reinstall the agent with the Manager's token:

```
msiexec /i CygnaDR.DcAgent.msi /qn ENROLLMENTTOKEN=<token>
```

The token is a credential that can enroll new agents: a valid token enrolls an agent immediately, with no approval step. Windows Installer masks it in installer logs. Keep it out of scripts and command history.

The Manager counts wrong tokens by the address they come from. After 10 different wrong tokens from one address within 15 minutes, it refuses every enrollment from that address, even one with the right token, until 15 minutes after the first of them. It logs `Enrollment refused for <machine> from <address>: too many wrong tokens from that address` and records the refusal in the audit trail. An agent that retries the same wrong token is counted once, so a mistyped installation does not lock out its own address. Machines behind the same NAT share an address: an agent refused because another machine guessed retries on its own, and enrolls once the period has passed.

Enrollment is refused because the name is in use

The Manager keeps one agent record for each machine name and agent kind, and refuses to enroll a machine onto the record of an agent that is connected under that name at that moment. The refusal is audited. It means two machines are answering to one name, for example an original domain controller that was not switched off after a recovery rebuilt it elsewhere. Take the machine that should not be there off the network. The refused machine retries on its own once the other session has ended.

The agent cannot reach the Manager

The agent logs that it could not fetch the Manager's CA, or that its session was lost, and retries with a growing delay. Check:

- **The port**–Agents connect to the Manager on TCP 5211, or the port given to the Manager with `GRPCPORT`. See Ports and Protocols.
- **The Manager's firewall rules**–The Manager installer adds an inbound rule for the agent port on every network profile. Check that it exists and is enabled with `Get-NetFirewallRule -DisplayName 'Cygna DR Manager*'`. A Group Policy that does not apply local firewall rules ignores it; in that case, allow the port in the policy.
- **Firewalls between the machines**–The heartbeat keeps the connection open. A firewall that drops idle connections faster than the **Heartbeat interval** breaks sessions.
- **The Manager service**–Check that `CygnaDRManager` is running on the Manager.

The Manager address is a name

An agent given a host name rather than an IP address logs a warning at startup that the address is a name. The Manager is normally not domain-joined, so its name may be in no zone the agent's machine queries. When Active Directory is down, DNS usually is too. If the agent cannot enroll, reinstall it with `MANAGERADDRESS` set to the Manager's IP address, and give the Manager a static address or a reserved DHCP lease.

Clocks

Keep the clocks of the Manager, the storage machine, and the domain controllers correct. Certificates are only valid within their dates, so a machine whose clock is far wrong can treat a valid certificate as expired or not yet valid. Full system recovery preflight also warns when a machine's clock differs from the Manager's by more than five minutes, because Kerberos on a restored DC refuses beyond that.

Cygna Forest Recovery does not change any machine's clock. The Manager measures how far each connected agent's clock is from its own, and when the difference is more than five seconds it corrects the times that agent reports onto the Manager's clock. A recovery environment starts without the time settings of the Windows it came from, so its clock is often hours out. The restore's timeline says once that the agent's times were corrected, and by how much.

Certificate problems

Agents renew their certificates automatically. If the agent logs that its certificate can no longer be used and that it is re-enrolling, no action is needed. This happens when a certificate has expired, or when the Manager was restored from a snapshot that predates the agent or was re-keyed. The agent enrolls again with its token. A renewal that fails is logged as a warning and retried, and the current certificate stays in use. Re-enrollment needs the agent's token to match the Manager's; see The enrollment token is refused.

Storage agent problems

Symptom	Cause and fix
The storage agent service stops at startup	The storage root is unreachable, for example because its volume is not yet present. The installer does not check the path. Once the volume is present, start the service by hand.
The log says no non-administrative share publishes the staging folder	No share exposes `<storage root>\staging`, so remote domain controllers cannot reach staging and their backups fail. Create a share over the staging folder (never over the storage root, and never an administrative share like `E$`). If staging is reached some other way, set **Staging share override (UNC)**.
The log says a share named by the staging share override does not exist	**Staging share override (UNC)** names a share that is not on the machine. Correct the setting, or clear it to let the agent find the share.
The log says the machine has several candidate addresses	The storage machine has more than one network address, cannot tell which one the domain controllers reach, and advertises its machine name instead, so backups depend on DNS name resolution. Set **Advertised host** to the address domain controllers should use, and restart the storage agent.
The share's permissions are reported at startup	The agent grants the staging writers group Change on the share when **Manage the staging share's permissions** is on, but never removes entries. It reports anything else that can write, and any deny entry, which it cannot repair. Review the reported entries, and remove a deny entry by hand.

Push installation failures

A push installation from the console or with `cdr agent deploy` reports one outcome per target, with a sentence and a transcript of each step:

Outcome	Meaning and common causes
Installed, **Upgraded**, **Reinstalled**	The agent is installed at the package's version and its service restarted.
Already current	The target already runs this build; nothing was changed. Use `--force` to reinstall anyway, as when re-pointing.
Planned only	A `--plan` run; nothing was changed.
Refused before starting	The deployment declined before it touched the target: the target has a newer agent and `--force` was not given; a fresh install has no Manager address, CA fingerprint, or enrollment token to write; the address is not https; or this machine already has a connection to the target under different credentials (Windows error 1219), which you must close first.
Unreachable	The target did not answer on a port the installation needs (TCP 445, and 5985 for WinRM or 135 for DCOM), or a service it needs, such as Remote Registry, did not answer. When WinRM is refused because the target is not in TrustedHosts, the result says so and gives the `Set-Item WSMan:\localhost\Client\TrustedHosts` command to run on the machine doing the install, which is the Manager for console installs. Or use DCOM, which needs no TrustedHosts entry but does need the Windows Management Instrumentation (WMI-In) firewall group enabled on the target.
Credential refused	The target rejected the account, or the account cannot write to the registry or `ADMIN$`. Use an account that is a local administrator on the domain controller, as `DOMAIN\user` or `user@domain`.
Install failed	The installer ran but the installation did not succeed, or it reported success but the target does not list the new version, or the agent service did not restart. The result gives the reason and the path of the installer log, which is kept in `%SystemRoot%\Temp` on the target when an installation fails.

Deployment errors that stop a whole run, such as **No package folder configured**, **No DC agent package in the folder**, or **A deployment is already running there**, are shown before any target is touched. A missing DC agent package stops the run only when some target needs a push. Set the **Package folder** on the **Configuration** screen, or wait for the other deployment to finish. For prerequisites, see Accounts and Permissions.

Self-update failures

When an agent updates itself, from the **Agents** screen or with `cdr agent update`, the result uses the same outcomes as a push installation, and the transcript gives the Manager's or the agent's own sentence. A refusal from the agent follows `the agent refused it:`. Common messages:

Message	Meaning and fix
`the agent is not connected to the manager`	**Unreachable**. The agent is offline. Check its service and its connection, as described earlier in this topic.
`the agent does not update itself: this machine's Policy switches it off`, or `this storage agent cannot update itself — …` (or `this Isolated Recovery Host agent …`)	**Refused before starting**. `AllowRemote` is `false` in its machine's policy. A DC agent can be pushed to instead. A storage agent or Isolated Recovery Host agent must be installed by hand: download its package from the **Agents** screen or with `cdr agent package download`, and run `msiexec /i <package> /qn` on the machine. See Maintaining Agents.
`updates from the manager are switched off on this machine (Policy)`	The agent's machine has `AllowRemote` set to `false`. Remove the policy value, or update the agent by another route.
`the package is not signed …`, `the package is not signed by Cygna Labs Corp.: …`, `the package's signature carries no timestamp …`, or `the package's signature is not valid: …`	The package in the Manager's package folder is not an original, signed release package. Replace it with the package from the Cygna Forest Recovery release.
`the package is '…', not this … agent's package`	The package folder holds the wrong product under this agent kind's file name. Put the right package back.
`the package folder holds no … package to update it with`	The package folder has no package for this agent's kind. The Manager's installer puts all three agent packages in `redist`. Restore the folder, or correct the **Package folder** setting.
`the agent runs …, newer than the package's …; an agent never updates itself to an older version`	The package folder holds an older build than the agent runs. Nothing is changed. A self-update never downgrades and never reinstalls the build an

Message	Meaning and fix
	agent already runs.
folder `has … MB free and a … MB package needs … MB with room to install`	The drive that holds the agent's data folder needs room for the package plus 256 MB. Free some space and try again.
A message that a backup, restore, recovery plan, or test restore is in progress, ending with when to update	**Refused before starting**. The machine has work in flight. Update it when that work has finished.
`another update is being checked or installed`, `update … is still installing`, or `a deployment is already running on …`	An earlier update of the same agent has not finished. Wait for it.
`the agent did not answer the offer within a minute`, `the agent received the package and said nothing more within five minutes`, or `no word from the agent within 20 minutes of starting the install of …`	The agent stopped responding during the update. Check the machine: whether the agent service is running and which version it reports, and the installer log in the `update` subfolder of the agent's data folder.
`still … after installing …; the installer rolled back or did not run`, or `the installer finished and … is still running …`	**Install failed**. The installer failed and Windows Installer rolled it back. The old build is still running. The transcript lists the installer's status and the log lines before the failure. The full log is in the `update` subfolder of the agent's data folder. See Installer Messages.
`no agent is enrolled with id …`	The agent ID is wrong, or the agent's record was deleted. Check the ID with `cdr agent list --wide`.

Unlike a push installation, a self-update leaves no log in `%SystemRoot%\Temp`. Its installer log stays in the agent's data folder, under `update`, and a summary of it comes back in the transcript.

Installer Messages

The four Cygna Forest Recovery packages (`CygnaDR.Manager.msi`, `CygnaDR.DcAgent.msi`, `CygnaDR.StorageAgent.msi`, and `CygnaDR.HostAgent.msi`) install

silently with `msiexec /i <package> /qn` and have no installer screens. When a package refuses to install, msiexec ends with an error, and the message is written to the installer log if you asked for one with `/l`. This topic lists each message, its cause, and the fix.

Note: An installer log records the properties given on the command line, except `ENROLLMENTTOKEN`, which Windows Installer masks. Keep commands that carry the token out of scripts and command history.

Messages that stop an installation

Message	Package	Cause and fix
`The Cygna Disaster Recovery Manager requires 64-bit Windows.` The DC agent, storage agent, and Isolated Recovery Host agent packages give the same message with their own names.	All	The machine runs 32-bit Windows. Install on 64-bit Windows Server 2019 or later, or for the DC agent, 64-bit Windows Server 2016 or later.
`The Cygna Disaster Recovery Manager requires Windows Server 2019 or later — build 17763 or higher. This machine reports build` *n*. The storage agent and Isolated Recovery Host agent packages give the same message with their own names. The DC agent package gives it with Windows Server 2016 and build 14393.	All	The operating system is older than the package supports. The message gives the build the machine reports. Install on a supported operating system: Windows Server 2019 or later for the Manager, storage agent, and Isolated Recovery Host agent, or Windows Server 2016 or later for the DC agent; see System Requirements.
`ENROLLMENTTOKEN is required, and must match the token given to the agent installers.`	Manager	The first installation of the Manager was run without `ENROLLMENTTOKEN`. Choose a token, give it on the command line, and store it offline: every agent needs the same token. An upgrade does not need it, because the installer keeps the stored token.
`MANAGERADDRESS is required.`	DC agent, storage	The first installation was run without `MANAGERADDRESS`. Give

Message	Package	Cause and fix
	agent, Isolated Recovery Host agent	the Manager's agent address, `https://<manager IP>:5211`. The **Agents** screen shows complete install lines with the right address.
`MANAGERCAFINGERPRINT is required: the manager CA fingerprint shown on the manager's console. Example: msiexec /i …`	DC agent, storage agent, Isolated Recovery Host agent	The first installation was run without `MANAGERCAFINGERPRINT`. Copy the fingerprint from the **Agents** screen. It is public, not secret.
`STORAGEROOT is required: the absolute path to this machine's storage volume, for example E:\CygnaDR. It cannot be set from the manager.`	Storage agent	The first installation was run without `STORAGEROOT`. Give an absolute path on the storage machine's backup volume.
`A newer version of the Cygna Disaster Recovery Manager is already installed. Uninstall it before installing this version.` The DC agent, storage agent, and Isolated Recovery Host agent packages give the same message with their own names.	All	The package is older than the installed version, and downgrades are not allowed. Install the current package instead. Uninstall first only if you really intend to go back to the older version.

Properties are required on a first installation only. An upgrade reads them back from the registry, so `msiexec /i <package> /qn` on its own is a complete upgrade, and a property you do give replaces the stored value. This includes the Manager's `RESTPORT`, `GRPCPORT`, `LISTENADDRESS`, and `REDIRECTPORT`. See Upgrading and Uninstalling.

Installer failures during a self-update

When an agent updates itself from the Manager, it runs the package with `msiexec /i … /qn /norestart /l*v` from a one-time scheduled task as SYSTEM, and writes the log to the `update` subfolder of its data folder, for example `%ProgramData%\Cygna Labs\Disaster Recovery Suite Storage Agent\update`. If the installer fails, Windows Installer rolls the change back, the old build keeps running, and the agent reports the failure. The update's transcript shows the msiexec status, the installer's failure message, and the log lines

before the first failure. The outcome is **Install failed**. For the messages an update gives before the installer runs, see Agent Enrollment and Connectivity.

The DC agent after a DC state restore

A DC state restore puts the registry back to how it was when the backup was taken. That includes a DC state recovery and each restore of a domain controller in a domain or forest recovery plan. Windows Installer's record of installed programs is in the registry, but its saved installer packages in `C:\Windows\Installer` are not, and the restore leaves them as they are. If the DC agent was updated after the backup, the restore brings back the record of the older agent, whose saved package the update deleted. Every install or update of the agent then fails with Error 1714, "The older version of Cygna Disaster Recovery DC Agent cannot be removed", and System Error 1612.

The agent mends this itself, so nothing is needed from you in the normal case:

- Before the restore, the agent copies its own installer package to its data folder, which a restore does not touch.
- On the first normal start after the restore, before it checks the restored domain controller, the agent looks at Windows Installer's record of itself. If the record names a version whose package is gone, the agent removes that one product's entries and reinstalls the package it copied, which is the version that was running before the restore. The agent's files, service, identity, and settings are kept. Its service restarts once, in about a minute. The run's timeline says what the agent did.

Afterwards the domain controller runs the same agent version as before the restore, and automatic and manual updates work as usual. A full system recovery restores whole volumes, including `C:\Windows\Installer`, and is not affected.

If the agent could not mend it, the restore still completes, and the run's **Steps** card notes that the agent on this domain controller needs reinstalling by hand, with the script's full path and the package to install. That happens when no package could be copied before the restore, or the reinstall failed; its log is `agent-reinstall.log` in the run's folder under the agent's restore data.

To reinstall the DC agent by hand:

1. On the domain controller, open an elevated PowerShell prompt in the agent's installation folder, for example `C:\Program Files\Cygna Labs\Disaster Recovery Suite DC Agent`.
2. List Windows Installer's stale records of the agent:

```
powershell -ExecutionPolicy Bypass -File .\Remove-StaleDcAgentRegistration.ps1
```

The script lists only agent versions whose installer package is missing. It never touches the agent's files, service, or settings, or any other product. If it finds nothing to remove, go on to step 4.

3. Remove the records it listed:

```
powershell -ExecutionPolicy Bypass -File .\Remove-StaleDcAgentRegistration.ps1 -Remove
```

4. Remove the agent's service, so that the installer can create it again:

```
sc.exe stop CygnaDRDcAgent
sc.exe delete CygnaDRDcAgent
```

5. Install the agent: the package the note names, or the current `CygnaDR.DcAgent.msi`:

```
msiexec /i CygnaDR.DcAgent.msi /qn
```

No Manager address or enrollment token is needed. The installer reads the existing settings, and the agent keeps its identity, so it reconnects as the same agent.

Problems after installation

Some problems appear only when the service starts. They are written to the service's log and, for warnings and errors, to the Application log.

Symptom	Cause and fix
The Manager service does not start, and logs that no listen endpoints are configured and it refuses to start on the framework default port	The Manager's port configuration is missing from its registry key and from `appsettings.json`. Reinstall the Manager with `msiexec /i CygnaDR.Manager.msi /qn`, giving `RESTPORT` and `GRPCPORT` if you use ports other than the defaults. Do not add your own listener settings (a `Kestrel` section) to `appsettings.json`: it would override the installed ports and leave the firewall rules on the old ones.
The Manager logs a warning that its redirect port is in use, and starts without it	Another program, such as IIS, holds the port given by `REDIRECTPORT` (80 by default). The console still works at `https://<manager>:5210/`. Free the port, or reinstall with `REDIRECTPORT=0` to turn the redirect off.

Symptom	Cause and fix
The storage agent service stops at startup	The storage root is not reachable, for example because its volume is not yet present. Once it is present, start the service by hand.
An agent logs that it cannot connect to a manager	Its Manager address or CA fingerprint cannot work. See Agent Enrollment and Connectivity.

Command-Line Reference

The `cdr` command-line tool gives you the same view of your environment as the console, from a command prompt or a script. It reads the same inventory, restore points, and verdicts from the Manager, and the Manager applies the same roles and capabilities to every request, so `cdr` and the console never disagree about what you are allowed to do.

Use `cdr` when a screen is not the right tool:

- **Monitoring**—`cdr alert list` exits with a non-zero code when any alert is raised, so a scheduled task or monitoring system can use it as a probe. `cdr alert notify status` exits with code 4 when alert email is on and not working.
- **Evidence and export**—`--json` on any command, and `cdr report show --json` and `cdr audit list --json` in particular, give you machine-readable records of backup outcomes and operator activity. `cdr audit export` writes the whole matching audit trail to a CSV, JSON, HTML, or PDF file.
- **Scripting**—Schedule, back up, plan restores, and change settings from scripts, with exit codes that say what happened.
- **Work the console does not do**—Push the DC agent to many domain controllers at once, size the backup storage before you install anything, build a virtual machine shaped like a domain controller's backup, and clean up the directory metadata of domain controllers, domains, and trusts that are gone.

Where cdr is installed

`cdr.exe` is installed with the Manager, in the Manager's program folder. In a default installation the path is:

```
C:\Program Files\Cygna Labs\Disaster Recovery Suite Manager\cdr.exe
```

The folder is not added to the `PATH` environment variable. Open a command prompt in that folder, or enter the full path. The DC agent package that `cdr agent deploy` pushes to domain controllers is in the `redist` folder beside it.

`cdr` runs on Windows. Signing in, the session cache, `cdr agent deploy`, `cdr cleanup`, and `cdr repository estimate` all depend on Windows features.

Running cdr

Commands are organized into groups. Enter the group, then the command, then any arguments and options:

```
cdr <group> <command> [arguments] [options]
```

Three commands—`login`, `logout`, and `whoami`—have no group. For example:

```
cdr login --username jsmith --manager https://10.0.0.10:5210
cdr dc list --protection unprotected
cdr backup start dc01.corp.example.com --wait
cdr restore show a1b2c3d4e5f6 --wait
```

Help is available at every level: `cdr --help` lists the groups, `cdr restore --help` lists the commands in a group, and `cdr restore plan --help` lists a command's arguments and options with examples. `cdr --version` shows the version of the tool.

An option that a command does not recognize is an error, not something `cdr` ignores. A mistyped filter would otherwise return a confident answer to a different question. In the same way, options that contradict each other, or a filter value that is not in the command's list, are refused with a message that names the accepted values.

Global options

Every command accepts these options:

Option	Description
`-m, --manager <URL>`	The Manager's console address, for example `https://10.0.0.10:5210`. Default: the `CYGNADR_MANAGER` environment variable, then `https://localhost:5210`.
`--manager-ca <FINGERPRINT>`	The Manager CA fingerprint to trust for this run, as shown on the console. Default: the `CYGNADR_MANAGER_CA` environment variable, then the fingerprint you saved for this address. Required when there is no terminal to confirm on and nothing is saved.
`--json`	Write raw JSON to standard output instead of a table. Fields that this version of `cdr` does not recognize are kept.
`--timeout <SECONDS>`	How long to wait for the Manager to answer. Default: 30.
`-h, --help`	Show help for the group or command.

Connecting to the Manager

Each run connects to one Manager. `cdr` takes the address from `--manager`, then from the `CYGNADR_MANAGER` environment variable, and otherwise uses `https://localhost:5210`, which is correct when you run `cdr` on the Manager itself. To run a series of commands against the same Manager, set the environment variable once:

```
set CYGNADR_MANAGER=https://10.0.0.10:5210
```

The Manager serves HTTPS only. An `http://` address is refused. If you enter a host without a scheme, `cdr` adds `https://`, and a host without a port goes to the Manager's default port, 5210. An address that starts with `https://` and has no port goes to port 443, for example to a reverse proxy.

Before it sends anything, `cdr` checks that the address answers with the certificate authority (CA) of the Manager you trust. The first time you connect to an address, `cdr` shows the Manager's CA fingerprint and asks you to compare it with the fingerprint shown on the console. When you confirm, it saves the fingerprint for your Windows account and does not ask again. Scripts and scheduled tasks, which cannot answer a question, pass the fingerprint with `--manager-ca` or `CYGNADR_MANAGER_CA`.

Commands that read or change the Manager's data need an operator session. Run `cdr login` once; the session is saved for your Windows account until it expires or you run `cdr logout`. For sign-in, sessions, and trust on first use, see Connecting and Signing In.

Commands that act outside the Manager

Most commands are requests to the Manager. A few do their work directly, with your own credentials, and use the Manager only to read inventory or record a result:

- **`cdr agent deploy`**—Connects to each domain controller with a local administrator account you supply for that run, and installs or upgrades the DC agent there. The domain controller's own security log records who installed it.
- **`cdr cleanup`**—Removes objects from Active Directory with a Domain Admin or Enterprise Admin account. Its only change to the Manager is recording that an object is gone.
- **`cdr dc rebuild`**—Creates a virtual machine on the Hyper-V host where you run it, or writes a script for VMware ESXi. Nothing on the Manager changes.
- **`cdr repository estimate`**—Measures your domain controllers to size the backup storage. It needs no Manager, so you can run it before you install anything.
- **`cdr trust list`**—Reads the fingerprints saved on this computer, without contacting any Manager.

Output

- **Standard output**—Tables, or JSON with `--json`. When output is redirected to a file or another program, tables are laid out 200 characters wide.

- **Standard error**—Errors, warnings, hints, progress, confirmation prompts, and messages such as "No restore runs." for an empty result. Standard output therefore carries only the answer, which keeps `--json` output clean when you redirect it.

With `--json`, list commands write the records the Manager returned. `cdr alert list --json` and `cdr audit list --json` write the whole response, including totals, the filter the Manager applied, and when it was evaluated. `cdr backup show --raw` writes a restore point's stored manifest byte for byte, which is the document its content hash covers.

Confirmations and secrets

Commands that change something you cannot easily undo ask for confirmation first. Some ask a yes-or-no question. Others ask you to type a name or code, so that you must read which object the action applies to before you approve it:

- `cdr restore run` asks you to type the domain controller's short name.
- `cdr cleanup dc`, `domain`, `trust`, and `forget` do nothing until you add `--confirm` with the object's own name.
- `cdr recovery ack` needs `--confirm` with the step's code.

Commands with a yes-or-no confirmation accept `--yes` to skip it. When standard input is not a terminal—in a script or scheduled task—these commands refuse to run without `--yes` rather than assume consent, and exit with code 1.

Passwords and tokens are never command-line arguments, because arguments are kept in shell history and are visible in process listings. Each command that needs one reads it from an environment variable or prompts for it, and the prompt hides what you type. In a script with no terminal, set the variable for that run only, and clear it afterward.

Exit codes

`cdr` returns an exit code that a script or scheduled task can act on:

Code	Meaning
`0`	Success.
`1`	Usage error: missing or contradictory arguments, or a confirmation that could not be given. Nothing was changed.
`2`	The Manager could not be reached at the address used.
`3`	The Manager refused the request, and nothing was changed. This includes a signed-in operator who lacks the capability the command needs.

Code	Meaning
`4`	Findings: the command ran and found something that needs attention–for example, a raised alert in `cdr alert list`, a failed backup with `cdr backup start --wait`, a plan with outstanding hard gates, or a restore that is waiting on an operator.
`5`	An unexpected problem in `cdr` itself, rather than a finding about your environment. Report it to support with the message `cdr` printed.
`6`	A local action failed–for example, creating the virtual machine in `cdr dc rebuild`, or writing a script file.
`7`	A remote action failed–for example, a `cdr agent deploy` target that failed, a `cdr repository estimate` that could not measure a domain controller, a failed restore, or a cleanup that stopped.
`8`	Outcome unknown: the Manager accepted a change, but its result could not be read back. Check the current state before you retry.
`9`	Untrusted: no Manager CA fingerprint is saved for the address, or the Manager presented a different one.
`10`	Not signed in, or the session has expired. Run `cdr login`.
`130`	Cancelled with Ctrl+C.

Environment variables

Variable	Used for
`CYGNADR_MANAGER`	The Manager address, when `--manager` is not given.
`CYGNADR_MANAGER_CA`	The Manager CA fingerprint to trust, when `--manager-ca` is not given.
`CYGNADR_PASSWORD`	Your operator password for `cdr login`.
`CYGNADR_DEPLOY_PASSWORD`	The password of the `--user` account for `cdr agent deploy`.
`CYGNADR_ENROLLMENT_TOKEN`	The enrollment token that `cdr agent deploy` gives new DC agents.
`CYGNADR_CLEANUP_PASSWORD`	The password of the `--user` account for `cdr cleanup`.

Variable	Used for
`CYGNADR_ESTIMATE_PASSWORD`	The password of the `--user` account for `cdr repository estimate`.
`CYGNADR_RECOVERY_ADMIN_PW`	The domain administrator password for `cdr recovery secrets`.
`CYGNADR_RECOVERY_RID500_PW`	Optional. The password the domain's built-in Administrator account had at the backup point, for `cdr recovery secrets`. Without it, `--rid500` prompts for the password.
`CYGNADR_RECOVERY_DSRM_PW`	The new DSRM administrator password for `cdr recovery plan --dsrm` and `cdr recovery secrets --dsrm`.
`CYGNADR_RESTORE_ADMIN_PW`	The domain administrator password for `cdr restore approve --admin` and `cdr restore reset-krbtgt --admin`, for a DC state recovery that resets `krbtgt`.
`CYGNADR_SETTING_SECRET`	The value of a password setting for `cdr config set`.
`CYGNADR_TRUST_FILE`	A different file for saved Manager fingerprints. Default: `%LOCALAPPDATA%\Cygna Labs\Disaster Recovery Suite\cdr-trusted-managers.json`.
`CYGNADR_SESSION_FILE`	A different file for cached sessions. Default: `%LOCALAPPDATA%\Cygna Labs\Disaster Recovery Suite\cdr-sessions.json`.

Permissions

The Manager checks every request against the capabilities of the operator who signed in. The reference for each command states the capability it needs and the role that grants it. When you lack it, the Manager refuses the request and `cdr` exits with code 3. When the Manager refuses for a reason of its own—for example, because you tried to remove your own account—`cdr` shows the Manager's message instead, and also exits with code 3. The break-glass account holds every capability. For the roles and what each grants, see Managing Operators and Roles.

Commands that act outside the Manager also need rights on the machines they reach—for example, local administrator rights on the domain controllers for `cdr agent deploy`. Those are listed with each command.

Manager maintenance commands

A few maintenance tasks are not part of `cdr`. They are commands of the Manager program itself, `CygnaDR.Manager.exe`. Run them locally on the Manager as an administrator; they cannot be reached over the network:

- `CygnaDR.Manager.exe ca status`—Shows whether the Manager's CA key is usable on this machine and whether an escrow copy exists.
- `CygnaDR.Manager.exe ca recover`—Recovers the CA key from its escrow copy with the recovery passphrase.
- `CygnaDR.Manager.exe ca rekey`—Replaces the Manager's CA.
- `CygnaDR.Manager.exe db snapshot [path]`—Takes an on-demand snapshot of the Manager database.
- `CygnaDR.Manager.exe support package [path] [--days N | --all-logs]`—Builds the Manager's part of a support package without the service, for when the service does not start.

For when and how to use them, see Recovering the Manager, Re-Keying the Manager CA, and Logs and Diagnostics.

Command groups

Group	Description
`login`, `logout`, `whoami`, `operator`	Sign in as an operator, sign out, and see who you are signed in as; list operator accounts, remove someone who has left, and restore someone who returns. See Connecting and Signing In.
`dc`, `domain`, `forest`, `site`	List the directory inventory with each object's protection, coverage, or readiness verdict, and build a virtual machine shaped like a domain controller's backup. See Inventory Commands.
`agent`, `recovery-host`, `trust`	List, deploy, update, restart, and uninstall agents, follow automatic agent updates, check, approve, and drain Isolated Recovery Hosts, and manage the Manager fingerprints saved on this computer. See Agent and Trust Commands.
`policy`, `backup`, `jobs`	Schedule backups, list and inspect restore points, start a backup now, and follow or cancel backup jobs. See Backup Commands.
`repository`, `storage`, `request`, `os-`	Check vault capacity and retention depth, follow up backups that did not reach the vault and changes to committed backups, estimate the storage you need, replace the storage machine and

Group	Description
`media`	decide requests that wait for a second operator, and check OS media coverage. See Storage and Media Commands.
`restore`	Plan, approve, run, and follow the restore of one domain controller, and start a DC state recovery's `krbtgt` reset again. See Restore Commands.
`recovery`	Plan and drive a domain or forest recovery. See Recovery Plan Commands.
`alert`, `audit`, `report`	List raised alerts, check alert email, read the audit trail, and report backup outcomes and coverage. See Alert, Audit, and Report Commands.
`config`, `exclusion`	Read and change settings, and put objects out of scope. See Configuration and Exclusion Commands.
`cleanup`	Remove the directory metadata of domain controllers, domains, and trusts that are gone, and record objects as removed. See Directory Cleanup Commands.
`support`	Gather the evidence for a support case from the Manager and every agent into one file, and list the support packages the Manager holds. See Support Commands.

In this chapter

- Connecting and Signing In
- Inventory Commands
- Agent and Trust Commands
- Backup Commands
- Storage and Media Commands
- Restore Commands
- Recovery Plan Commands
- Alert, Audit, and Report Commands
- Configuration and Exclusion Commands
- Directory Cleanup Commands
- Support Commands

Connecting and Signing In

Before `cdr` sends a request to a Manager, two things must be in place: `cdr` must trust the Manager's certificate authority (CA), and you must be signed in as an operator. Both are saved for your Windows account on this computer, so you set them up once per Manager.

Operator accounts are the Manager's own accounts, the same ones you use in the console. They are not Windows or Active Directory accounts, so you can sign in while the directory is down. The `cdr operator` commands at the end of this topic list, remove, and restore those accounts.

Trusting the Manager on first use

The first time you connect to a Manager address, `cdr` fetches the Manager's CA certificate, shows its fingerprint, and asks whether to trust it:

```
cdr has not connected to the manager at https://10.0.0.10:5210/ before.
Its CA fingerprint is:
 3F2A-91C0-5B7E-04D8-C1A6-2E9F-7B30-D4C2-8E15-6A9B-F0D3-1C47-B8E2-5D90-A6F1-
E7F8
Compare it with the fingerprint on the manager's console, or in its event log. Trust it only if they
match: a different value means you are not talking to your manager.
Trust this manager and remember it for this address?
```

Compare the value with a second source before you answer, such as the Manager's event log or the copy in your recovery notes; for where else it is shown, see Certificates and Trust. When you confirm, `cdr` saves the fingerprint for this address in `%LOCALAPPDATA%\Cygna Labs\Disaster Recovery Suite\cdr-trusted-managers.json`. Every later connection to the address must present the same CA, or `cdr` refuses to send anything and exits with code 9.

A script or scheduled task has no terminal to confirm on. Give it the fingerprint with `--manager-ca` or the `CYGNADR_MANAGER_CA` environment variable; `cdr` checks it against what the Manager presents. You can also save it once in advance with `cdr trust pin`. The fingerprint can be entered with or without the hyphens.

To list, save, or forget fingerprints, use the `cdr trust` commands described in Agent and Trust Commands. For background, see Certificates and Trust.

Sessions

- **Saved per Windows user and Manager**—`cdr login` saves the session for the Manager address you signed in to, protected with Windows data protection for your Windows account, in `%LOCALAPPDATA%\Cygna Labs\Disaster Recovery Suite\cdr-sessions.json`. Another Windows user on the same computer has to sign in separately.

- **Lifetime**–A session ends 12 hours after sign-in, or after 4 hours without use, whichever comes first. When it has ended, commands exit with code 10; run `cdr login` again.
- **Same rules as the console**–A `cdr` session is the same kind of session the console holds. The Manager applies the same capability checks, and the audit trail records your username against everything you do.
- **Ended by account changes**–Disabling or removing your account, an administrator resetting your password or your multi-factor authentication (MFA), changing your own password, or turning on MFA for yourself ends your other sessions, including the one `cdr` holds.

login

Signs in to the Manager as an operator and saves the session for this Windows user until it expires or you run `cdr logout`.

The password is never an argument. `cdr` reads it from the `CYGNADR_PASSWORD` environment variable, or prompts for it. If the account has a temporary password–a new account, or one an administrator has reset–`cdr` asks you to choose a new password and to repeat it, as the console does. Choosing a new password needs a terminal; from a script, sign in once interactively or in the console first.

If your account uses MFA, `cdr` asks for the code after the password, with the prompt `Authenticator code (or a recovery code):`. Enter the 6-digit code your authenticator app shows, or one of your recovery codes. If the code is refused, `cdr` shows why and asks again. It prompts for a code at most three times in one run. When too many wrong codes have locked authenticator codes, it asks for a `Recovery code:` instead, because recovery codes still work during the lock. You can also give the code with `--code`. Unlike the password, a code is safe on the command line: it is accepted only once, so a copy in your shell history is useless. A recovery code given with `--code` is also used up by the sign-in. MFA is turned on and off in the console, not with `cdr`. See Multi-Factor Authentication.

When there is no terminal, for example in a scheduled task, an account with MFA needs `--code`. Without it, `cdr login` stops with `This account uses MFA and there is no terminal to prompt on. Pass --code with the code your authenticator app shows.` and exits with code 1. A wrong or locked code exits with code 3, like any other refused sign-in.

After five failed attempts in a row, the account is locked for 15 minutes. A refused sign-in does not say whether the account exists, is locked, or is disabled; the audit trail records the reason. No more than 10 failed sign-ins from one address are allowed in any 15 minutes: after that, sign-ins from it are refused with `Too many failed sign-ins from this address` until the oldest failure is 15 minutes old. The break-glass account is never locked by wrong passwords, and is refused for its address only when it does not sign in on the Manager machine itself. When you sign in with the break-glass account, `cdr` warns you that

everything it does is audited as break-glass, and that you should use a named account when one is available.

Requirements: An enabled operator account on the Manager. Runs on Windows only.

Syntax

```
cdr login [options]
```

Options

Option	Description
`-u, --username <NAME>`	The operator to sign in as. Default: you are prompted for it. Required when there is no terminal.
`--code <CODE>`	For an account with MFA: the code your authenticator app shows now, or a recovery code. Default: you are prompted for it when the account needs one. Required for an account with MFA when there is no terminal.

The global options also apply. Use `--manager` to choose the Manager to sign in to.

Examples

```
cdr login
cdr login --username jsmith --manager https://10.0.0.10:5210
cdr login --username jsmith --code 492817
```

From a scheduled task, with the fingerprint and password supplied by the task's environment:

```
cdr login --username svc-monitor --manager https://10.0.0.10:5210 --manager-ca 3F2A-91C0-5B7E-04D8-C1A6-2E9F-7B30-D4C2-8E15-6A9B-F0D3-1C47-B8E2-5D90-A6F1-E7F8
```

logout

Ends your session on the Manager and removes it from this computer. The session is ended on the Manager first, so a copy of the token taken earlier stops working too. If the Manager cannot be reached, the saved session is still removed, and the session on the Manager expires on its own. If you are not signed in, the command says so and succeeds.

Requirements: None. Runs on Windows only.

Syntax

```
cdr logout [options]
```

Takes the global options only.

Example

```
cdr logout --manager https://10.0.0.10:5210
```

whoami

Shows who the Manager says this run is signed in as: the username and display name, the Manager address, the roles (or break-glass), the capabilities those roles grant, whether MFA is on and how many recovery codes are left, and when the session ends. The answer comes from the Manager, not from the saved session, so it reflects any role change made since you signed in.

Requirements: A current session.

Syntax

```
cdr whoami [options]
```

Takes the global options only.

Example

```
cdr whoami --json
```

operator list

Lists the operator accounts on the Manager, as the **Operators** tab in the console shows them: username, display name, roles (or break-glass), status, whether MFA is on, and the last sign-in. The status is `active`, `temporary password`, `locked until` a time, `disabled`, or `removed` with the date and who removed it.

The table leaves removed accounts out, and says on standard error how many it left out. `--removed` includes them. `--json` always writes every account; a removed account has a `removedAtUtc` value.

Creating, editing, disabling, and enabling operators, and resetting passwords and MFA, are done in the console. See Operators and Roles Screen.

Requirements: The `accounts.manage` capability (System Admin role).

Syntax

```
cdr operator list [--removed]
```

Option	Description
`--removed`	Include removed operators in the table.

Examples

```
cdr operator list
cdr operator list --removed
cdr operator list --json
```

operator remove

Removes the account of someone who has left. Every session the account has, in the console and in `cdr`, ends at once. Its password, MFA, and roles are cleared, not kept. The account stays in the audit trail and its username stays reserved, so nobody else can be created under it. For a temporary absence, disable the account in the console instead: that keeps its roles. For the difference, see Managing Operators and Roles.

Before it asks for confirmation, the command shows the account's display name and the roles, MFA, and disabled state being cleared. If the account is already removed, it says when and by whom, changes nothing, and exits with code 0. The Manager refuses to remove your own account or the break-glass account; `cdr` shows the Manager's message and exits with code 3.

Requirements: The `accounts.manage` capability (System Admin role).

Syntax

```
cdr operator remove <USERNAME> [--yes]
```

Argument or option	Description
`<USERNAME>`	**Required.** The operator who has left. Case does not matter.
`--yes`	Skip the confirmation. Required when standard input is not a terminal.

Exit codes: 0 when the account is removed, was already removed, or you answered no; 3 when no operator has the username or the Manager refused; 1 for a usage error, including no `--yes` without a terminal.

Examples

```
cdr operator remove jsmith
cdr operator remove jsmith --yes
```

operator restore

Restores a removed account, for the same person returning. Nothing comes back from before: the account gets a new temporary password, no MFA, and only the roles you name now. The command prints the temporary password once, on standard output; with `--json`, it is in the `temporaryPassword` field. Give it to the operator by a channel you trust. They choose their own password at their next sign-in, in the console or with `cdr login`, and turn MFA on again from the console's account menu.

The Manager refuses an account that is not removed; `cdr` shows the Manager's message and exits with code 3.

Requirements: The `accounts.manage` capability (System Admin role).

Syntax

```
cdr operator restore <USERNAME> --role <ROLE> [--role <ROLE>...] [--display-name <NAME>]
```

Argument or option	Description
`<USERNAME>`	**Required.** The removed operator who is returning. Case does not matter.
`-r, --role <ROLE>`	**Required.** A role to grant: `viewer` (Viewer / Auditor), `backupOperator`, `restoreOperator`, or `systemAdmin`. Case does not matter. Repeat the option for several roles.
`--display-name <NAME>`	A new display name. Default: the display name the account had.

Exit codes: 0 when the account is restored; 3 when no operator has the username or the Manager refused; 1 for a usage error, including no `--role` or a name that is not a role.

Examples

```
cdr operator restore jsmith --role viewer
cdr operator restore jsmith --role backupOperator --role restoreOperator
```

Inventory Commands

The `dc`, `domain`, `forest`, and `site` commands list your directory as the Manager knows it, with the verdict the Manager has reached for each object–the same verdicts the console shows. They list what the directory defines, including objects nothing is protecting yet, which is how you find the gaps. The `dc rebuild` command builds a virtual machine shaped like a domain controller's backup.

The list commands only read. To change what is counted, use exclusions (Configuration and Exclusion Commands); to record that an object has been removed from the directory, use Directory Cleanup Commands.

Every filter that takes a verdict code checks it. If you enter a code that does not exist, the command is refused and lists the accepted codes, rather than returning an empty list that looks like a healthy result.

For how to run `cdr` and the global options, see Command-Line Reference.

dc list

Lists the domain controllers the directory defines, with each one's protection verdict. A summary of how many domain controllers have each verdict follows the table.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr dc list [options]
```

Options

Option	Description
`-p, --protection <CODE>`	Only domain controllers with this verdict: `protected`, `stale` (the last good backup is older than twice the schedule interval), `failing` (the last backup failed; a backup refused before the machine's current agent enrolled does not count), `unscheduled` (no backup schedule), `unprotected` (a backup target with nothing to restore from), or `rebuild` (not a backup target: no enabled policy and nothing in the vault, so recovery rebuilds it).
`-d, --domain <FQDN>`	Only domain controllers in this domain.
`--site <NAME>`	Only domain controllers in this site.

Option	Description
`--reach <CODE>`	Only domain controllers whose reachability from the agents is `reachable`, `partial`, `unreachable`, `identity-mismatch`, or `not-probed`.
`--wide`	Add the operations master (FSMO), global catalog and read-only, and topology corroboration columns.
`--retired`	Include domain controllers recorded as removed from the directory. Default: they are left out.

Examples

```
cdr dc list --protection unprotected
cdr dc list --protection rebuild
cdr dc list --domain corp.example.com --wide
```

dc rebuild

Creates a virtual machine shaped like a domain controller's backup: the same firmware type and generation, disks of the right number and size, adapters, memory, and processors, as recorded in the backup's manifest. Use it to prepare a replacement machine—for example a candidate for a full system recovery, or a machine to receive a DC state recovery. With `--restore-run`, it builds the new VM for a full system recovery onto a new VM and attaches that run's recovery ISO. The machine is created empty. `cdr` does not install an operating system, restore anything, or promote a domain controller. It reports the original's network addressing for you to apply, but does not apply it.

By default the build uses the domain controller's newest manifest, and looks up matching Windows Server installation media in the OS media library to attach to the machine.

- **Hyper-V**—`cdr` builds the machine on the Hyper-V host where you run it. Run it in an elevated command prompt. To build on another Hyper-V host, write the build script with `--script` and run that script in an elevated PowerShell session on the host.
- **VMware ESXi**—`cdr` never builds the machine itself, because that would mean holding credentials for your vCenter. With `--hypervisor esxi` you must give `--script` (to write a PowerCLI script) or `--plan` (to see the shape). Run the script in your own PowerCLI session; it takes `-Server` and `-Credential` and prompts for what it is not given.

Options that belong to the other hypervisor are refused rather than ignored—for example `--path` with an ESXi build, or `--datastore` with a Hyper-V build.

Exit codes: 4 when the manifest does not record enough to build a matching machine, or no manifest exists for the domain controller; 6 when the build or the script write fails. If a

Hyper-V build fails or is cancelled partway, the virtual machine may be partly created; `cdr` says so.

Requirements: The `backup.view` capability (every role). For a Hyper-V build, an elevated session on a Windows Hyper-V host.

Syntax

```
cdr dc rebuild <FQDN> [options]
```

Arguments

Argument	Description
`<FQDN>`	**Required.** The domain controller whose manifest the new machine is shaped from.

Options

Option	Description
`--hypervisor <KIND>`	`hyper-v` or `esxi`. An ESXi build is only written with `--script`, never run by `cdr`. Default: `hyper-v`.
`--name <NAME>`	Name for the new virtual machine. Default: the domain controller's NetBIOS name followed by `-rebuild`.
`--switch <NAME>`	The Hyper-V virtual switch, or the ESXi port group, to connect the adapters to. The `<SWITCH>` placeholder in the command the Manager prints for a run is refused: replace it with a switch name. `Get-VMSwitch` lists them.
`--path <DIR>`	Hyper-V only. Where the virtual machine and its disks are created. Default: the Hyper-V host's own default path.
`--datastore <NAME>`	ESXi only, and required for ESXi. The datastore the virtual machine and its disks are created on.
`--vmhost <NAME>`	ESXi only, and required for ESXi. The ESXi host to place the machine on, as vCenter names it.
`--vcenter <ADDRESS>`	ESXi only. The vCenter the script connects to by default. Never a credential.
`--guest-id <ID>`	ESXi only. The VMware guest operating system identifier. Default: the generic Windows Server identifier.

Option	Description
`--iso <PATH>`	Installation media to attach, instead of looking it up in the OS media library.
`--no-iso`	Do not attach installation media, and do not look any up. Cannot be combined with `--iso`.
`--mac`	Give each adapter the MAC address the manifest recorded, instead of one the host allocates. Use this only when the original machine is off the network: two identical MAC addresses on one network segment conflict. It is useful when DHCP reservations or licensing depend on the MAC address.
`--memory <SIZE>`	Use this much memory instead of the manifest's, for example `8GB`.
`--processors <N>`	Use this many logical processors instead of the manifest's. Minimum: 1.
`--restore-disk [SIZE]`	Add a data disk for a DC state recovery to stage the backup onto. Without a size, the disk is sized from this machine's largest backup.
`--generation <N>`	State the machine's shape when the manifest could not establish it: `1` is BIOS with MBR disks, `2` is UEFI with GPT disks.
`--manifest <ID>`	Build from this manifest instead of the domain controller's newest. `cdr backup list` prints manifest IDs.
`--restore-run <RUN>`	Build the VM for a full system recovery onto a new VM: shape it from that run's restore point, copy the run's recovery ISO to this host, and attach it with the disk first in the boot order. No operating system is installed; start the VM (`--start`) and the restore runs unattended. Hyper-V only, and run on Windows. Cannot be combined with `--iso` or `--no-iso`. Fetching the media needs the `plan.execute` capability. The local copy of the ISO holds the run's enrollment token until the run ends; delete it afterwards.
`--media-dir <DIR>`	With `--restore-run`: where the recovery ISO is copied to. Default: `%ProgramData%\CygnaDR\RecoveryMedia`.

Option	Description
`--plan`	Show what would be built, and stop. Nothing is created.
`--script <PATH>`	Write the build script to this file, and stop. Nothing is created. The folder must already exist.
`--start`	Start the virtual machine once it is built.

Examples

See what would be built, then build it on this Hyper-V host with a staging disk for a DC state recovery:

```
cdr dc rebuild dc04.corp.example.com --plan
cdr dc rebuild dc04.corp.example.com --switch Recovery --restore-disk --start
```

Build and start the VM for a full system recovery onto a new VM, once the restore run has executed and its recovery ISO is built:

```
cdr dc rebuild dc04.corp.example.com --restore-run <RUN> --switch Recovery --start
```

For the procedure, see Full System Recovery onto a New VM.

Write a PowerCLI script for an ESXi build:

```
cdr dc rebuild dc04.corp.example.com --hypervisor esxi --vmhost esx03.corp.example.com --data-store vsanDatastore --switch "VM Network" --script build-dc04.ps1
```

domain list

Lists every domain that the directory defines or that a domain controller claims, with its coverage verdict: whether the domain could be recovered by restoring one writable domain controller and promoting the rest.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr domain list [options]
```

Options

Option	Description
`-c, --coverage <CODE>`	Only domains with this verdict, for example `covered`, `single-point`, `unrecoverable`, or `beyond-tombstone`. An unknown code is refused, and the accepted codes are listed.
`-f, --forest <FQDN>`	Only domains in this forest.
`--wide`	Add the NetBIOS name, functional level, SYSVOL replication, operations master, built-in Administrator account, and recovery depth columns.
`--retired`	Also list domains recorded as removed from the directory. Default: they are left out.

Examples

```
cdr domain list --coverage single-point
cdr domain list --forest corp.example.com --wide
```

The **PROTECTED** column gives the protected domain controllers out of the domain's backup targets. Domain controllers rebuilt on recovery, excluded ones, and ones removed from the directory are named beside the figure, for example `2/2 (4 rebuilt on recovery, 1 excluded)`. A domain with too few backup targets keeps the domain controllers it needs backed up in the figure and names them first, for example `0/2 (2 need a backup policy, 1 rebuilt on recovery)`. See Backup Targets and Rebuilt Domain Controllers. Excluded domain controllers stay in the domain's figures, so they reconcile with `cdr dc list`.

Excluded domain controllers that no source places in a domain are not counted, and a hint below the table says how many.

forest list

Lists every forest known from the directory, a domain, a domain controller, or a trust, with its readiness verdict and how fresh its topology is. With `--trusts`, it lists the trust inventory instead.

Domains you have excluded, other than the forest root, are left out of the forest's readiness and its domain and domain controller counts; the counts say how many with `(n excluded)`. The **PROTECTED** column gives the protected domain controllers out of the backup targets in scope, and says what the others are, for example `1/4 (2 last backup failed, 1 too old, 6 rebuilt on recovery, 1 excluded)`. Domain controllers rebuilt on recovery are not backup targets, so they are named rather than counted as a shortfall. Those a domain

needs backed up because it has too few backup targets stay in the figure and are named first, as `need a backup policy`. Excluded domain controllers with no forest are not counted, and a hint below the table says how many. See Coverage and Readiness.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr forest list [options]
```

Options

Option	Description
`-r, --readiness <CODE>`	Only forests with this verdict, for example `ready`, `root-at-risk`, or `unmanaged`. An unknown code is refused, and the accepted codes are listed. Cannot be combined with `--trusts`.
`--wide`	Add the schema, functional level, forest operations masters, global catalog, site, trust, and reporting agent columns.
`--trusts`	List the trust inventory instead: each trust's type and direction, and whether its password reset can be mirrored during a recovery.
`--retired`	With `--trusts`, include trusts recorded as removed from the directory. Default: they are left out.

Examples

```
cdr forest list --readiness root-at-risk
cdr forest list --trusts --retired --json
```

site list

Lists every site the directory defines or a domain controller names, with its coverage verdict: what could be restored back into each location. With `--subnets` or `--links`, it lists the subnet map or the site links instead.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr site list [options]
```

Options

Option	Description
`-c, --coverage <CODE>`	Only sites with this verdict, for example `covered`, `partial`, `rebuild` (domain controllers and no backup target among them), or `empty`. An unknown code is refused, and the accepted codes are listed. Applies to the site list only.
`-f, --forest <FQDN>`	Only sites, subnets, or site links in this forest.
`--wide`	Add the location, domain, operations master, concentration, and site link columns. Applies to the site list only.
`--subnets`	List the subnet map instead, including subnets assigned to no site.
`--links`	List the site links instead, with their cost, replication interval, and member sites. Cannot be combined with `--subnets`.

Examples

```
cdr site list --coverage partial
cdr site list --forest corp.example.com --subnets
```

The **PROTECTED** column gives the protected domain controllers out of the site's backup targets, with those rebuilt on recovery and excluded ones named beside it, for example `1/1 (3 rebuilt on recovery)`. A site with domain controllers and no backup target reads `no backup targets`.

Excluded domain controllers placed in no site are not counted, and a hint below the table says how many.

Agent and Trust Commands

The `agent` commands list the agents that have enrolled with the Manager, bring agents of every kind up to the build the Manager holds, restart and uninstall agents through the Manager, install the DC agent on domain controllers, save agent packages, report what automatic updates are doing, and delete the record of an agent whose machine is gone. The `recovery-host` commands show whether each Isolated Recovery Host can run test restores, and approve, drain, and undrain hosts. The `trust` commands manage the Manager CA fingerprints that `cdr` has saved for your Windows account on this computer.

An agent list and a domain controller list answer different questions. `cdr agent list` shows what has enrolled; `cdr dc list` shows what the directory defines, including domain controllers with no agent. Use `cdr agent deploy --without-agent` to close the gap.

For how to run `cdr` and the global options, see Command-Line Reference.

agent list

Lists the enrolled DC agents, storage agents, and Isolated Recovery Host agents: which build each runs, and whether the Manager is talking to it now. After a storage agent's kind, the list says which is the storage machine (`(storage machine)`) and marks any other as `(candidate, no work)` or `(replaced)`. After an Isolated Recovery Host agent's kind, it marks a host `(awaiting approval)` or `(draining)`; an approved host has no mark.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr agent list [options]
```

Options

Option	Description
`-k, --kind <CODE>`	Only agents of this kind: `dc`, `storage`, or `host`.
`-M, --domain <FQDN>`	Only agents on machines in this domain. The short form is `-M`.
`--offline`	Only agents with no session open to the Manager right now.
`--outdated`	Only agents running a build older than the package for their kind in the Manager's package folder.
`--silent-for <MINUTES>`	Only agents whose last heartbeat is older than this many minutes, or that never sent one.
`--wide`	Add the agent ID, NetBIOS name, forest, site, enrollment date, and a **CAPABILITIES** column with what each connected agent's session says it can do. The column shows a dash for an agent with no session. The agent ID is what `cdr config set --agent`, `cdr restore plan --onto`, and `cdr agent forget` take.

Examples

```
cdr agent list --outdated
cdr agent list --offline --silent-for 60
cdr agent list --kind storage --wide
```

agent forget

Deletes an agent's record from the Manager, for a machine that is gone. It touches no machine. Without `--confirm`, it only says what it would delete: the agent's kind, machine, build, enrollment date, and last heartbeat, and any other record under the same name, which stays.

The Manager refuses, and the command lists why, while any of these is true:

- The agent is connected to the Manager right now.
- A restore run that has not finished names it.
- An approved, running, or halted recovery plan has bound it as a candidate machine.
- A test restore that has not finished runs on it.
- It is the storage machine. Replace it with another storage agent first, with `cdr storage replace`.

The record's session history and backup jobs are kept, and the audit trail keeps a copy of the record. If the machine is still running, its agent enrolls again under its own name on its next connection. On the **Agents** screen, **Forget this agent record…** in the agent's row menu does the same.

Requirements: The `agents.manage` capability (System Admin role).

Syntax

```
cdr agent forget <ID> [--confirm <ID>]
```

Argument or option	Description
`<ID>`	**Required.** The agent ID whose record to delete, as `cdr agent list --wide` shows it.
`--confirm <ID>`	Type the agent ID again to delete its record. Without it, `cdr` only says what would happen.

Examples

```
cdr agent forget 0847c3af-ec9c-4389-bcb1-7b706c5dc164
cdr agent forget 0847c3af-ec9c-4389-bcb1-7b706c5dc164 --confirm 0847c3af-ec9c-4389-bcb1-7b706c5dc164
```

agent update

Brings agents up to the build the Manager holds. The Manager chooses the route for each agent. An agent that is connected and able to update itself is sent the package over its own session and installs it after checking it (**self-update**). A DC agent that cannot update itself is installed remotely with an account you give (**push install**). A storage agent or Isolated Recovery Host agent that cannot update itself must be installed by hand (**hand install**). Before it starts, the command prints each agent with its current and new version and the route. It then follows the run and prints each agent's progress.

Requirements: The `agents.manage` capability (System Admin role).

Syntax

```
cdr agent update [<AGENTS>...] [options]
```

Arguments and options

Argument or option	Description
`<AGENTS>`	Agents to update, by agent ID (`cdr agent list --wide` shows it), or by machine name when that names only one agent. Name agents or use `--outdated`, not both.
`--outdated`	Every agent behind the package the Manager holds for its kind. Agents that must be installed by hand, and DC agents that would need a push when you give no `--user`, are skipped with a warning.
`-k, --kind <CODE>`	With `--outdated`, only agents of this kind: `dc`, `storage`, or `host`.
`-u, --user <ACCOUNT>`	Only for a DC agent that cannot update itself and is pushed to: a local administrator on the domain controller, as `DOMAIN\user`. The password is read from the `CYGNADR_DEPLOY_PASSWORD` environment variable, or prompted for.
`--transport <NAME>`	For a push install: `winrm` or `dcom`. Without it, the Manager's configured default is used.
`--wait <MINUTES>`	How long to follow the run before leaving it to carry on on the Manager. Default: 30.

When the run ends, the command prints a table with the columns **AGENT**, **KIND**, **ROUTE**, **OUTCOME**, **BEFORE**, **AFTER**, and **MESSAGE**. The Manager refuses to update an agent

while it is busy, for example while its domain controller is being backed up or a test restore runs on its host; the message says so.

Exit codes: 0 when every agent was updated or was already current; 7 when any agent refused, failed, or needs a hand install; 8 when the run was still going when `--wait` ran out; 4 when nothing could be selected because every candidate was skipped; 3 when the Manager did not start the run; 1 for a usage error.

Examples

```
cdr agent update --outdated
cdr agent update --outdated --kind storage
cdr agent update storage01
cdr agent update --outdated --kind dc --user CORP\admin
```

agent auto-update

Says what automatic agent updates are doing: whether they are on, the interval and how many agents are updated at once, when the Manager last checked and will next, and how many outdated agents are updating, queued, and waiting. It then lists the agents that need an operator: those that cannot update themselves and those whose last automatic update failed. Turning automatic updates on and off is a setting: `cdr config set CygnaDR:Deployment:AutoUpdate true`. See Updating agents automatically.

Requirements: The `environment.view` capability (every role). `--check-now` needs `agents.manage` (System Admin role).

Syntax

```
cdr agent auto-update [options]
```

Options

Option	Description
`--all`	List every outdated agent and why it is waiting, not only those that need an operator.
`--check-now`	Check now rather than at the interval, starting updates for the agents that can take one. Starts nothing while automatic updates are off.

The table has the columns **MACHINE**, **KIND**, **RUNNING**, **PACKAGE**, **STATE**, and **WHY**. The state is `started`, `updating`, `queued`, `busy, later`, `away, later`, `next check`, `needs an operator`, or `failed, retrying later`, with when a failed update is tried again.

Exit codes: 4 when an outdated agent needs an operator, because it cannot update itself or its automatic update failed; 0 otherwise; 3 when a `--check-now` did not run. With `--json`, the counts cover every outdated agent, and the list holds those that need an operator, or every one with `--all`.

Examples

```
cdr agent auto-update
cdr agent auto-update --all
cdr agent auto-update --check-now
```

agent restart

Restarts agents' services through the Manager, so settings that take effect at restart come into force without signing in to each machine. The console's **Agents** and **Configuration** screens send the same request.

The Manager refuses an agent that is not connected, an agent that is not running as its service, and an agent whose work a restart would cut short. That work is, for a DC agent, a backup or restore on its domain controller, or any running recovery plan; for the storage agent, any backup or restore; for an Isolated Recovery Host agent, a test restore on its host.

Requirements: The `agents.manage` capability (System Admin role).

Syntax

```
cdr agent restart [<AGENTS>...] [options]
```

Arguments and options

Argument or option	Description
`<AGENTS>`	Agents to restart, by agent ID (`cdr agent list --wide` shows it), or by machine name when that names only one agent. Name agents or use `--restart-pending`, not both.
`--restart-pending`	Every connected agent holding a saved setting that takes effect at restart. An agent that is not running as its service is skipped with a warning.
`-k, --kind <CODE>`	With `--restart-pending`, only agents of this kind: `dc`, `storage`, or `host`.

The command prints a table with the columns **AGENT**, **KIND**, **OUTCOME**, and **DETAIL**. The outcome is `restarting` when the agent has scheduled its restart; it reconnects within

a few seconds, and `cdr agent list` shows its new session. Any other outcome, such as `not-connected`, `not-supported`, `busy`, `refused`, or `no-answer`, comes with the reason.

Exit codes: 0 when every agent is restarting, or when `--restart-pending` finds no agent waiting; 7 when any agent was not restarted; 4 when every agent waiting for a restart was skipped because it is not running as its service; 3 when the Manager did not restart anything; 1 for a usage error.

Examples

```
cdr agent restart --restart-pending
cdr agent restart --restart-pending --kind dc
cdr agent restart dc01.corp.example.com storage01
```

agent uninstall

Has an agent uninstall itself from its machine, as the row menu on the console's **Agents** screen does. Without `--confirm`, the command only says what it would do; give the agent ID again with `--confirm` to uninstall it.

The Manager refuses a DC agent on a domain controller that is neither retired nor excluded, an agent that is not connected, any agent with work in flight, and the storage machine, which is replaced with `cdr storage replace` instead. Retire the domain controller with `cdr cleanup retire` or exclude it with `cdr exclusion add` first. The agent itself refuses when it was not installed by its own package or its machine's `AllowRemote` policy is `false`. The agent's record stays on the Manager; once it shows as disconnected, delete it with `cdr agent forget`.

Requirements: The `agents.manage` capability (System Admin role).

Syntax

```
cdr agent uninstall <AGENT> [--confirm <ID>]
```

Argument or option	Description
`<AGENT>`	**Required.** The agent to uninstall, by agent ID, or by machine name when that names only one agent.
`--confirm <ID>`	Type the agent ID to uninstall the agent. Without it, `cdr` only says what would happen.

When the agent accepts, the command prints what it is removing and the path of the installer log on the machine. Exit codes: 0 when the agent is uninstalling, or when nothing was confirmed; 7 when the Manager or the agent refused; 3 when the request failed; 1 for a usage error, including a `--confirm` value that is not the agent's ID.

Examples

```
cdr agent uninstall spare01.corp.example.com
cdr agent uninstall 0847c3af-ec9c-4389-bcb1-7b706c5dc164 --confirm 0847c3af-ec9c-4389-bcb1-7b706c5dc164
```

agent package download

Saves an agent installer that the Manager holds, and checks the saved file against the package's SHA-256 hash. If the hashes differ, the file is removed. An existing file is never overwritten.

Requirements: The `agents.manage` capability (System Admin role).

Syntax

```
cdr agent package download <KIND> [-o|--output <PATH>]
```

Argument or option	Description
`<KIND>`	**Required.** `dc`, `storage`, or `host`.
`-o, --output <PATH>`	A file name, or a folder to save the package in under its own name. Default: the current folder.

Examples

```
cdr agent package download storage
cdr agent package download host --output C:\Install
```

recovery-host list

Lists each Isolated Recovery Host, whether it is connected, and what stops it running test restores.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr recovery-host list [--json]
```

The table has the columns **HOST**, **USE** (`approved`, `awaiting approval`, `draining`, or `drained`), **STATE** (`ready`, `not ready`, or `unknown`), **CONNECTED**, **HYPER-V** (for example `running`, `not installed`, or `restart needed`), **SWITCHES**, **FREE FOR VMS**, **MEMORY**, and **REPORTED**. Under each host, it shows the host's agent ID; for a host

awaiting approval or draining, what to do next; the folder for test restore virtual machines and the recovery network; the test restore running now, the last one and its result, and what the schedule last decided for this host; and each problem or warning the host reported, with what to do about it.

With no host enrolled, the command says so. Test restores are optional; install the Isolated Recovery Host agent on a Hyper-V host to offer them. See Setting Up an Isolated Recovery Host.

recovery-host approve, drain, and undrain

Decide whether test restores go to a host. `approve` approves a host for test restores; a host is given none until it is approved. `drain` stops new test restores going to a host so that it can be removed; the test restore running on it finishes, and `cdr recovery-host list` says `drained` once nothing runs on it. `undrain` puts a draining or drained host back in use. See Moving an Isolated Recovery Host.

Requirements: The `agents.manage` capability (System Admin role). Each action is recorded in the audit trail.

Syntax

```
cdr recovery-host approve <ID>
cdr recovery-host drain <ID>
cdr recovery-host undrain <ID>
```

`<ID>` is the host agent's ID, as `cdr recovery-host list` shows it under each host. The command says what changed, or why the Manager refused, and exits with code 3 when nothing was changed.

Examples

```
cdr recovery-host approve 3f2a91c4-0d7e-4b8a-9c1f-5e6d7a8b9c0d
cdr recovery-host drain 3f2a91c4-0d7e-4b8a-9c1f-5e6d7a8b9c0d
```

agent deploy

Installs or upgrades the DC agent on domain controllers from the computer where you run `cdr`. Use it for a first install; to update agents that are already installed, `cdr agent update` is simpler, because agents that can update themselves need no credential. Name the domain controllers, select them from the Manager's inventory with `--all`, `--without-agent`, or `--outdated`, or combine both.

For each target, `cdr` connects with the account you give in `--user`, copies the package over the `ADMIN$` share, reads the installed version from the registry, and runs the installer through WinRM or DCOM. It decides per target whether to install, upgrade, or leave the

agent alone. A target already at the package's version is reported as current, and a downgrade is refused, unless you add `--force`. The password is used for this run only; it is never sent to the Manager or stored.

A new agent needs the Manager's agent address, the Manager CA fingerprint, and the enrollment token:

- **Address**–`cdr` uses `--manager-address`, or derives it from `--manager` with port 5211. It cannot derive one from `localhost`, so when you run `cdr` on the Manager, give `--manager-address` with the address the domain controllers will reach, for example `https://10.0.0.10:5211`. Use the Manager's IP address: the Manager is usually not in the domain controllers' DNS, and the agents must reach it when DNS is down.
- **Fingerprint**–`cdr` seeds each agent with the fingerprint that this run trusts: `--manager-ca`, the fingerprint you saved for the Manager, or a confirmation on first use. It checks the fingerprint against what the Manager's agent port presents, so the agents trust exactly the Manager you reached.
- **Enrollment token**–Read from the `CYGNADR_ENROLLMENT_TOKEN` environment variable, or prompted for. It is written only where no agent is installed yet, and, with `--repoint`, to every target. Treat the token as a secret: anyone who has it can enroll an agent.

Upgrades need no token. To move existing agents to a new Manager address, add `--repoint` with `--manager-address`. With `--repoint`, the enrollment token is optional: one you give is written to every target, which is how agents get a new token after the Manager is re-keyed; with none, each agent keeps its own. When you name the targets and use no selector, `cdr` does not need the Manager's inventory, so you can re-point agents at a Manager that is not answering yet.

The command prints a table of each target's action, version before and after, and result, and exits with code 7 if any target failed.

Requirements: An account that is a local administrator on each target, given as `DOMAIN\user` or `user@domain`. Each target must allow the `ADMIN$` share and the Remote Registry service, and either WinRM (TCP 5985) or DCOM (TCP 135 and the dynamic RPC range, with the Windows Management Instrumentation (WMI-In) firewall rules enabled). Over WinRM from a computer outside the domain, the targets must be in that computer's TrustedHosts list; DCOM needs no TrustedHosts entry. The selectors read the inventory, which needs the `environment.view` capability. Runs on Windows only. For details, see Deploying DC Agents.

Syntax

```
cdr agent deploy [TARGET ...] --user <ACCOUNT> [options]
```

Arguments

Argument	Description
`[TARGET ...]`	Domain controllers to deploy to, by name. Combine with the selectors, or use instead of them. Either a target or a selector is required.

Options

Option	Description
`-u, --user <ACCOUNT>`	**Required.** An account that is a local administrator on the targets, as `DOMAIN\user` or `user@domain`. The password is read from `CYGNADR_DEPLOY_PASSWORD`, or prompted for.
`--package <MSI>`	The `CygnaDR.DcAgent.msi` to deploy. Default: the package in the `redist` folder beside `cdr.exe`, then one directly beside it.
`--manager-address <URL>`	The agent address a new agent connects to, for example `https://10.0.0.10:5211`. Default: derived from `--manager`; required when that is `localhost`.
`--repoint`	Also write `--manager-address`, the CA fingerprint, and the enrollment token to targets that already have an agent, so that they follow the Manager to a new address, or to a new CA and token after a re-key. With no token given, each agent keeps its own. Requires `--manager-address`.
`--all`	Every domain controller in the inventory.
`--without-agent`	Every domain controller in the inventory that has no agent.
`--outdated`	Every domain controller whose agent reports a version older than the package's.
`-d, --domain <FQDN>`	Restrict the selectors to one domain. Only with `--all`, `--without-agent`, or `--outdated`.
`--transport <NAME>`	How the installer is started on the target: `winrm` (TCP 5985) or `dcom` (TCP 135 and RPC; needs no TrustedHosts entry). Default: `winrm`.
`--parallel <N>`	How many targets to work on at the same time. Default: 4.
`--install-timeout`	How long the installer may run on a target. Default: 10.

Option	Description
`<MINUTES>`	
`--force`	Reinstall on a target already at this version, or attempt a downgrade.
`--plan`	Connect, read each target's installed version, and report what would happen. Nothing is changed, and no token or fingerprint is needed.

Examples

See what would happen across the inventory, then install on every domain controller in one domain that has no agent, from the Manager:

```
cdr agent deploy --all --user CORP\admin --plan
cdr agent deploy --without-agent --domain corp.example.com --user CORP\admin --manager-address https://10.0.0.10:5211
```

Upgrade every outdated agent, and reinstall one domain controller over DCOM:

```
cdr agent deploy --outdated --user CORP\admin
cdr agent deploy dc02.corp.example.com --user CORP\admin --transport dcom --force
```

After the Manager moves to a new address, point every agent at it:

```
cdr agent deploy --all --user CORP\admin --manager-address https://10.0.0.20:5211 --repoint --force
```

trust list

Lists the Managers whose CA fingerprint is saved for your Windows account, with the fingerprint and when it was saved. It reads only the local file and contacts no Manager, so it works when every Manager is down.

Requirements: None.

Syntax

```
cdr trust list [options]
```

Takes the global options only.

Example

```
cdr trust list
```

trust pin

Trusts the Manager at `--manager` and saves its fingerprint. With `--manager-ca` or `CYGNADR_MANAGER_CA`, `cdr` checks that the Manager presents that fingerprint and saves it without asking—the way to prepare a computer or account that will run scripts. Without one, `cdr` shows the fingerprint and asks you to confirm it, even if one is already saved for the address.

Requirements: None. Without a fingerprint, a terminal to confirm on.

Syntax

```
cdr trust pin --manager <URL> [--manager-ca <FINGERPRINT>]
```

Takes the global options only.

Examples

```
cdr trust pin --manager https://10.0.0.10:5210
cdr trust pin --manager https://10.0.0.10:5210 --manager-ca 3F2A-91C0-5B7E-04D8-C1A6-2E9F-7B30-D4C2-8E15-6A9B-F0D3-1C47-B8E2-5D90-A6F1-E7F8
```

trust remove

Forgets the saved fingerprint for the Manager at `--manager`, so that the next connection asks you to confirm the CA again. Use it when a Manager has been deliberately rebuilt or its CA re-keyed and now presents a different CA. If nothing is saved for the address, the command warns you and exits with code 1.

Note: A Manager that presents a different CA when you did not change it is a warning sign, not something to clear. Confirm what happened before you remove the saved fingerprint.

Requirements: None.

Syntax

```
cdr trust remove --manager <URL>
```

Takes the global options only.

Example

```
cdr trust remove --manager https://10.0.0.10:5210
```

For re-keying, see Re-Keying the Manager CA.

Backup Commands

The `policy` commands manage each domain controller's backup schedule. The `backup` commands list and inspect restore points and start a backup outside the schedule. The `jobs` commands follow backup jobs and cancel one that is running.

For how backups are scheduled and stored, see How Backup Works and Backup Types. For how to run `cdr` and the global options, see Command-Line Reference.

Backup types are given as codes: `bmr` for a full system backup, which supports both DC state recovery and full system recovery, and `systemstate` for a DC state backup, which supports DC state recovery only.

policy list

Lists the backup schedules this installation holds: each machine's interval, start time, backup type, whether the schedule is enabled, and when it fires next. The **STARTS** column shows the start time in the Manager's local time, or `any time` for a schedule that runs on its interval alone. When a listed schedule has a start time, a hint under the table says that it is the earliest each backup is due, and that with many domain controllers on one time the last start later.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr policy list [options]
```

Options

Option	Description
`-M, --machine <FQDN>`	Only the schedule for this machine.
`-t, --type <CODE>`	Only schedules of this backup type: `bmr` (Full system) or `systemstate` (DC state).
`--disabled`	Only schedules that exist but are switched off.
`--due`	Only schedules the Manager would start on its next check.
`--dc-state-only`	Only machines with DC state backups, which require an OS that still boots and cannot do a full system recovery.

Examples

```
cdr policy list --due
cdr policy list --dc-state-only
cdr policy list --machine dc01.corp.example.com --json
```

For a machine whose DC agent has never come online, the next-due time reads `waiting for agent`: the schedule is held, not failing, and its first backup runs as soon as the agent connects. Such a schedule is not listed by `--due`.

policy set

Creates or changes one machine's backup schedule. `cdr` reads the current schedule first, so an edit changes only what you give and keeps the rest. The machine does not need an agent yet, so you can schedule a domain controller before you deploy its agent. The command then says that no backup runs until a DC agent is installed and connects, and when the first backup then runs: at the schedule's start time if that is still ahead, or as soon as the agent connects if that is later. With no start time ahead, it runs as soon as the agent connects.

The interval is in minutes, from 5 minutes to 129,600 minutes (90 days). A new schedule takes full system backups, is enabled, and has no start time, unless you say otherwise. Without a start time, the Manager spreads backups around the interval.

`--at` sets a start time: 24-hour, in the Manager's local time. It is the earliest each backup is due; the backup then waits for a free backup slot. The interval must run at the same times every day: a whole number of days, or one that divides a day evenly (every 1, 2, 3, 4, 6, 8, or 12 hours, or a divisor of an hour). Otherwise `cdr` refuses the change before it reaches the Manager. This also applies when you change only the interval of a schedule that has a start time; the message then names the start time and tells you to give `--at none` or another interval. The before-and-after lines and the result show the start time, for example `every 1d from 22:00 (manager time)`. After saving an enabled schedule with a start time, `cdr` asks the Manager how late the domain controllers on that time should start and prints a warning when they share it with more than **Maximum concurrent backups** allows at once. The schedule is saved either way.

Changing a schedule to DC state takes away full system recovery for that machine: if it is wiped or replaced, it must be rebuilt rather than restored. `cdr` warns you and asks for confirmation when an edit makes that change. Changing only the interval of a schedule that is already DC state does not ask again. In a script, pass `--yes`; without it, the change is refused when there is no terminal.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr policy set <FQDN> [options]
```

Arguments

Argument	Description
`<FQDN>`	**Required.** The machine to schedule.

Options

Option	Description
`-e, --every <MINUTES>`	How often to back up, in minutes. Required when you create a schedule. Daily is `1440`.
`-t, --type <CODE>`	`bmr` (Full system) or `systemstate` (DC state). Default: `bmr` on a new schedule; unchanged on an edit.
`--at <TIME>`	The earliest each backup starts, 24-hour, in the Manager's local time, for example `22:00`. `none` removes the start time, so the schedule runs on its interval alone. Unchanged on an edit when not given.
`--disabled`	Create the schedule switched off, or switch an existing one off. The interval, start time, and type are kept for later.
`--enabled`	Switch an existing schedule back on. Cannot be combined with `--disabled`.
`--yes`	Skip the confirmation that a DC state-only schedule removes full system recovery.

Examples

```
cdr policy set dc01.corp.example.com --every 1440
cdr policy set dc01.corp.example.com --every 720 --type bmr
cdr policy set dc01.corp.example.com --every 1440 --at 22:00
cdr policy set dc01.corp.example.com --at none
cdr policy set dc01.corp.example.com --disabled
cdr policy set dc01.corp.example.com --enabled
```

Exit codes: 0 when the schedule is saved, or when you decline the DC state confirmation (nothing is saved); 1 for an invalid option, such as a malformed `--at` time, an interval that cannot keep the start time, or a missing `--every` on a new schedule; 3 when the Manager rejects the schedule, with its reasons.

policy remove

Removes a machine's schedule entirely. This is not the same as disabling it: a disabled schedule keeps its interval and type, and a removed one does not. Removing a schedule deletes no backups. Afterward, the domain controller counts as unscheduled.

The command asks for confirmation. In a script, pass `--yes`; without it, the command is refused when there is no terminal.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr policy remove <FQDN> [--yes]
```

Arguments and options

Argument or option	Description
`<FQDN>`	**Required.** The machine whose schedule is removed.
`--yes`	Skip the confirmation. Required when standard input is not a terminal.

Example

```
cdr policy remove dc09.corp.example.com --yes
```

backup list

Lists the restore points this installation holds, with each one's build gate: whether the operating system build recorded in the backup still matches the machine running under that name. The **ID** column is the manifest ID that `cdr backup show` and `cdr dc rebuild --manifest` take. The list does not show the backup's version identifier; `cdr backup show` does.

Backups that retention has pruned are not restore points, because their content is gone. The list leaves them out and says how many it left out; `--include-pruned` lists them, marked **pruned**, as history. A restore point taken while a recovery of its forest was under way is marked **during recovery**, because it holds the directory part-way through that recovery. A line under the table names the recovery plan and the last step it had completed.

The **TAKEN** column also shows the result of each version's latest test restore: **verified** when it passed, **restore failed** when it did not, and **verifying** while one is running. A version

never tested, or whose latest test restore ended without a verdict, has no tag. See Test Restores.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr backup list [options]
```

Options

Option	Description
`-M, --machine <FQDN>`	Only restore points for this machine.
`--newest`	Only each machine's newest restore point, one row per domain controller.
`--rebuildable`	Only restore points whose manifest records the hardware details that `cdr dc rebuild` needs.
`--gate-failing`	Only newest restore points whose build no longer matches the machine running under that name.
`--roles`	Show which domain controller has held each operations master role, as the backups record it, instead of the list. Cannot be combined with `--newest`, `--rebuildable`, or `--gate-failing`.
`--include-pruned`	Also list backups that retention has pruned, marked as such. Their content is gone, so they are history, not restore points; without this option they are counted and left out.
`--wide`	Add the domain, hardware shape, agent build, and content hash.

Examples

```
cdr backup list --newest
cdr backup list --gate-failing
cdr backup list --machine dc01.corp.example.com --wide
```

backup show

Shows one restore point's manifest in full: the machine's identity, operating system, hardware, directory roles, and backup details. This is the record a full system recovery is planned from. The **version id** row is the backup's version identifier, which `cdr restore plan --version` and `cdr recovery plan --restore` take.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr backup show <ID> [options]
```

Arguments and options

Argument or option	Description
`<ID>`	**Required.** The manifest ID, as `cdr backup list` prints it.
`--disks`	Add the disk and volume tables.
`--network`	Add the adapter and persistent route tables.
`--raw`	Write the stored manifest to standard output byte for byte. This is the document the content hash covers. Cannot be combined with `--disks`, `--network`, or `--json`.

Examples

```
cdr backup show 412 --disks --network
cdr backup show 412 --raw > dc01-manifest-412.json
```

backup start

Starts a backup of one domain controller now, outside its schedule. Without `--wait`, the command returns as soon as the Manager has queued the backup. With `--wait`, it follows the job until it finishes and exits on the outcome.

The Manager refuses the backup when the domain controller's agent or the storage agent is not connected (exit code 4), or when a backup of that machine is already queued or running, a recovery is working on the machine, or the Manager has no active license (exit code 3). A backup refused because a recovery is working on the machine starts by itself when the recovery is done; the Recovery Plans screen shows the plans, and `cdr restore list` the restore runs. With `--wait`, a backup that fails exits with code 4, and a job whose outcome cannot be followed exits with code 8.

Requirements: The `backup.run` capability (Backup Operator role).

Syntax

```
cdr backup start <FQDN> [options]
```

Arguments and options

Argument or option	Description
`<FQDN>`	**Required.** The domain controller to back up.
`-t, --type <CODE>`	`bmr` (Full system) or `systemstate` (DC state). A DC state backup cannot do a full system recovery. Default: `bmr`.
`--wait`	Follow the job until it finishes, and exit on its outcome.
`--poll <SECONDS>`	How often to check the job while `--wait` follows it. Default: 15.

Examples

```
cdr backup start dc01.corp.example.com
cdr backup start dc01.corp.example.com --type systemstate --wait
```

jobs list

Lists recent backup jobs and their outcomes, newest first. The **JOB** column shows a short form of each job ID, which `cdr jobs cancel` accepts. A backup still waiting for a backup slot has the status `queued`, no start time, and `waiting for a slot` with how long it has waited in place of a duration.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr jobs list [options]
```

Options

Option	Description
`-M, --machine <FQDN>`	Only jobs for this machine.
`-s, --status <STATUS>`	Only jobs with this exact status, for example `running`, `completed`, or `failed`.
`--running`	Only jobs still in progress: `queued`, `created`, `staging`, `running`, or `committing`.
`--failed`	Only jobs that finished as anything other than completed or cancelled. Cannot be combined with `--running`.
`--detail`	Add the staging path, the agent's log path, and the full error message. For a backup that was not committed and whose content the storage agent has since reported in neither staging nor the vault, a `content gone` line says so.
`-n, --limit <N>`	How many matching jobs to show, newest first, from 1 to 200. Default: 50.

Examples

```
cdr jobs list --running
cdr jobs list --failed --detail
cdr jobs list --machine dc01.corp.example.com --json
```

jobs cancel

Asks the domain controller's agent to stop a running backup, or cancels a queued one at once. A queued backup is reported as cancelled before it started, because it was still waiting for a backup slot. Give the full job ID or the short form `cdr jobs list` prints; a short form must match exactly one job. A job that has already finished cannot be cancelled. If the agent is not connected, the Manager cannot ask it to stop: the backup keeps running, and the command exits with code 4.

The command asks for confirmation. In a script, pass `--yes`; without it, the command is refused when there is no terminal.

Requirements: The `backup.run` capability (Backup Operator role).

Syntax

```
cdr jobs cancel <JOB> [--yes]
```

Arguments and options

Argument or option	Description
`<JOB>`	**Required.** The job ID, or the short form `cdr jobs list` prints.
`--yes`	Skip the confirmation. Required when standard input is not a terminal.

Examples

```
cdr jobs cancel 4f3a1c9e
cdr jobs cancel 4f3a1c9e-2b70-4d51-9a3e-6c8b1f0d2e44 --yes
```

For the console equivalent, see Monitoring and Cancelling Backup Jobs.

Storage and Media Commands

The `repository` commands report on the vault and estimate how large it needs to be. They show its capacity, how much history it holds for each machine, what retention has deleted, which finished backups did not reach it, and whether committed backups were changed outside retention. The `storage replace` command makes a candidate storage agent the storage machine, and the `request` commands list and decide actions that wait for a second operator. The `os-media` command reports whether the Windows Server installation media you supplied covers the operating systems your domain controllers run, which decides whether each one can use full system recovery.

For how to run `cdr` and the global options, see Command-Line Reference.

repository status

Shows each storage agent's vault capacity and headroom, with its capacity verdict and the retention settings behind it. The vault view also shows what each storage agent is hashing now, under **hashing now**: for each hash, the storage agent, what it is for (`commit`, the scheduled `verify` pass, or `restore` with the run it is for), the backup, the machine, and progress with rate and time left. When nothing is being hashed, it says so. Hashes that run together on one storage agent share its disk and processor, so each runs slower. The vault view also notes how many domain controllers removed from the directory, with nothing in the vault, are left out; `cdr dc list --retired` names them. With `--machines`, it shows the retention depth for each machine instead; with `--prunes`, the recent prune reports.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr repository status [options]
```

Options

Option	Description
`-c, --capacity <CODE>`	Only vaults with this verdict: `ample`, `tight` (no room for another version), `full` (refusing backups), `unreadable` (the volume could not be measured), or `unreported`. Applies to the vault view only.
`--machines`	List retention depth per machine instead: versions held, size, and the oldest and newest version.
`--no-restore-point`	With `--machines`, only machines with no restorable version at all.
`--prunes`	List recent prune reports instead: what was deleted, by which retention rule, and how much space it freed. Cannot be combined with `--machines`.

Examples

```
cdr repository status
cdr repository status --machines --no-restore-point
cdr repository status --prunes
```

For what the figures mean, see Repository and Retention.

repository uncommitted

Lists the backups a domain controller finished that the Manager did not see reach the vault, with what the Manager last found of each on the storage machine. The Manager checks every hour and whenever the storage agent connects: content still in staging is committed, content already in the vault is recorded, and content that is gone stops being counted.

The command exits with code 4 while any such backup remains, and with code 3 when the Manager did not run the check.

Requirements: The `backup.view` capability (every role); `--check` needs `backup.run` (Backup Operator role).

Syntax

```
cdr repository uncommitted [--check]
```

Options

Option	Description
`--check`	Check the storage machine now instead of waiting for the hourly check, and print what it committed, recorded, or found gone.

Examples

```
cdr repository uncommitted
cdr repository uncommitted --check
cdr repository uncommitted --json
```

repository changes

Lists changes to committed backups that retention did not make, and whether each storage machine is watching its vault for them. For each change: the machine, the version, what was done (`delete`, `rename`, `write`, `truncate`, `security`, `attributes`, `create`, or `changed-unobserved` for a change found by comparing the vault with its commit records), how many paths, when, who (where the storage machine's audit policy recorded it), and whether it is open or reviewed. Above the table, each storage machine is listed as watching or not watching, with the reason. The open changes are the ones the **Committed vault content changed outside retention** alert names.

The command exits with code 4 while any change is open or any storage machine is not watching.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr repository changes [list] [options]
```

Options

Option	Description
`--all`	Include changes already reviewed, newest review first.
`--paths`	List the sample of changed paths and who changed them under each version.

Examples

```
cdr repository changes
cdr repository changes --paths
cdr repository changes list --all --json
```

repository changes review

Records that a version's changes were reviewed, which clears the **Committed vault content changed outside retention** alert for that version. The note is recorded with your name in the audit trail. A later change to the same version raises the alert again.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr repository changes review <VERSION> --note <TEXT> [--storage <FQDN>] [--yes]
```

Arguments and options

Argument or option	Description
`<VERSION>`	The version as `cdr repository changes` lists it (`machine\stamp_job`).
`-n, --note <TEXT>`	What the review found. Required.
`--storage <FQDN>`	The storage machine, when more than one vault holds a version of that name.
`--yes`	Do not ask for confirmation. Required when input is not a terminal.

Example

```
cdr repository changes review dc01.corp.example.com\20261021-020000_ab12cd34 --note
"Restore test by J. Smith"
```

repository estimate

Estimates the vault size a forest needs, by measuring its domain controllers. It needs no Manager, and nothing needs to be installed, so you can run it while you plan the deployment. It writes nothing anywhere: no directory change, no agent, no Manager record.

The command reads the forest's list of domain controllers from the directory, then reads each domain controller's volumes over CIM. For each one, the backup size is the used space on its critical volumes, less the page file, hibernation file, and shadow copy storage, plus an image overhead. It then applies the retention settings (how many versions are kept per domain controller and for how long) and the number of backups that run at once. It reports a required size and a recommended size that adds an allowance for growth. Every backup is a full backup, so each version kept counts in full.

Unless you give other values, the retention and concurrency figures are the product defaults, the same ones the storage agent starts with. If it could not measure a domain controller, the command says so and exits with code 7; if it found no domain controllers, it exits with code 4.

Requirements: Run on a Windows computer that is joined to a domain in the forest, as an account that is an administrator on the domain controllers, or give such an account with `--user`. The domain controllers are reached with Kerberos, over WinRM (TCP 5985) or DCOM (TCP 135 and the dynamic RPC range). Does not need the Manager or an operator session. For how to use the result, see Sizing Backup Storage.

Syntax

```
cdr repository estimate [options]
```

Options

Option	Description
`--dc <FQDN>`	Measure only this domain controller. Repeat the option for more. Default: every domain controller in the forest. Cannot be combined with `--domain`, `--server`, or `--per-domain`.
`-d, --domain <FQDN>`	Measure only the domain controllers of this domain.

Option	Description
`-s, --server <DC>`	The domain controller to read the forest's list of domain controllers from. Default: one the DC locator finds for this computer's domain.
`-u, --user <ACCOUNT>`	An administrator on the domain controllers. Give the domain by its DNS name, as `user@corp.example.com` or `corp.example.com\user`; `CORP\user` and a bare user name are refused. The password is read from `CYGNADR_ESTIMATE_PASSWORD`, or prompted for. Default: your own identity. Required on a computer that is not in the domain.
`--transport <NAME>`	`winrm` (TCP 5985) or `dcom` (TCP 135 and the dynamic RPC range). Default: `winrm`.
`--parallel <N>`	How many domain controllers to measure at the same time. Default: 8.
`--dc-timeout <SECONDS>`	How long reading one domain controller may take in total. Default: 60.
`--versions <N>`	Backups kept per domain controller—the vault's maximum versions per DC. `0` means no count limit. Default: the product default.
`--days <N>`	The vault's maximum version age, in days. `0` means no age limit. Default: the product default.
`-e, --every <MINUTES>`	How often each domain controller is backed up, used with `--days`. Default: 1440 (daily).
`--per-domain <N>`	Back up this many domain controllers in each domain instead of every one, each sized at its domain's average. A domain with fewer domain controllers backs up all it has.
`--concurrent <N>`	How many backups run at once—the maximum concurrent backups setting. Default: the product default.
`--overhead <PERCENT>`	Image overhead on top of the used space a backup copies. Default: 15.
`--growth <PERCENT>`	Allowance for growth over the repository's life, included in the recommended figure. Default: 25.
`--volumes`	List every volume of every domain controller, critical or not, with what was subtracted and why.

Examples

Measure the whole forest from a domain-joined computer, keeping 14 versions per domain controller:

```
cdr repository estimate --versions 14 --volumes
```

Size for 180 days of daily backups of three domain controllers per domain, with an explicit account (`cdr` prompts for its password):

```
cdr repository estimate --days 180 --every 1440 --per-domain 3 --server dc01.corp.example.com --user admin@corp.example.com
```

storage replace

Makes a candidate storage agent the storage machine. Every restore point on the current storage machine stops being restorable, and backups start again from zero on the new one. Without `--confirm`, the command only says what the replacement would do: the current and new machine, how many restore points end, how many uncommitted backups are given up, how many domain controllers are backed up at once, the two machines' locally set retention values with any that are `weaker` on the new one, and anything that blocks it now. For the whole procedure, see Replacing the storage machine.

With `--confirm`, an operator holding both `agents.manage` and `backup.manage` replaces the storage machine at once. An operator holding one of them creates a request instead, which an operator holding the other approves with `cdr request approve` within 24 hours. Nothing changes until then.

Requirements: The `environment.view` capability to see what it would do. To ask for it, `agents.manage` or `backup.manage`; to carry it out alone, both.

Syntax

```
cdr storage replace <ID> [--confirm <ID>] [--note <TEXT>]
```

Argument or option	Description
`<ID>`	Required. The candidate storage agent's ID, as `cdr agent list --kind storage --wide` shows it.
`--confirm <ID>`	Type the candidate's agent ID again to ask for the replacement. Without it, `cdr` only says what would happen.
`--note <TEXT>`	Why, for the operator who approves it and for the audit trail.

Exit codes: 0 when the replacement was carried out or asked for, or when nothing was confirmed; 4 when something blocks it now, or the Manager refused it because of the state of the installation; 3 when the request failed or the candidate is not enrolled.

Examples

```
cdr storage replace 7c1e0b52-93d4-4f6a-a2e8-1b9c3d4e5f60
cdr storage replace 7c1e0b52-93d4-4f6a-a2e8-1b9c3d4e5f60 --confirm 7c1e0b52-93d4-4f6a-a2e8-1b9c3d4e5f60 --note "Moving to production"
```

request list, show, approve, decline, and cancel

An action that needs two capabilities, such as replacing the storage machine, waits as a request when the operator who asked holds only one of them. These commands list requests and decide them. See Requests.

- **`request list`**—Pending requests and those decided in the last week, pending first, with the columns **ID**, **STATE**, **ACTION**, **TARGET**, **ASKED BY**, **ASKED**, **NEEDS**, and **DECIDED BY**. `--all` lists every request. Exits with code 4 while one is pending.
- **`request show <ID>`**—One request, and for a pending one what approving it would do now, which can differ from when it was asked.
- **`request approve <ID> --confirm <TARGET>`**—Approves a request, and it is carried out at once. For a storage replacement, the target is the candidate's agent ID. Without `--confirm`, the command shows what approving would do and changes nothing. You need the capability the requester lacked, and cannot approve your own request.
- **`request decline <ID>`**—Declines a request you could have approved.
- **`request cancel <ID>`**—Takes back a request: your own, or any whose action needs a capability you hold.

`approve`, `decline`, and `cancel` take `--note <TEXT>` for the audit trail. A refusal exits with code 4 when the request can no longer be decided that way, for example because it was already decided or something blocks it now, and 3 otherwise.

Requirements: The `dashboard.view` capability (every role) to list and show requests. Approving, declining, and cancelling depend on the request, as described above.

Examples

```
cdr request list
cdr request show a1b2c3d4e5f6
cdr request approve a1b2c3d4e5f6 --confirm 7c1e0b52-93d4-4f6a-a2e8-1b9c3d4e5f60
cdr request decline a1b2c3d4e5f6 --note "Not until the cutover weekend"
```

os-media list

Lists the operating systems your domain controllers run—build, edition, installation type, architecture (**ARCH**), and language—with each one's media coverage: whether the OS media library holds installation media from which a recovery image for that operating system can be built. Architecture is compared wherever both the machine's backup and the media record it. With `--media`, it lists the media instead; with `--images`, the recovery images.

Below the table, the command says:

- How many domain controllers have no manifest recording their build, edition, or install type, and why: no DC agent (`cdr agent deploy` installs one), an agent and no completed backup yet (`cdr backup start` gives each a row), or backed up but the agent could not read the build, edition, or install type. Excluded domain controllers are not counted, and a hint says how many.
- For each storage agent's recovery image library, one line: how many images are retained, how many are to be retired, and how many have been retired so far, each with its size. If the Manager is not deciding retirement for a library, a hint says why.

Requirements: The `backup.view` capability (every role).

Syntax

```
cdr os-media list [options]
```

Options

Option	Description
`-c, --coverage <CODE>`	Only operating systems with this verdict: `covered`, `uncovered`, `language-mismatch`, `unconfirmed`, `unscanned`, or `no-library`.
`--uncovered`	Shorthand for `--coverage uncovered`: the machines that could not be rebuilt from the media on hand. Cannot be combined with `--coverage`.

Option	Description
`--media`	List the media instead: the configured folders, their scans, and the images inside each ISO file.
`--images`	List the recovery images instead: what each storage agent's library holds, which it keeps and which it retires, and what it has retired. For each library, a table with the columns **FILE**, **BUILD**, **EDITION**, **ARCH**, **VERDICT**, **SIZE**, **BUILT**, **FATE** (`keep:` or `retire:` with the reason), and **SHA-256** (the image's hash, which `cdr restore plan --verify --image` takes), the retention line, and how many images it has retired, with a pointer to `cdr audit list --action recovery-image.retired`, where each retirement is recorded. Cannot be combined with `--media`.

Examples

```
cdr os-media list --uncovered
cdr os-media list --media
cdr os-media list --images
```

For how to supply media, see Setting Up OS Media and Recovery Images.

Restore Commands

The `restore` commands restore one domain controller, by DC state recovery or by full system recovery, or test-restore one of its backups on an Isolated Recovery Host. They follow the same steps as the console: plan the restore and read its preflight checks, approve it, run it, and follow it to the end. Planning, approving, and running are separate commands because approving and executing are separate capabilities: the person who approves a restore need not be the one who runs it.

A typical sequence:

```
cdr backup list --machine dc01.corp.example.com
cdr backup show 412
cdr restore plan dc01.corp.example.com --version 10/05/2026-02:30 --authoritative-subtree
"OU=Sales,DC=corp,DC=example,DC=com"
cdr restore approve a1b2c3d4e5f6 --accept authoritative.overwrites
cdr restore run a1b2c3d4e5f6 --wait
```

Each run has an ID, which `cdr restore plan` prints and `cdr restore list` shows. For the lifecycle of a run, see Recovery Lifecycle; for the procedures, see Restoring a DC with DC State Recovery, Full System Recovery in Place, Full System Recovery onto a

Candidate, Full System Recovery onto a New VM, and Running a Test Restore. To recover a whole domain or forest, use the `recovery` commands in Recovery Plan Commands.

For how to run `cdr` and the global options, see Command-Line Reference.

restore list

Lists the restores this installation has planned, test restores included, a page at a time: runs in progress first, then newest first. **WHEN** is when the run started, or when it was planned if it has not started, and **DURATION** is how long it took, with `+` after a figure that is still growing. When there are more runs than the page shows, the command says how many and how to page. If any run is waiting on an operator, the command warns you and exits with code 4, so a scripted check cannot pass over a domain controller waiting on a person.

A test restore is marked after the domain controller's name: `(test, scheduled)` for one the schedule started, `(test, approved by` *operator*`)`, or `(test, manual)` when no approver is recorded.

Requirements: The `plan.view` capability (every role).

Syntax

```
cdr restore list [--machine <FQDN>] [--no-tests] [--limit <N>] [--skip <N>]
```

Options

Option	Description
`--machine <FQDN>`	Only runs for this domain controller. This option has no short form.
`--no-tests`	Leave test restores out. The command says how many it did not list. Runs waiting on an operator are still counted, test restores included.
`-n`, `--limit <N>`	How many runs to show, from 1 to 200. Default: 50.
`--skip <N>`	Skip this many first. With the default limit, `--skip 50` is the second page.

Example

```
cdr restore list --machine dc01.corp.example.com
```

restore show

Shows one run in full: its preflight gates, its steps, and what the machine reported, including the network adapters the machine described (which `--recovery-adapter` and `--restored-adapter` refer to). With `--wait`, it re-reads the run until it reaches a final state.

For a full system recovery, the `onto` fact says where the restore goes: `in place`, `onto a candidate machine`, or `onto a new VM booted from per-run recovery media`. A run onto a new VM also shows:

- **`recovery media`**—The ISO's file name, its size, and when it was built.
- **`media sha256`**—The ISO's SHA-256 hash.
- **`media at`**—Where the ISO is exposed, followed by `(taken back)` once the media has expired.
- **`vault handed over`**—When the vault credential was handed to the recovery environment, or `not yet: no VM has enrolled`.

While the run waits for a VM, a hint gives the `cdr dc rebuild … --restore-run` command that builds and boots it.

For a DC state recovery, an `authoritative subtree` or `authoritative object` line names each entry the run marks authoritative. A DC state recovery that a domain or forest plan started shows `authoritative nothing` instead: it is non-authoritative. For a DC state recovery that resets `krbtgt`, a `krbtgt` line says whether the Manager holds the credentials for the reset (`held in the manager's memory` or `not held`), and, when the reset can be started again, a hint gives the `cdr restore reset-krbtgt` command.

For a test restore, the output adds a verification block: a line with the host and the verdict (`restore verified`, `restore failed at` a step, or `no verdict yet` with the reason), a line with the VM's state and heartbeat and what the restored domain controller reports, a table of the checks with the columns CHECK, RESULT, and DETAIL, and whether the VM, its disks, the ISO copy, and its private switch have been deleted. Once the VM is deleted, its line starts `VM deleted (last seen` with its last state, heartbeat, and time. While the run is still open and the host cannot read the VM or has stopped reporting on it, a line starting `as of` gives the time of the last read and the Manager's account of why, followed by `the restored DC may be further along than this`. If the restored copy did not apply every post-recovery directive, a warning points you to the verification verdict and checks above it, which say what failed. Unlike a real restore, there is nothing to set by hand afterward, because the copy is deleted with the run. See Test Restores.

Exit codes: 0 when the run is in progress, completed, or cancelled; 4 when it needs an operator; 7 when it failed; 8 when `--wait` loses track of the run.

Requirements: The `plan.view` capability (every role).

Syntax

```
cdr restore show <RUN> [options]
```

Arguments and options

Argument or option	Description
`<RUN>`	**Required.** The run ID.
`--wait`	Follow the run until it reaches a final state.
`--poll <SECONDS>`	How often to re-read while waiting, from 5 to 600 seconds. Default: 15.

Example

```
cdr restore show a1b2c3d4e5f6 --wait --poll 30
```

restore plan

Plans a restore of one domain controller from a chosen backup, and runs its preflight checks. It records the run and writes nothing to any machine. The output lists the gates: hard gates block approval, confirm gates block it until you accept each one by its code with `cdr restore approve --accept`, and advisory gates are for you to read. A plan with hard or confirm gates is still created, so that you can read what is in the way. The command then exits with code 4 and, for confirm gates, prints the approve command that accepts them. Some checks can only run against the live environment and are made when the restore executes.

You must name the backup with `--version`. `cdr` never picks one for you, because choosing the backup is choosing how far back to roll the directory. `cdr backup show` prints a backup's version identifier.

- **DC state recovery** (`--mode rollback`, the default)–Windows still starts on the domain controller. Its agent restores DC state from Directory Services Restore Mode, and marks the subtrees and objects you name with `--authoritative-subtree` and `--authoritative-object` authoritative, so that they go back to the backup on every domain controller of the domain. Name at least one. Works from either backup type. See Restoring a DC with DC State Recovery for what is accepted.
- **Full system recovery** (`--mode server`)–Restores the whole server from a full system backup through the recovery image. In place, the domain controller's agent stages the recovery image and restarts into it. With `--onto`, the restore goes onto a candidate machine instead. With `--onto-new-vm`, it goes onto an empty VM booted from a

recovery ISO that the storage agent builds for the run when you execute it; `cdr dc rebuild --restore-run` then builds the VM and boots it. In both cases, the original domain controller must be powered off before you run it.

- **Test restore** (`--verify`, or `cdr restore verify`)—Restores the backup onto a new VM on an Isolated Recovery Host, moves it to a private switch before it starts, checks it as the first domain controller of a recovered domain, and deletes it. Nothing in production changes, so the original keeps running. Implies `--mode server`.

For a full system recovery you can also choose the network settings of the recovery environment and of the restored system at its first start. Each takes `unchanged`, `original`, or `override`; giving an address alone implies `override`. The recovery environment defaults to `unchanged`; for the restored system, leaving the option out lets the Manager choose. The Manager checks the settings—for example, that the gateway is on the address's subnet—when you plan.

Requirements: The `plan.approve` capability (Restore Operator role). For what a restore needs, see Recovery Requirements; for each gate, see Preflight Check Reference.

Syntax

```
cdr restore plan <FQDN> --version <ID> [options]
```

Arguments

Argument	Description
`<FQDN>`	**Required.** The domain controller to restore.

Options

Option	Description
`--version <ID>`	**Required.** The version identifier of the backup to restore, as `cdr backup show` prints it.
`--mode <MODE>`	`rollback` (or `A`) for a DC state recovery, or `server` (or `B`) for a full system recovery. Default: `rollback`.
`--authoritative-subtree <DN>`, `--authoritative-object <DN>`	DC state recovery only. A subtree to roll back—the object and everything beneath it—or one object, marked authoritative after the restore so that it replicates out instead of being overwritten from the other domain controllers. Repeat either option for each. A DC state recovery needs at least one; a full system recovery refuses both.

Option	Description
`--reset-krbtgt`	DC state recovery only, with a subtree that holds `krbtgt`, such as `CN=Users,DC=…`. Reset `krbtgt` twice once the rollback validates, the second time after the first has reached every writable domain controller. Approving then needs `--admin`. The Manager refuses it for a DC state recovery whose subtrees do not hold `krbtgt`.
`--onto <AGENT_ID>`	Full system recovery only. Recover onto a candidate machine, by its agent ID, instead of in place. `cdr recovery candidates` lists them. Requires `--mode server`.
`--onto-new-vm`	Full system recovery only. Recover onto an empty VM booted from a recovery ISO built for this run, instead of in place or onto a candidate. Nothing is installed on the VM first: when you execute the run, the storage agent builds the ISO, and `cdr dc rebuild --restore-run` builds the VM and attaches it. The original domain controller must be powered off before you execute. Requires `--mode server`, and cannot be combined with `--onto`.
`--verify`	Plan a test restore instead of a recovery: restore this version onto a new VM on an Isolated Recovery Host, isolate it, check it, and delete it. Nothing in production changes. Implies `--mode server`, and cannot be combined with `--onto` or `--onto-new-vm`.
`--host <AGENT_ID>`	With `--verify`: the Isolated Recovery Host to use, by its agent ID, as `cdr recovery-host list` shows it. Default: the host that can run it now.
`--image <SHA256>`	With `--verify`: the recovery image to boot, by its SHA-256 hash (`cdr os-media list --images`), instead of the one the Manager would choose. Use it to prove an untested image when a proven one would otherwise be used.
`--reverify`	Re-hash the whole backup before restoring it, instead of trusting the storage agent's own recent verification. Adds the time a full read takes.
`--recovery-network <CHOICE>`	Full system recovery only. The recovery environment's address: `unchanged` (DHCP), `original` (this machine's own static address, as it last described itself), or `override` (the address given with `--recovery-address`). Default:

Option	Description
	`unchanged`.
`--recovery-address <CIDR>`	The address the recovery environment takes instead of DHCP, with its prefix length, for example `10.99.0.10/24`.
`--recovery-gateway <IP>`	The recovery environment's gateway. Omit it on a network segment with no route out.
`--recovery-dns <IPS>`	The recovery environment's DNS servers, comma-separated, in order, up to four. The recovery environment reaches the vault at the storage agent's advertised address, its IP address by default; it needs a DNS server only if the storage machine is advertised by name.
`--recovery-adapter <MAC>`	The adapter that takes the address, by MAC address, when the machine has more than one.
`--restored-network <CHOICE>`	Full system recovery only. The restored system's address at its first start: `unchanged` (whatever the backup had), `original` (the backup's own static address, put on the target's adapter–what new hardware needs), or `override` (the address given with `--restored-address`, for example on a new subnet). Applied by the restored domain controller's own agent before it reconnects to the Manager. Omitted, the Manager decides: `original` when the backup recorded one static address, its agent can apply it, and the target's adapter is known (onto a candidate, or in place when the machine's description names the adapter); `unchanged` otherwise.
`--restored-address <CIDR>`	The address the restored system takes at its first start, with its prefix length.
`--restored-gateway <IP>`	The restored system's gateway. Omit it on a network segment with no route out.
`--restored-dns <IPS>`	The restored system's DNS servers, comma-separated, in order–usually itself, or another domain controller on the recovery network.
`--restored-adapter <MAC>`	The target adapter that takes the restored system's address, by MAC address, when it has more than one.

Examples

Plan a DC state recovery of one organizational unit and one user:

```
cdr restore plan dc01.corp.example.com --version 10/05/2026-02:30 --authoritative-subtree
"OU=Sales,DC=corp,DC=example,DC=com" --authoritative-object "CN=Bob,O-
OU=IT,DC=corp,DC=example,DC=com"
```

Undo a mass password reset by rolling back `CN=Users`, with the `krbtgt` reset afterwards:

```
cdr restore plan dc01.corp.example.com --version 10/05/2026-02:30 --authoritative-subtree
"CN=Users,DC=corp,DC=example,DC=com" --reset-krbtgt
```

Plan a full system recovery onto a candidate machine on an isolated network, giving the restored system its original address:

```
cdr recovery candidates
cdr restore plan dc01.corp.example.com --version 10/05/2026-02:30 --mode server --onto 6d2f9a1c-
4b8e-4f7a-9c3d-2e1b0a9f8c7d --recovery-address 10.99.0.10/24 --recovery-dns 10.99.0.5 --
restored-network original
```

Recover onto a new VM: plan, approve, and run the restore, which builds the recovery ISO; then, on the Hyper-V host, build the VM and boot it, and follow the run:

```
cdr restore plan dc01.corp.example.com --version 10/05/2026-02:30 --mode server --onto-new-vm
cdr restore approve a1b2c3d4e5f6
cdr restore run a1b2c3d4e5f6
cdr dc rebuild dc01.corp.example.com --restore-run a1b2c3d4e5f6 --switch "Recovery" --start
cdr restore show a1b2c3d4e5f6 --wait
```

For the whole procedure, see Full System Recovery onto a New VM.

restore verify

Plans a test restore of one version on an Isolated Recovery Host: the backup is restored onto a new VM, isolated on a private switch, checked as the first domain controller of a recovered domain, and deleted. Nothing in production changes. `cdr restore verify` is the same as `cdr restore plan --verify`, and takes all of its options, including `--host` and `--image`. Approve and run the plan as any other, and follow it with `cdr restore show --wait`.

If no host can run it, the Manager refuses the plan and names the host and what stops it; `cdr recovery-host list` shows each host and its state.

Requirements: The `plan.approve` capability (Restore Operator role), and an Isolated Recovery Host that is ready. See Setting Up an Isolated Recovery Host.

Syntax

```
cdr restore verify <FQDN> --version <ID> [options]
```

Example

```
cdr restore verify dc01.corp.example.com --version 10/05/2026-02:30
cdr restore approve a1b2c3d4e5f6
cdr restore run a1b2c3d4e5f6 --wait
```

For the procedure and how to read the result, see Running a Test Restore.

restore survey

Asks the machine a planned restore will run on to describe itself again—its disks, firmware, adapters, and the other facts the preflight checks compare with the backup—and re-evaluates the gates. Use it after you change something on the target, such as adding a disk, so that the plan reflects the machine as it is now. The machine has up to 90 seconds to answer.

Requirements: The `plan.approve` capability (Restore Operator role). The target's agent must be connected.

Syntax

```
cdr restore survey <RUN>
```

Argument	Description
`<RUN>`	**Required.** The run ID `cdr restore plan` printed.

Example

```
cdr restore survey a1b2c3d4e5f6
```

restore approve

Approves a planned restore whose hard gates are clear. The approval is recorded under your name in the audit trail, with each confirm gate you accept. A confirm gate is something that failed from where the Manager looked but may not fail from where the recovery environment runs, such as a storage machine name the target cannot resolve now. Read its remediation in `cdr restore show` first, and accept it only if you know the restore will get past it safely; otherwise cancel or discard the run. There is no option to accept every gate at once. A DC state recovery raises its own confirm gates: `authoritative.overwrites`, which lists what is rolled back on every domain controller, and, when a subtree holds `krbtgt`, `authoritative.krbtgt`. If a forest or domain plan is waiting on this run and you also hold `plan.execute`, the plan carries on at once.

For a DC state recovery planned with `--reset-krbtgt`, pass `--admin` with a domain administrator. `cdr` prompts for its password, or reads it from the `CYGNADR_RESTORE_ADMIN_PW` environment variable; with no terminal to prompt on and no variable set, the command stops. The password is never an argument. The Manager holds the credentials only in memory, until the reset is done or fails, for at most 8 hours. It forgets them if it restarts. You can leave `--admin` out when the Manager already holds credentials for the run.

Requirements: The `plan.approve` capability (Restore Operator role).

Syntax

```
cdr restore approve <RUN> [--accept <GATE>]... [--admin <USER>]
```

Argument	Description
`<RUN>`	**Required.** The run ID `cdr restore plan` printed.
`--accept <GATE>`	A confirm gate to accept, by its code. Repeat for each. Only gates the preflight raised count.
`--admin <USER>`	For a rollback that resets `krbtgt`: a domain administrator, as `DOMAIN\user` or `user@domain`. Its password is prompted for, or read from `CYGNADR_RESTORE_ADMIN_PW`.

Examples

```
cdr restore approve a1b2c3d4e5f6
cdr restore approve a1b2c3d4e5f6 --accept target.vault-unresolved
cdr restore approve a1b2c3d4e5f6 --accept authoritative.overwrites --accept authoritative.krbtgt --admin CORP\admin
```

restore run

Executes an approved restore. This replaces the domain controller's directory with the copy from the chosen backup, and anything written to it since that backup and not replicated elsewhere is lost. For a recovery onto a new VM, running it builds the recovery ISO; nothing is restored until a VM boots it.

Before it starts, `cdr` describes what will happen and asks you to type the domain controller's short name–the first part of its name, for example `DC01` for `dc01.corp.example.com`. A test restore changes nothing in production, so for one the command only asks you to confirm. In a script, pass `--yes`; without it, the command is refused when there is no terminal, rather than assume consent.

With `--wait`, the command follows the run to its end and exits as `cdr restore show --wait` does.

Without an active license, the Manager refuses to execute the run, and the command exits with code 3. See Licensing.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr restore run <RUN> [options]
```

Arguments and options

Argument or option	Description
`<RUN>`	**Required.** The approved run ID to execute.
`--yes`	Skip the confirmation. Required when standard input is not a terminal.
`--wait`	Follow the run until it reaches a final state.
`--poll <SECONDS>`	How often to re-read while waiting, from 5 to 600 seconds. Default: 15.

Examples

```
cdr restore run a1b2c3d4e5f6 --wait
cdr restore run a1b2c3d4e5f6 --yes --wait
```

restore reset-krbtgt

Starts the `krbtgt` reset of a completed DC state recovery again: after the reset failed, after the second reset was not made because the first had not reached every writable domain controller in time, or after the Manager forgot the credentials given at approval because 8 hours passed or it restarted. It does only what is left: if the first reset was applied, only the second is made, once the first has reached every writable domain controller. Follow it with `cdr restore show`, where it is step `R-04`.

`cdr` prompts for the domain administrator's password, or reads it from the `CYGNADR_RESTORE_ADMIN_PW` environment variable. The Manager refuses a run that was not planned to reset `krbtgt`, one that has not validated, one whose reset is already running, and one where `krbtgt` has already been reset twice.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr restore reset-krbtgt <RUN> --admin <USER>
```

Argument or option	Description
`<RUN>`	**Required.** The rollback whose `krbtgt` reset to start again. `cdr restore list` prints the IDs.
`--admin <USER>`	**Required.** A domain administrator, as `DOMAIN\user` or `user@domain`.

Example

```
cdr restore reset-krbtgt a1b2c3d4e5f6 --admin CORP\admin
```

restore cancel

Asks a running restore to stop. This is possible only until the restore starts writing to the machine, and, for a DC state recovery, while the machine's agent can still be reached: once the machine has restarted into DSRM and its agent is away, the restore may be running there. After that, the Manager refuses: the restore runs to its end, and the command warns you and exits with code 4. Follow it with `cdr restore show --wait`. A test restore can be cancelled at any point, even after it has started writing: it changes nothing in production, its VM is torn down, and the version stays unverified. For a recovery onto a new VM whose recovery environment is connected, the Manager refuses with `cancel-refused-recovery-environment` and the command exits with code 4: power the VM off from its hypervisor before wbadmin starts writing, and cancel again once the run has gone quiet.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr restore cancel <RUN>
```

Argument	Description
`<RUN>`	**Required.** The run ID to stop.

Example

```
cdr restore cancel a1b2c3d4e5f6
```

restore discard

Discards a planned or approved restore that was never executed—for example, one whose preflight failed. The Manager refuses to discard a run that has already been executed (use `cdr restore cancel`), or a run that a domain or forest recovery plan owns (abort the plan with `cdr recovery abort`).

Requirements: The `plan.approve` capability (Restore Operator role).

Syntax

```
cdr restore discard <RUN>
```

Argument	Description
`<RUN>`	**Required.** The run ID of the planned restore to discard.

Example

```
cdr restore discard a1b2c3d4e5f6
```

restore abandon

Closes a run that cannot be cancelled because its machine is gone or will be rebuilt—for example, a candidate virtual machine that was deleted. The reason is required and is recorded in the audit trail.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr restore abandon <RUN> --reason <TEXT>
```

Argument or option	Description
`<RUN>`	**Required.** The run ID to close.
`--reason <TEXT>`	**Required.** Why the run is being given up. Recorded in the audit trail.

Example

```
cdr restore abandon a1b2c3d4e5f6 --reason "Candidate VM deleted; rebuilding on new hardware"
```

restore validate

Asks a restored machine to check that its domain controller works—AD DS running, SYSVOL ready, replication converging, and dcdiag clean—and records the result on the run. Use it after a DC state recovery whose first check came too early for replication, or after fixing what made a check fail. For a DC state recovery, it also imports any back-link file that failed to import and reads the marks again; a DC state recovery counts as restored only once its marks are read back. Otherwise it changes nothing on the machine. The machine waits up to 15 minutes for replication to converge, then reports; `cdr restore show` shows the result.

The run must have finished, or failed on its checks, and the machine's agent must be connected and able to make the checks. Otherwise the command says why it cannot.

Requirements: The `plan.approve` capability (Restore Operator role).

Syntax

```
cdr restore validate <RUN>
```

Argument	Description
`<RUN>`	**Required.** The run whose restored domain controller to check. `cdr restore list` prints the IDs.

Example

```
cdr restore validate a1b2c3d4e5f6
```

Recovery Plan Commands

The `recovery` commands build and drive a recovery plan for a whole forest or, with `--domain`, for one domain in a forest that stays live. You decide what happens to each domain controller: restore it from a backup, rebuild it (clean up its metadata and promote a new one), or exclude it (clean up its metadata and not bring it back). The Manager builds the ordered steps from the topology. You then approve the plan, supply each domain's administrator credential and the new DSRM password, execute the plan, and answer the decisions a running plan waits on. You can supply credentials at any time before the steps that need them run; a step that needs one it does not have waits for it. A plan built with `--unattended` does not stop for those decisions, except at trust resets carried out by hand.

A typical forest recovery:

```
cdr recovery plan corp.example.com --restore dc01.corp.example.com=10/05/2026-02:30 --rebuild
dc02.corp.example.com --ntp time.example.com
cdr recovery approve 7c41e0a2b9d3
cdr recovery secrets 7c41e0a2b9d3 --domain corp.example.com --admin CORP\Administrator --
dsrm
cdr recovery execute 7c41e0a2b9d3
cdr recovery show 7c41e0a2b9d3 --wait
cdr recovery report 7c41e0a2b9d3
```

A plan moves through these states: `draft`, `approved`, `running`, then `completed` or `aborted`. A running plan that stops on a failed or rejected step is `halted` until the step is retried with `cdr recovery retry` or the plan is aborted. Abort is final: a recovery is planned again, not resumed.

Steps are identified by a step ID and a code, both shown by `cdr recovery show`. For each step waiting on you, `cdr recovery show` prints the exact command to run next, including the step ID and, for a point of no return, the code to type.

For the procedures, see Recovering a Domain and Recovering a Forest. To restore a single domain controller, use the `restore` commands in Restore Commands. For how to run `cdr` and the global options, see Command-Line Reference.

recovery list

Lists the domain and forest recovery plans, a page at a time: plans in progress first, then newest first. For each it shows its forest, scope, status, and steps completed, when it was created (**CREATED**) and ended (**ENDED**), and how long it ran from execution to its end, including any time halted (**DURATION**, with `+` after a figure that is still growing). When there are more plans than the page shows, the command says how many and how to page.

Requirements: The `plan.view` capability (every role).

Syntax

```
cdr recovery list [--limit <N>] [--skip <N>]
```

`-n` or `--limit` sets how many plans to show, from 1 to 200 (default 50). `--skip` skips that many first, so `--skip 50` with the default limit is the second page.

Example

```
cdr recovery list
```

recovery candidates

Lists the candidate machines: bare machines with the DC agent enrolled that are not domain controllers, and can receive a full system recovery. The **AGENT ID** column is what `--onto` takes, here and in `cdr restore plan`.

Requirements: The `plan.view` capability (every role).

Syntax

```
cdr recovery candidates [options]
```

Takes the global options only.

Example

```
cdr recovery candidates
```

recovery plan

Builds a forest recovery plan, or with `--domain` a domain recovery plan, from the topology and your choice for each domain controller, and shows its gates and steps. At least one domain controller must be restored.

If the plan has hard gates, clear what they report and plan again. Otherwise, approve the plan, then supply each domain's credential with `cdr recovery secrets`.

Each domain must restore exactly one domain controller; restoring two in one domain is a hard gate, with or without `--breach`. Rebuild the others. Some options apply to only one kind of plan: `--ntp` and `--gc` to a forest recovery, and `--gc-strategy` to a domain recovery. Giving one to the other kind is refused rather than ignored.

Requirements: The `plan.approve` capability (Restore Operator role). For what a recovery needs, see Recovery Requirements.

Syntax

```
cdr recovery plan <FOREST> --restore <SPEC> [options]
```

Arguments

Argument	Description
`<FOREST>`	**Required.** The forest to recover, by the DNS name of its root domain.

Options

Option	Description
`--restore <SPEC>`	**Required at least once.** A domain controller to restore, as `<fqdn>=<version>[:mode]`, where `mode` is `rollback` (DC state recovery) or `server` (full system recovery). Default mode: `server`. A plan's DC state recovery is non-authoritative and names nothing to roll back; to roll back subtrees or objects across a running domain, use `cdr restore plan` with `--authoritative-subtree` or `--authoritative-object`. Repeat the option for each domain controller.
`--onto <SPEC>`	Bind a candidate machine to a domain controller, as `<fqdn>=<agent-id>`. A restore recovers onto it instead of in place; a rebuild renames and promotes it instead of waiting for the domain controller's own machine. Each machine can be bound to only one domain controller. `cdr recovery candidates` lists the agent IDs. Repeatable.
`--rebuild <FQDN>`	A domain controller to clean out of the directory and promote again, with its roles and features reinstalled from its last backup's record. Every domain controller of a domain but the one restored. Repeatable.
`--exclude <FQDN>`	A domain controller to neither restore nor rebuild: its metadata is cleaned out of the directory along with the domain's other domain controllers, and it is not brought back. Keep the machine off the network. Repeatable.
`--breach`	Plan a recovery from a security breach: all trusts are reset, and the breach-response resets are added. Without it, a forest recovery resets no trust, and each side of a trust keeps the trust password its own restore point holds. A domain recovery resets the domain's trusts either way.
`--reverify`	Re-hash every backup in full before it is restored, instead of trusting the storage agent's own recent verification. Adds the time a full read takes, for each domain controller.
`--ntp <SERVER>`	Forest recovery only. An external time source for the forest root domain's PDC emulator. Repeatable.
`--rid-increase <N>`	How many RIDs to invalidate, to protect the RID pool from reuse. Default: 100000.

Option	Description
`--gc <FQDN>`	Forest recovery only. The restored root domain controller to make a global catalog again.
`--domain <FQDN>`	Plan a domain recovery of this one domain instead of a forest recovery. The rest of the forest stays live; only this domain's domain controllers are restored, rebuilt, or excluded.
`--gc-strategy <STRATEGY>`	Domain recovery only. How the global catalogs in the other domains are reconciled with the recovered domain: `lingering-objects` (they stay online) or `rehost` (the partition is removed from them and added back, site by site). Default: `lingering-objects`.
`--unattended`	Run without stopping for anyone: every point of no return, attestation, backup checkpoint, and restore preflight confirmation is accepted in the name of whoever executes the plan, and audited. Confirmed at execution with `cdr recovery execute --confirm <NAME> --isolated`. A failure still halts, a missing credential still waits, and a trust reset carried out by hand–an external trust, or a partner with no DC agent–still stops for a person. Trust resets the agents carry out are accepted.
`--notify <ADDRESS>`	Email the plan's halts, aborts, and waits for a person, and its recovery report on completion, to this address. Repeatable, or comma-separated. Without it nothing is emailed. Needs `plan.execute`.
`--dsrm`	Set the new DSRM administrator password the plan puts on every domain controller it recovers, now. It is prompted for twice, or read from the `CYGNADR_RECOVERY_DSRM_PW` environment variable, and held by the Manager in memory only. Without it, the DSRM step waits for one (`cdr recovery secrets <PLAN> --dsrm`).

Examples

A forest recovery that restores one domain controller from a full system backup and rebuilds another:

```
cdr recovery plan corp.example.com --restore dc01.corp.example.com=10/05/2026-02:30 --rebuild dc02.corp.example.com --ntp time.example.com
```

A domain recovery of a child domain, restoring onto a candidate machine:

```
cdr recovery plan corp.example.com --domain emea.corp.example.com --restore dc1.e-mea.corp.example.com=10/05/2026-01:15:server --onto dc1.emea.corp.example.com=6d2f9a1c-4b8e-4f7a-9c3d-2e1b0a9f8c7d --rebuild dc2.emea.corp.example.com
```

An unattended forest recovery that emails the incident team and holds the new DSRM password from the start:

```
cdr recovery plan corp.example.com --restore dc01.corp.example.com=10/05/2026-02:30 --rebuild dc02.corp.example.com --ntp time.example.com --unattended --notify ad-team@corp.example.com --dsrm
```

recovery show

Shows a plan in full: its gates, what each domain controller is planned to do, at which step, and from which backup, what each rebuild reinstalls and does not carry, the time source, and every step with its state and the reason for any check that failed. From the reconnect until the last rebuilt domain controller is back, it says which domain controllers each domain is serving on. For each step waiting on you, it prints the command that unblocks it. For a plan that is a draft, approved, running, or halted, it also lists the credentials that steps still to run need: `domain credentials held (in memory): <domains>`, and, for each domain whose credential the Manager does not hold, `domain credential not held: <domain> — cdr recovery secrets <plan> --domain <domain> --admin <DOMAIN\user>`. With `--wait`, it re-reads the plan every 10 seconds until the plan is completed or aborted.

Requirements: The `plan.view` capability (every role).

Syntax

```
cdr recovery show <PLAN> [--wait]
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID `cdr recovery list` prints.
`--wait`	Follow the plan, re-reading it until it reaches a final state.

Example

```
cdr recovery show 7c41e0a2b9d3 --wait
```

recovery secrets

Supplies a domain's administrator credential, the new DSRM password, or both, to a plan. The Manager holds them in memory only, and drops them when the plan ends or the Manager restarts. Run the command once for each domain in the plan, at any time before the steps that need the credential run; a step that needs one the Manager does not hold waits for it. For the forest root domain, give an account that is an Enterprise Admin and a Schema Admin; for other domains, a Domain Admin.

The domain password is read from the `CYGNADR_RECOVERY_ADMIN_PW` environment variable, or prompted for. You can also give the password that the domain's built-in Administrator account had at the backup point, with `--rid500`: it is read from the `CYGNADR_RECOVERY_RID500_PW` environment variable, or prompted for. The variable is read without `--rid500` too. The new DSRM password is read from `CYGNADR_RECOVERY_DSRM_PW`, or prompted for twice. Keep it per your policy: it is never shown again.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery secrets <PLAN> [--domain <FQDN> --admin <USER> [--rid500]] [--dsrm]
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`--domain <FQDN>`	The domain the credential is for. Give it with `--admin`.
`--admin <USER>`	A domain administrator, for example `CORP\Administrator`. For the forest root domain, an Enterprise Admin and Schema Admin. Give it with `--domain`.
`--rid500`	Also supply the password the domain's built-in Administrator account (RID 500) had at the backup point. Give it with `--domain` and `--admin`.
`--dsrm`	Supply the new DSRM administrator password the plan sets on every domain controller it recovers. With or without `--domain` and `--admin`; give at least one of the two.

Examples

```
cdr recovery secrets 7c41e0a2b9d3 --domain corp.example.com --admin CORP\Administrator
cdr recovery secrets 7c41e0a2b9d3 --domain corp.example.com --admin CORP\Administrator --rid500
cdr recovery secrets 7c41e0a2b9d3 --dsrm
```

recovery approve

Approves a plan whose hard gates are clear. The approval is recorded under your name.

Requirements: The `plan.approve` capability (Restore Operator role).

Syntax

```
cdr recovery approve <PLAN>
```

Argument	Description
`<PLAN>`	**Required.** The plan ID.

Example

```
cdr recovery approve 7c41e0a2b9d3
```

recovery execute

Starts an approved plan. Follow it with `cdr recovery show --wait`. An unattended plan also needs `--confirm` with its name and `--isolated`. With them, you accept in your own name every pause the plan would otherwise stop for, and attest now that the restore targets are isolated, because nobody will be asked when the plan reaches that step. Make sure every original domain controller is already off and the network is ready for the reconnect; see Recovering a Forest.

Requirements: The `plan.execute` capability (Restore Operator role), and an active license: without one the Manager refuses to execute the plan, and the command exits with code 3. See Licensing.

Syntax

```
cdr recovery execute <PLAN> [--confirm <NAME> --isolated]
```

Argument	Description
`<PLAN>`	**Required.** The plan ID.
`--confirm <NAME>`	An unattended plan only. The forest's name (for a domain plan, the domain's), typed to confirm that every pause for a person is accepted in your name.
`--isolated`	An unattended plan only. Attest now that every restore target is isolated. See Isolating Recovery Targets.

Examples

```
cdr recovery execute 7c41e0a2b9d3
cdr recovery execute 7c41e0a2b9d3 --confirm corp.example.com --isolated
```

recovery ack

Acknowledges a step that is a point of no return, so that it can be issued. A plan stops before each such step—for example, seizing the operations master roles or cleaning up metadata—and waits for you. To acknowledge it, type the step's own code with `--confirm`. `cdr recovery show` prints the step ID and code, and the full command, for each step waiting on acknowledgment. Every acknowledgment is audited.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery ack <PLAN> <STEP> --confirm <CODE>
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`<STEP>`	**Required.** The step ID, as `cdr recovery show` prints it in the **STEP** column.
`--confirm <CODE>`	**Required.** The step's code, as `cdr recovery show` prints it in the **CODE** column, typed to acknowledge the point of no return.

Example

```
cdr recovery show 7c41e0a2b9d3
cdr recovery ack 7c41e0a2b9d3 <STEP> --confirm <CODE>
```

recovery attest

Attests, or rejects, a step that an operator performs or checks outside Cygna Forest Recovery—for example, confirming that the recovered domain controllers are isolated from the network, or the check that the backup predates the damage. Attesting marks the step done. Rejecting it fails the step and halts the plan, and needs a note saying why. A restore step waiting on its restore's preflight cannot be attested: decide it on the restore run instead, with `cdr restore approve --accept`, or cancel the run.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery attest <PLAN> <STEP> [--reject --note <TEXT>]
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`<STEP>`	**Required.** The step ID.
`--reject`	Reject the step instead of attesting it. This fails the step and halts the plan.
`--note <TEXT>`	A note recorded with the attestation. Required with `--reject`.

Examples

```
cdr recovery attest 7c41e0a2b9d3 <STEP>
cdr recovery attest 7c41e0a2b9d3 <STEP> --reject --note "dc01 is still connected to the production network"
```

recovery retry

Retries the step that halted a plan, once what failed it is fixed—for example, an agent updated or a machine brought back. The step runs again from its checkpoint, without repeating what it already did, and keeps its point-of-no-return acknowledgment. The plan then carries on. `cdr recovery show` prints the step ID of the failed step.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery retry <PLAN> <STEP>
```

Argument	Description
`<PLAN>`	**Required.** The plan ID.
`<STEP>`	**Required.** The step ID of the failed step.

Example

```
cdr recovery retry 7c41e0a2b9d3 FR-09@corp.example.com
```

recovery override-wait

Shortens a step's wait. By default there is no wait between the first and second reset of the krbtgt account: both run while the directory is isolated, and going live waits for the second. Only a plan created with a gap between the resets waits, and only then is there anything to override. Overriding the wait makes the step run now, and Kerberos tickets issued in the window are invalidated. The reason is required, because it is the audit record of the decision; the recovery report lists it.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery override-wait <PLAN> <STEP> --reason <TEXT>
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`<STEP>`	**Required.** The step ID of the waiting step.
`--reason <TEXT>`	**Required.** Why the wait is being shortened. Recorded in the audit trail.

Example

```
cdr recovery override-wait 7c41e0a2b9d3 <STEP> --reason "Business outage; accepted ticket invalidation per incident 4471"
```

recovery abort

Abandons a plan. This is final: the plan cannot be resumed, and the credentials it held are dropped. To continue the recovery, plan it again. Before it aborts, the command says what aborting now leaves—which points of no return are behind the plan, and which changes stay to finish or undo by hand—and asks you to confirm.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery abort <PLAN> [--reason <TEXT>] [--yes]
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`--reason <TEXT>`	Why the plan is being abandoned.
`--yes`	Skip the confirmation. Required when standard input is not a terminal.

Example

```
cdr recovery abort 7c41e0a2b9d3 --reason "Re-planning with dc03 as the restore source"
```

recovery attended

Takes an unattended plan back: from now on every pause waits for a person, as on any plan. A plan cannot be made unattended again once it is running.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery attended <PLAN>
```

Example

```
cdr recovery attended 7c41e0a2b9d3
```

recovery notify

Changes who a plan emails: its halts, aborts, and waits for a person, and its recovery report when it completes. The addresses replace the ones the plan had. Emails need the Manager's email settings; see Settings Reference.

Requirements: The `plan.execute` capability (Restore Operator role).

Syntax

```
cdr recovery notify <PLAN> (--to <ADDRESS>... | --clear)
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`--to <ADDRESS>`	Where the plan's notifications and report go. Repeatable, or comma-separated.
`--clear`	Stop emailing this plan's notifications. Give `--to` or `--clear`, not both.

Examples

```
cdr recovery notify 7c41e0a2b9d3 --to ad-team@corp.example.com
cdr recovery notify 7c41e0a2b9d3 --clear
```

recovery report

Gets the recovery report of a plan that has started: what the plan did, what went wrong on the way, and the manual follow-ups it left. It is built from the plan's records each time, so it describes the plan as it is now. Saved as HTML (the default, in the current folder as `cygnadr-recovery-<plan>.html`) or as a PDF, or emailed. For what it contains, see After a Recovery.

Requirements: The `plan.view` capability (every role) to save it; `plan.execute` (Restore Operator role) to email it. The Manager makes the PDF with its own renderer. Email needs the Manager's email settings; see Settings Reference.

Syntax

```
cdr recovery report <PLAN> [--html <FILE>] [--pdf <FILE>] [--email [--to <ADDRESS>...]]
```

Argument or option	Description
`<PLAN>`	**Required.** The plan ID.
`--html <FILE>`	Save the report as HTML here. The default when nothing else is asked for.
`--pdf <FILE>`	Save the report as a PDF here, made by the Manager's own PDF renderer. `cdr` waits for the PDF however short `--timeout` is.

Argument or option	Description
`--email`	Email the report, with its PDF attached when the Manager can make one.
`--to <ADDRESS>`	Who to email it to, with `--email`. Repeatable. Without it, the plan's notification addresses.

Examples

```
cdr recovery report 7c41e0a2b9d3
cdr recovery report 7c41e0a2b9d3 --pdf corp-recovery.pdf
cdr recovery report 7c41e0a2b9d3 --email --to ad-team@corp.example.com
```

Alert, Audit, and Report Commands

These commands answer three questions from a script: what is wrong now (`cdr alert list`), who did what (`cdr audit list`), and how backups have gone over a period (`cdr report show`). With `--json`, each gives you a record you can keep, and `cdr report show --json` is a machine-readable copy of the backup report. To save the audit trail as a file, use `cdr audit export`. `cdr alert notify status` and `cdr alert notify test` check the alert email.

For how to run `cdr` and the global options, see Command-Line Reference.

alert list

Lists the raised alerts, each with its severity, area, and the number of objects it is about. The command exits with code 4 when any alert is raised and 0 when none is, so you can run it from a scheduled task or a monitoring system as a probe. It exits with code 2, not 0, when the Manager cannot be reached, so an outage is never reported as all clear.

Alerts have no acknowledge or snooze. To stop an alert for an object you have deliberately put out of scope, add an exclusion (see Configuration and Exclusion Commands).

Requirements: The `dashboard.view` capability (every role).

Syntax

```
cdr alert list [options]
```

Options

Option	Description
`-s, --severity <CODE>`	Only alerts at this severity: `critical`, `warning`, `info`, or `unknown`.
`-o, --origin <CODE>`	Only alerts from this area: `protection`, `topology`, `storage`, `security`, or `recovery`.
`--critical`	Only critical alerts. Shorthand for `--severity critical`; give one or the other.
`--subjects`	Show the named objects under each alert, rather than only the count.
`--sources`	List the alert catalog instead of the raised alerts: every alert source, with its severity and area. Cannot be combined with `--subjects`.

Examples

```
cdr alert list --critical --subjects
cdr alert list --json > alerts.json
```

As a probe in a batch file:

```
cdr alert list --critical --manager https://10.0.0.10:5210
if %ERRORLEVEL% EQU 4 echo Critical alerts are raised
```

For what each alert means and what to do about it, see Alerts Reference.

alert notify status

Shows whether alerts are emailed and whether that is working: the recipients, whether the mail server is configured, the minimum severity, the check and reminder intervals, the last and next check, how many objects are raised at or above the minimum severity, and the last email with its counts or the reason it was not sent. The last check is the last one since the Manager started. When no recipients are set, the command says that alert email is off. Until the first check after the Manager starts, a few minutes after the service starts, and before any email has been sent, it says that alert email is not proven yet; `cdr alert notify test` checks it at once.

The command exits with code 4 when alert email is on and not working: the mail server is not configured, the last check failed, or the last email was not sent. Raised alerts are then

held and sent once it works. Run it from a monitoring system to watch the alert email itself. It exits with code 0 when email is working or off.

To change the settings, use `cdr config set` with the keys `CygnaDR:AlertEmail:Recipients`, `CygnaDR:AlertEmail:CheckMinutes`, `CygnaDR:AlertEmail:ReminderHours`, and `CygnaDR:AlertEmail:MinimumSeverity`, or the **Alert notifications** settings on the **Configuration** screen. See Settings Reference.

Requirements: The `dashboard.view` capability (every role).

Syntax

```
cdr alert notify status [options]
```

The command takes only the global options, such as `--json`, which prints the status as the Manager returned it.

Examples

```
cdr alert notify status
cdr config set CygnaDR:AlertEmail:Recipients "ops@example.com, ad-team@example.com"
```

alert notify test

Sends the alert recipients a test email in the real format, marked `[Test]`, listing what is raised now. It does not change what the next check reports. The Manager sends one test email at a time and at most one every 30 seconds, and records each one in the audit trail as **Test alert email requested** with the operator who asked.

Requirements: The `config.manage` capability (System Admin).

Syntax

```
cdr alert notify test [options]
```

The command takes only the global options.

Exit codes: 0 when the email was sent; 3 when it was not sent, with the reason (for example, no recipients are set, the mail server refused it, or a test email was sent less than 30 seconds ago), or when the Manager refused the request.

For the same status and test on the **Alerts** screen, see Alerts.

audit list

Lists audit trail rows, newest first: when, the actor, the action code, and the subject. The Manager applies the filters, so the rows are the newest that match across the whole audit

trail, not a filtered view of the newest few. The Manager returns at most the 200 newest matching rows, and the command says how many matched in total; narrow the filter or the time window to see older rows.

The actor is the username of the operator who acted, or `system`, `manager`, or `storage-agent` for actions the product took itself.

Requirements: The `audit.view` capability (every role).

Syntax

```
cdr audit list [options]
```

Options

Option	Description
`-a, --actor <NAME>`	Only rows recorded against this actor: an operator's username, or `system`, `manager`, or `storage-agent`.
`-A, --action <CODE>`	Only rows with this action code, for example `backup.job.completed` or `policy.updated`.
`-S, --subject <TEXT>`	Only rows about this subject—usually a machine or object name.
`--from <TIMESTAMP>`	Only rows at or after this UTC time, as `yyyy-MM-dd` or a full timestamp.
`--to <TIMESTAMP>`	Only rows at or before this UTC time.
`--detail`	Add each row's recorded detail, exactly as stored.
`--actors`	List the distinct actors and action codes in the whole audit trail, instead of the rows. Cannot be combined with any filter, including `--from` and `--to`.

Examples

```
cdr audit list --subject dc01.corp.example.com --detail
cdr audit list --actor jsmith --from 2026-10-01 --to 2026-10-07
cdr audit list --actors
cdr audit list --from 2026-09-01 --to 2026-09-30 --json > audit-2026-09.json
```

For the action codes and how to review operator activity, see Auditing Operator Activity.

audit export

Saves every audit trail row that matches the filters as a CSV, JSON, HTML, or PDF file: the same export as **Export** on the **Audit Trail** screen. Unlike `cdr audit list`, it is not limited to the 200 newest rows. CSV and JSON hold up to 100,000 rows, and HTML and PDF up to 5,000. When more rows match, the Manager refuses the export with the reason rather than cutting it short; narrow the dates or the filter.

The file holds action and actor codes, where the console's export adds its wording beside them, and times in UTC and in the time zone of the machine `cdr` runs on. In a CSV file, a value that starts with `=`, `+`, `-`, `@`, a tab, or a carriage return gets a leading apostrophe, so a spreadsheet does not run it as a formula. The Manager makes the PDF with its own renderer; `cdr` waits for it however short `--timeout` is.

Unless `--output` names a file, the file takes the Manager's name for it, `cygnadr-audit-<yyyyMMdd-HHmmss>.<format>`. The Manager records each export in the audit trail as `audit.exported`, with that file name as the subject, before the data leaves. `cdr` never overwrites an existing file.

Requirements: The `audit.view` capability (every role).

Syntax

```
cdr audit export [options]
```

Options

Option	Description
`-f, --format <FORMAT>`	`csv` (the default), `json`, `html`, or `pdf`.
`-o, --output <PATH>`	Where to save it: a file, or a folder to save it in under its own name. Default: the current folder.
`-a, --actor <NAME>`	Only rows recorded against this actor.
`-A, --action <CODE>`	Only rows with this action code, for example `backup.job.completed`.
`-S, --subject <TEXT>`	Only rows about this subject–usually a machine or object name.
`--from <TIMESTAMP>`	Only rows at or after this UTC time, as `yyyy-MM-dd` or a full timestamp.

Option	Description
`--to <TIMESTAMP>`	Only rows at or before this UTC time.
`--search <TEXT>`	Only rows whose actor, action code, subject, or UTC time contains this text, as the console's search box does.

With `--json`, `cdr` prints the saved path, the format, and the size in bytes instead of a sentence.

Examples

```
cdr audit export
cdr audit export --subject dc01.corp.example.com --format pdf
cdr audit export --from 2026-09-01 --to 2026-09-30 --format json -o C:\evidence\
```

Exit codes: 0 when the file was saved; 3 when the Manager refused the export, for example because more rows match than the format holds or the PDF could not be made; 6 when the file already exists, so nothing was overwritten; 1 for a usage error, such as an unknown format.

report show

Reports backup outcomes over a window: how many finished backups succeeded, failed, and were cancelled, grouped by the status and error codes the Manager stored rather than by message text. Only finished jobs are counted. The window includes both ends; by default it is the last 30 days, and it can be at most 366 days.

The coverage tables added by `--coverage` show each domain and forest as they stand now, not as they were during the window. In the forest table, excluded domains and domain controllers are left out of the counts and shown as `(n excluded)`. Both tables have a **REBUILT ON RECOVERY** column: domain controllers that are not backup targets, which a recovery rebuilds by promotion. They are not counted short of **PROTECTED**, except those a domain with too few backup targets needs backed up, which the **NEED A POLICY** column counts.

Requirements: The `reports.view` capability (every role).

Syntax

```
cdr report show [options]
```

Options

Option	Description
`--from <TIMESTAMP>`	Start of the window, inclusive, as `yyyy-MM-dd` or a full UTC timestamp.
`--to <TIMESTAMP>`	End of the window, inclusive. Default: now.
`--days <N>`	A window ending now: the last `N` days, from 1 to 366. Default: 30. Cannot be combined with `--from` or `--to`.
`--machines`	Add the per-machine table.
`--daily`	Add the per-day table, with one row for every day of the window, including days when nothing ran.
`--coverage`	Add the per-domain and per-forest coverage snapshot as it stands now.

Examples

```
cdr report show --days 90 --machines
cdr report show --from 2026-09-01 --to 2026-09-30 --daily
cdr report show --coverage --json > backup-report.json
```

For the console report and its CSV export, see Protection Summary.

Configuration and Exclusion Commands

The `config` commands read and change the settings the Manager holds for itself, the storage agent, the DC agents, and the Isolated Recovery Host agents, and show which machines are running the current settings. The `config backup` commands show and take the Manager's configuration backups. The `license` commands show the installed license and apply or remove a license key. The `exclusion` commands put objects out of scope, so that they stop counting toward alerts, and bring them back.

For how to run `cdr` and the global options, see Command-Line Reference.

config show

Shows every setting with its current value, product default, where the value comes from, and what it applies to. Settings are grouped in four scopes: `manager`, `storageAgent`, `dcAgent`, and `hostAgent`. With `--agents`, it shows instead what each machine reports it is running: the configuration revision, any values the agent has clamped to a safer limit, and restarts pending.

A password setting, such as `CygnaDR:Email:SmtpPassword`, is never shown: its value reads `(set)` or `(not set)`.

The keys it prints, such as `CygnaDR:Vault:Retention:MaxVersionsPerDc`, are what `cdr config set` takes. For what each setting does, see Settings Reference.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr config show [options]
```

Options

Option	Description
`-s, --scope <CODE>`	Only one scope: `manager`, `storageAgent`, `dcAgent`, or `hostAgent`.
`--set`	Only settings that have been changed from the product default. Cannot be combined with `--agents`.
`--agents`	List what each machine reports it is running instead: configuration revision, clamped values, and restarts pending.
`--behind`	With `--agents`, only machines that are not running the current revision.

Examples

```
cdr config show --set
cdr config show --scope storageAgent
cdr config show --agents --behind
```

config set

Changes one setting, for the whole fleet or, with `--agent`, for one agent. The Manager validates the value and saves a new configuration revision; connected agents receive it at once, and agents that are offline fetch it when they reconnect. If the value was already set, nothing changes. `cdr config show --agents --behind` shows which machines have not applied the change yet.

The scope is worked out from the key when only one scope has it. If more than one does, name it with `--scope`. If you mistype a key, `cdr` suggests keys with a similar name.

Some settings can delete backups. For example, shortening a retention limit deletes the versions that fall outside it, and they cannot be brought back. For these destructive settings, `cdr` warns you and asks for confirmation, including when you reset one to its default. In a script, pass `--yes`; without it, the change is refused when there is no terminal.

To set a setting to an empty string, pass `""` as the value. This is how some settings are switched off, such as the OS media folder. Omitting the value is not the same; use `--reset` to go back to the product default.

A password setting, such as `CygnaDR:Email:SmtpPassword`, is never taken on the command line, where it would be kept in the shell's history and shown in process listings. `cdr` refuses a value given there. Omit the value: `cdr` reads it from the `CYGNADR_SETTING_SECRET` environment variable when that is set, and otherwise prompts for it without echoing it. With no terminal and no variable, the command is refused. Pressing Enter at the prompt without typing anything changes nothing; `--reset` clears the password. The confirmation says only whether the password was set before and whether a new value is being set, never what it is.

Requirements: The `config.manage` capability (System Admin role).

Syntax

```
cdr config set <KEY> [VALUE] [options]
```

Arguments

Argument	Description
`<KEY>`	**Required.** The setting's key, as `cdr config show` prints it.
`[VALUE]`	The new value. Required unless you give `--reset`. For a password setting, omit it: `cdr` prompts for the password, or reads `CYGNADR_SETTING_SECRET`.

Options

Option	Description
`-s, --scope <CODE>`	The scope the key belongs to: `manager`, `storageAgent`, `dcAgent`, or `hostAgent`. Default: worked out from the key when only one scope has it.
`--agent <ID>`	Set the value for one agent instead of fleet-wide. Takes the agent ID from `cdr agent list --wide`.

Option	Description
`--reset`	Go back to the product default instead of storing a value. Cannot be combined with a value.
`--yes`	Skip the confirmation. Required for a destructive setting when standard input is not a terminal.

Examples

```
cdr config set CygnaDR:MaxConcurrentBackups 4
cdr config set CygnaDR:Vault:Retention:MaxAgeDays 60
cdr config set CygnaDR:HeartbeatSeconds 30 --scope dcAgent
cdr config set CygnaDR:Media:Folder --reset
cdr config set CygnaDR:Email:SmtpHost smtp.corp.example.com
cdr config set CygnaDR:Email:SmtpPassword
```

config backup

Shows whether the Manager's configuration backups are running, where the copies are, and how the recent attempts ended: the same as the **Configuration backups** card on the **License** screen. `cdr config backup` and `cdr config backup show` are the same command.

When backups are not running, the command first prints a warning with the reason. It then prints the schedule and when a backup was last stored, a warning if the storage machines hold backups written under an earlier recovery passphrase, a table of storage machines with the columns `STORAGE`, `CONNECTED`, `HELD`, `NEWEST`, and `FOLDER`, and the last ten attempts with the columns `STARTED`, `OUTCOME`, `BY`, `STORAGE`, `SIZE`, and `DETAIL`. The `FOLDER` column is where to find the backups when you rebuild the Manager.

The command exits with code 0 when configuration backups are running, and 4 when they are not, for example because the recovery passphrase is not set. With `--json`, it writes the Manager's whole answer. For what a configuration backup is, see Manager Configuration Backups.

Requirements: The `config.manage` capability (System Admin role).

Syntax

```
cdr config backup [show] [options]
```

Examples

```
cdr config backup
cdr config backup show --json
```

config backup run

Takes a configuration backup now, as **Back up now** does on the **Configuration backups** card. The Manager writes the backup and sends it to the storage machine, and the command waits until it has finished and lists how each storage machine's attempt ended.

If the Manager does not start a backup because one is already running, the command says so and exits with code 3. It exits with 0 when every storage machine stored the backup, and 7 when any did not or the backup could not be taken, for example because the recovery passphrase is not set; the `DETAIL` column says why. If the backup is still running when the wait ends, the command says so and exits with 8; the backup carries on on the Manager, and `cdr config backup` shows how it ended. With `--json`, it writes whether the backup is still running and this run's attempts.

Requirements: The `config.export` capability (System Admin role).

Syntax

```
cdr config backup run [options]
```

Options

Option	Description
`--wait <MINUTES>`	How long to wait for the backup to be stored. At least 1. Default: 10.

Examples

```
cdr config backup run
cdr config backup run --wait 30
```

license show

Shows the installed license, where it stands today, and the enabled user accounts beside its licensed count. The `LICENSE` table lists the `state` (`licensed`, `expiring`, `grace period`, `expired`, `does not license Cygna Forest Recovery for Active Directory`, or `none installed`) with the license's sentence. Then come `licensed to` and `issued` for the whole key, then `module` with the module's name and, indented below it, its `type`, `people`, `ends`,

`operations stop`, and `maintenance ends`, and last `installed`. When new backups and recovery operations are refused, the command says so.

A second table lists each forest with `ENABLED USERS`, `DOMAINS COUNTED`, and `OLDEST COUNT`, followed by the total against the licensed people. The total is a guide only: the license counts people, and one person often holds several accounts. Domains with no DC agent are not counted. For what the license enforces, see Licensing.

The command exits with code 0 whatever the license's standing. With `--json`, it writes the Manager's whole answer, including the state, whether operations are allowed, the days to the end date and to the stop, and the per-forest counts.

Requirements: A signed-in session (every role).

Syntax

```
cdr license show [options]
```

Examples

```
cdr license show
cdr license show --json
```

license preview

Shows what a license key says—its standing and terms—without installing it. The command prints the same `LICENSE` table as `cdr license show`. If the Manager cannot use the key, the command prints the reason and exits with code 3.

Requirements: The `config.manage` capability (System Admin role).

Syntax

```
cdr license preview <KEY> [options]
```

Arguments

Argument	Description
`<KEY>`	**Required.** The license key, as Cygna sent it. Quote it if your shell would split it.

license apply

Installs a license key in place of the installed one, if there is one. On success, the command prints `Installed the license key.` and the `LICENSE` table for the new key. If backups and recovery operations were stopped, they resume at once. The installation is recorded in the audit trail.

The Manager refuses a key that is not a Cygna version 2 license key, or that does not license Cygna Forest Recovery for Active Directory; the command prints the reason and nothing is installed.

Requirements: The `config.manage` capability (System Admin role).

Syntax

```
cdr license apply <KEY> [options]
```

Arguments

Argument	Description
`<KEY>`	**Required.** The license key, as Cygna sent it. Quote it if your shell would split it.

Exit codes

Code	Meaning
`0`	The key was installed.
`3`	The key was refused, or the Manager refused the request. Nothing was installed.

Examples

```
cdr license apply "<KEY>"
```

license remove

Removes the installed license. New backups, restores, recovery plans, and test restores are then refused until a key is applied; work already under way finishes. The command warns you and asks for confirmation. If no license is installed, it says so and exits with code 0. The removal is recorded in the audit trail.

Requirements: The `config.manage` capability (System Admin role).

Syntax

```
cdr license remove [options]
```

Options

Option	Description
`--yes`	Skip the confirmation. Required when standard input is not a terminal.

Examples

```
cdr license remove --yes
```

About exclusions

An exclusion records that you know about an object and have deliberately put it out of scope—for example, a domain controller that is being decommissioned, or an orphaned domain partition that will never have a domain controller. An excluded object stops counting toward alerts. It is still discovered and still listed, and its verdict is still computed and shown: an exclusion changes what is counted, never what is shown.

- **A reason is required**—The reason is kept with the exclusion, so that anyone can see later why the object stopped counting.
- **Review date or permanent**—Give a review-by date, after which the exclusion lapses and the object raises alerts again, or record it as permanent on purpose with `--permanent`. You must choose one.
- **Containers pass it down**—Excluding a forest makes every domain, site, and domain controller in it inherit the exclusion. Excluding a domain controller does not change its domain's coverage verdict; exclude the domain itself to stop the domain raising alerts.

For when to use exclusions and retirements, see Excluding and Retiring Objects.

exclusion list

Lists every exclusion with its state, review-by date, when and by whom it was set, and the reason. Exclusions that have lapsed or been voided are listed too, because their objects raise alerts again. One on a domain controller or domain since recorded as removed is marked `(retired)` and counted apart, under `on retired objects`: nothing can alert for it and no review is due, and `cdr exclusion remove` clears it.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr exclusion list [options]
```

Options

Option	Description
`-k, --kind <CODE>`	Only exclusions on this kind of object: `dc`, `domain`, `site`, or `forest`.
`--active`	Only exclusions still in effect.
`--spent`	Only exclusions that have lapsed or been voided: due for review, or on an object since retired. Cannot be combined with `--active`, `--due`, or `--permanent`.
`--due <DAYS>`	Only active exclusions whose review-by date falls within this many days.
`--permanent`	Only active exclusions recorded with no review-by date. Cannot be combined with `--due`.

Examples

```
cdr exclusion list --due 30
cdr exclusion list --permanent
cdr exclusion list --spent
```

exclusion add

Puts an object out of scope. If the object is already excluded, the new exclusion replaces the old one; both are audited.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr exclusion add <KIND> <KEY> --reason <TEXT> (--review-by <DATE> | --permanent)
```

Arguments

Argument	Description
`<KIND>`	**Required.** What is being excluded: `dc`, `domain`, `site`, or `forest`.
`<KEY>`	**Required.** The object: a domain controller's FQDN, a domain's FQDN, a site's distinguished name, or a forest's FQDN.

Options

Option	Description
`-r, --reason <TEXT>`	**Required.** Why the object is out of scope.
`--review-by <DATE>`	When to reconsider, as `yyyy-MM-dd` or a full timestamp such as `2027-03-01T09:00:00Z`. Give this or `--permanent`.
`--permanent`	Record the exclusion with no review date, on purpose. Give this or `--review-by`.

Examples

```
cdr exclusion add dc dc09.corp.example.com --reason "Decommissioned; awaiting metadata cleanup" --review-by 2026-12-01
cdr exclusion add domain blue.corp.example.com --reason "Orphaned partition, no DCs and never will have" --permanent
cdr exclusion add site "CN=Branch-Test,CN-N=Sites,CN=Configuration,DC=corp,DC=example,DC=com" --reason "Test site with no production DCs" --review-by 2027-01-15
```

exclusion remove

Clears an exclusion. The object counts again from the next time the Manager reads it. If the object has no exclusion of its own but inherits one from its domain or forest, clear the exclusion on the container instead; `cdr exclusion list` shows where each exclusion is recorded.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr exclusion remove <KIND> <KEY>
```

Argument	Description
`<KIND>`	**Required.** What was excluded: `dc`, `domain`, `site`, or `forest`.
`<KEY>`	**Required.** The object: a domain controller's FQDN, a domain's FQDN, a site's distinguished name, or a forest's FQDN.

Example

```
cdr exclusion remove dc dc09.corp.example.com
```

Directory Cleanup Commands

The `cleanup` commands remove what is left in Active Directory after a domain controller, a domain, or a trust is gone: a dead domain controller's NTDS Settings, server, and computer objects and its replication memberships; an orphaned domain's remaining domain controller metadata, its crossRef, and its partition; or this forest's side of a trust. They also record in the Manager that an object no longer exists, so that its inventory stops presenting it.



Note: **Important:** A cleanup cannot be undone. To bring back a domain controller whose metadata has been cleaned up, you must promote a new one. Run the plan first, read it, and confirm only when every object it lists should go.

For how to run `cdr` and the global options, see Command-Line Reference.

How a cleanup runs

1. **Plan (the default)**—Without `--confirm`, `cdr cleanup dc`, `domain`, and `trust` are a dry run. They bind to a domain controller, read the directory, and print every object they would remove, in the order they would remove it, followed by any refusals. Nothing is changed.
2. **Checks**—Anything that can be checked beforehand is checked, and a failed check is a refusal, not a warning. For example, a domain controller that still answers is refused, because removing the metadata of a running domain controller is the worst outcome available. A few refusals can be passed with `--force` when you know something the tool cannot; the others cannot be overridden. See Refusal codes.
3. **Removal**—With `--confirm` and the object's own name, the objects are removed in order. NTDS Settings goes first, because that is what makes Active Directory treat the domain controller as removed and replicate the removal; the rest is tidying up the

objects it leaves behind. If an object cannot be removed, the cleanup stops there and leaves everything after it alone, and the command exits with code 7. An object that is already gone is reported as absent, which is normal after a partial cleanup by hand.

4. **Retirement**—After a successful removal, `cdr` tells the Manager that the object is gone, with the reason you gave. If nothing was left to remove, a confirmed run still records the retirement. If recording it fails, the directory change stands, and `cdr` tells you the command that records it. Use `--skip-retire` to leave the Manager's inventory unchanged.

Exit codes: 0 when the plan was shown or the cleanup succeeded; 4 when the cleanup is refused; 7 when it stopped partway.

Required rights

- **Directory rights**—An account with Domain Admin rights in the affected domain, or Enterprise Admin rights. Removing a domain from the forest needs Enterprise Admin rights. `cdr` uses your own Windows identity, or the account in `--user`, whose password is read from the `CYGNADR_CLEANUP_PASSWORD` environment variable or prompted for. The credential goes only to the directory; the Manager never sees it, and the domain controller's security log records who removed what.
- **Manager capabilities**—Without `--server`, `cdr` asks the Manager for a domain controller it knows is up, which needs the `environment.view` capability. Recording a retirement needs the `backup.manage` capability (Backup Operator role).
- **Windows**—The `cleanup dc`, `domain`, and `trust` commands run on Windows only, and need LDAP access to the domain controller they bind to.

Options shared by cleanup dc, domain, and trust

Option	Description
`-s, --server <DC>`	The domain controller to bind to, by name or IP address. Default: one the Manager knows is up.
`-u, --user <ACCOUNT>`	The account to use, as `DOMAIN\user` or `user@domain`. The password is read from `CYGNADR_CLEANUP_PASSWORD`, or prompted for. Default: your own Windows identity.
`--confirm <NAME>`	Type the object's own name—the domain controller's FQDN, the domain's FQDN, or the trust partner's name—to carry out the removal. Without it, the command only shows the plan.
`--force`	Pass the refusals that can be overridden. Never all of them. Only with `--confirm`.

Option	Description
`--skip-dns`	Leave DNS records in place.
`--skip-sites`	Leave a site object in place when its last server is removed.
`--skip-retire`	Do not tell the Manager that the object is gone. Its inventory keeps showing it.
`--reason <TEXT>`	Why the object was removed. Recorded with the Manager's record of the object. Required with `--confirm`, unless you give `--skip-retire`.

cleanup list

Lists what the Manager's inventory suggests can be cleaned up, and why: domain controllers that have been missing from every directory read for a period. With `--retired`, it lists the machines already recorded as removed instead.

Requirements: The `environment.view` capability (every role).

Syntax

```
cdr cleanup list [options]
```

Options

Option	Description
`--stale-days <DAYS>`	How long a domain controller must have been missing from every directory read to be listed. Minimum: 1. Default: 30.
`--retired`	List machines already recorded as removed, instead of candidates for removal.

Examples

```
cdr cleanup list --stale-days 90
cdr cleanup list --retired
```

cleanup dc

Removes a dead domain controller's objects and metadata from the directory: its NTDS Settings and DNS Settings objects, its server object, its computer object, its FRS or DFSR

replication memberships, and, unless you skip them, its DNS records and a site left with no servers.

The plan is refused if the domain controller still answers, holds an operations master role (seize the roles elsewhere first), or is the last domain controller in its domain.

Requirements: See Required rights.

Syntax

```
cdr cleanup dc <FQDN> [options]
```

Argument	Description
`<FQDN>`	**Required.** The domain controller to remove from the directory.

Takes the shared cleanup options.

Examples

```
cdr cleanup dc dc09.corp.example.com --server dc01.corp.example.com
cdr cleanup dc dc09.corp.example.com -u CORP\Administrator --confirm dc09.corp.example.com --reason "Hardware failed; replaced by dc11"
```

cleanup domain

Removes a dead domain from the forest: the metadata of its remaining domain controllers, the application partitions beneath it that no other domain controller hosts, this forest's own trusts that name it, and then its crossRef and partition.

The plan is refused for the forest root domain, for a domain that still has domain controllers or that a trust still references, and for a domain with a child domain or an application partition still in use. A domain whose domain controller objects are only leftover metadata can be removed with `--force`, after you have confirmed that none of them is running.

Requirements: See Required rights. Removing a domain needs Enterprise Admin rights.

Syntax

```
cdr cleanup domain <FQDN> [options]
```

Argument	Description
`<FQDN>`	**Required.** The domain to remove from the forest.

Takes the shared cleanup options.

Examples

```
cdr cleanup domain blue.corp.example.com
cdr cleanup domain blue.corp.example.com --force --confirm blue.corp.example.com --reason
"Orphaned crossRef; its DC objects are residue"
```

cleanup trust

Removes this forest's side of a trust with another forest, domain, or realm, together with its trust account. The partner keeps its own side; remove that separately on the partner. The plan says what kind of trust it is about to remove.

Requirements: See Required rights.

Syntax

```
cdr cleanup trust <PARTNER> [options]
```

Argument or option	Description
`<PARTNER>`	**Required.** The trust partner: the other forest, domain, or realm.
`--from <DOMAIN>`	Which of this forest's domains holds the local side. Default: found automatically when there is only one.

Also takes the shared cleanup options.

Examples

```
cdr cleanup trust partner.example.com --from corp.example.com
cdr cleanup trust partner.example.com --confirm partner.example.com --reason "Partnership ended;
agreed removal date passed"
```

cleanup retire

Records that a domain controller, domain, or trust is no longer defined in the directory. Use it for an object that someone removed by other means, so that there is nothing left for a cleanup command to remove. It changes the Manager's inventory only. The Manager checks the claim: it refuses, and the command exits with code 4, if a current complete read of the directory still lists the object, or if the object is not in its inventory.

Discovery records most removals by itself. Use this command for removals it holds for review, or that it has not yet waited long enough to confirm.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr cleanup retire <NAME> --reason <TEXT> [options]
```

Argument or option	Description
`<NAME>`	**Required.** What the directory no longer defines: a domain controller's FQDN, a domain's FQDN, or a trust partner's name. A trust can also be named `from-domain->partner`, as alerts print it.
`--kind <KIND>`	`dc`, `domain`, or `trust`. Default: `dc`.
`--from <DOMAIN>`	With `--kind trust`, only this domain's trust to the partner. Default: every domain's trust to the partner.
`--reason <TEXT>`	**Required.** Why it is gone. Recorded with the Manager's record of the object.

Examples

```
cdr cleanup retire dc02.corp.example.com --reason "Demoted and metadata-cleaned by hand in 2023"
cdr cleanup retire partner.example.com --kind trust --from corp.example.com --reason "Forest trust removed in Active Directory Domains and Trusts"
```

cleanup unretire

Withdraws the Manager's record that an object is gone, so that the inventory presents it again. It changes the Manager's inventory only, and nothing in the directory.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr cleanup unretire <KIND> <KEY>
```

Argument	Description
`<KIND>`	**Required.** What was recorded as gone: `dc`, `domain`, or `trust`.
`<KEY>`	**Required.** The object: a domain controller's FQDN, a domain's FQDN, or the trust partner's name.

Example

```
cdr cleanup unretire dc dc09.corp.example.com
```

cleanup forget

Deletes a retired domain controller's record from the Manager when nothing else refers to it. It does not touch the directory. Without `--confirm`, it says what would happen. The Manager keeps the record while any of these is true, and the command lists which:

- It is not recorded as removed from the directory.
- An agent is still enrolled under that name.
- Backup jobs exist for it.
- A backup schedule still names it.

Requirements: The `backup.manage` capability (Backup Operator role).

Syntax

```
cdr cleanup forget <FQDN> [--confirm <FQDN>]
```

Argument or option	Description
`<FQDN>`	**Required.** The retired domain controller whose record is deleted from the Manager.
`--confirm <NAME>`	Type the machine's own name to delete its record. Without it, `cdr` only says what would happen.

Examples

```
cdr cleanup forget dc02.corp.example.com
cdr cleanup forget dc02.corp.example.com --confirm dc02.corp.example.com
```

Refusal codes

The plan output names each refusal by its code. Refusals marked as overridable can be passed with `--force` together with `--confirm`; `cdr` still prints them before anything runs, and the JSON output lists them as overridden.

Code	Meaning	Overridable
`target-is-alive`	The domain controller answered when contacted. It may be running and replicating, or the answer may come from a stale DNS record or a rebuilt machine with the same name.	Yes
`holds-fsmo-roles`	The domain controller holds an operations master role. Seize the role on another domain controller first.	No
`last-dc-in-domain`	Removing it would leave its domain with no domain controllers. To remove the domain, use `cdr cleanup domain`.	No
`domain-unknown`	The domain the machine belonged to could not be worked out, typically after a partial demotion. Only its configuration partition objects can be removed.	Yes
`presence-unknown`	Whether an object exists could not be established, because the read failed. Bind to a domain controller that can see it with `--server`.	No
`server-has-unrecognised-children`	The server object has a child object other than NTDS Settings and DNS Settings. Look at it before you remove anything.	No
`is-forest-root`	The forest root domain cannot be removed.	No
`not-a-domain-partition`	The crossRef names a partition that is not a domain, such as a DNS application partition.	No
`domain-still-has-dcs`	The domain still shows domain controllers. Remove them first, one at a time. If they are leftover metadata of machines long gone, override.	Yes
`child-partition-in-use`	A child domain, or an application partition	No

Code	Meaning	Overridable
	hosted by a domain controller outside the plan, is beneath the domain. Remove the child domain first.	
`domain-still-trusted`	A trust still references the domain being removed.	No
`trust-unconfirmed`	The global catalog shows another domain holding a trust that names the domain being removed, and no domain controller of that domain could be read to confirm it. Override when that domain is itself gone.	Yes
`not-found`	Nothing by that name was found to clean up.	No
`not-confirmed`	The `--confirm` value did not match the object's own name.	No

Support Commands

The `support` commands gather the evidence for a support case from the Manager and every agent into one file: every machine's logs, the Manager's database with all credentials removed, and host facts. `cdr support package` is the command-line twin of the **Support package** card on the **License** screen, with more control over which agents are included and how much each sends.

For what a support package contains, what its database copy leaves out, and how long the Manager keeps it, see Logs and Diagnostics. For how to run `cdr` and the global options, see Command-Line Reference.



Note: A support package is confidential and not encrypted; `cdr` and the package's `README.txt` mark it `CONFIDENTIAL`. Credentials are removed, but it still describes your forest in detail. Store and send it as you would your own network documentation, and delete it when the support case is closed.

support package

Asks the Manager to build a support package, follows the collection, downloads the file, and then deletes the Manager's copy. While the Manager collects, the command prints a line for each machine as it finishes, with its state, file count, size, and any detail. Every 30 seconds without news, it says which machines it is still waiting for. At the end it prints a table of every source (SOURCE, KIND, STATE, FILES, SIZE, DETAIL), the path and size of the saved file, and the confidentiality warning.

The Manager builds one package at a time, and refuses to start one when its data disk has less than 2 GB free. If the Manager is still collecting when `--wait` runs out, the package carries on there; download it later with `--id`. The Manager keeps it for 24 hours after it finished.

The command never overwrites a file. If the output file already exists, it saves nothing and leaves the package on the Manager. A download that does not finish is deleted.

Requirements: The `config.export` capability (System Admin role).

Syntax

```
cdr support package [options]
```

Options

Option	Description
`-o, --output <PATH>`	Where to save the package: a file, or a folder to save it in under the Manager's file name for it. A path is a folder when it exists as one or ends in `\`. Default: the current folder.
`--days <N>`	Logs last written within this many days, from 1 to 366. Default: 3. Cannot be combined with `--all-logs`.
`--all-logs`	Everything each machine still retains, whatever its age. Can be large on a big estate.
`--agent <NAME>`	Only this agent's logs, by machine name. Repeat the option for more agents. A name that is not an enrolled DC agent, storage agent, or Isolated Recovery Host agent is refused. Default: every connected agent.
`--no-agents`	The Manager's part only: its logs, its database copy, and host facts. Cannot be combined with `--agent`.
`--max-agent-mb <MB>`	The most any one agent sends, newest files first, from 1 to 8192. Default: 512.
`--wait <MINUTES>`	How long to wait for the Manager to finish collecting, at least 1. Default: 60.

Option	Description
`--id <ID>`	Download a package already started, for example after `--wait` ran out, instead of starting a new one. Cannot be combined with `--days`, `--all-logs`, `--agent`, `--no-agents`, or `--max-agent-mb`, because what goes in the package was decided when it was started.
`--keep-on-manager`	Leave the package on the Manager after saving it. By default it is deleted there once saved. The Manager deletes it 24 hours after it finished either way.

With `--json`, the command writes the saved file's path and size, whether it was removed from the Manager, and the package's full status, including each source.

Exit codes

Code	Meaning
`0`	Saved, and every machine was collected in full.
`3`	The Manager refused to start the package, or the package failed on the Manager.
`6`	The output file already exists. Nothing was overwritten, and the package stays on the Manager.
`7`	Saved, but at least one machine is missing or partial—for example, an agent that was offline or over its size limit. The table gives the reason.
`8`	Still collecting when `--wait` ran out. Download it later with `--id`.

Examples

```
cdr support package
cdr support package --days 7 --output C:\cases\
cdr support package --all-logs --agent dc01.corp.example.com --agent dc02.corp.example.com
cdr support package --no-agents
cdr support package --id 3f9a1c0b7e21
```

support list

Lists the support packages the Manager holds now: one being collected, or ones not yet downloaded and removed. Columns: ID, STATE, LOGS (the period collected), CREATED,

BY, SIZE (`not yet` until the package is ready), and MACHINES (how many machines were collected in full, of how many). To save one, run `cdr support package --id <id>`.

Requirements: The `config.export` capability (System Admin role).

Syntax

```
cdr support list
```

When the Manager service does not start

`cdr` needs the Manager service. If the service does not start, run `CygnaDR.Manager.exe support package` on the Manager instead. It builds the Manager's part of a package without the service, with no agent logs. See Logs and Diagnostics.

Glossary

This glossary defines the terms this guide uses with a specific meaning in Cygna Forest Recovery.

A

- **abandon**—To close a restore run that cannot be cancelled because its machine is gone or will be rebuilt. A reason is required and is recorded in the audit trail. See Recovery Lifecycle.
- **abort**—To end a running recovery plan for good. An aborted plan cannot be resumed; the recovery is planned again instead. See Recovery Lifecycle.
- **abort class**—What stopping a recovery step would leave behind: safe to stop (nothing was changed), stopping needs cleanup (the machine was changed), or point of no return. A restore run's **In progress** card shows it, and a recovery plan's abort dialog says what aborting at that point leaves.
- **advisory gate**—A preflight check result that you should read before you approve a restore or recovery plan, but that does not block approval. See **gate**.
- **agent ID**—The identifier the Manager gives an enrolled agent. Commands that act on one agent, such as recovering onto a candidate machine, take it.
- **approve**—To authorize a planned restore or recovery plan once its hard gates are clear. Approving needs a different capability from executing, so one person can approve and another run. See Recovery Lifecycle.
- **at risk**—A coverage verdict for a domain or forest that could be recovered today, but with a weakness that needs attention—for example, a domain whose recovery depends on a single domain controller. See Coverage and Readiness.
- **attestation**—An operator's recorded confirmation that a recovery step performed or checked outside Cygna Forest Recovery is done—for example, that the recovered domain controllers are isolated. Rejecting the step instead fails it and halts the plan.
- **authoritative restore**—A restore in which chosen objects are marked so that they replicate out to the other domain controllers instead of being overwritten by them. A DC state recovery of one domain controller restores the subtrees and objects you name authoritatively. See Restore Methods.

B

- **backup target**—A domain controller with an enabled backup policy. Protection verdicts and alerts about missing backups apply to backup targets; a domain controller that is

not one is rebuilt on recovery. See Backup Targets and Rebuilt Domain Controllers.

- **breach flag**—A recovery plan option for recovering from a security breach. It adds the breach-response resets on every restored domain controller, and in a forest recovery it resets every trust. Without it, a forest recovery resets no trust. See Recovering a Forest.
- **break-glass account**—The operator account created at the first console sign-in. It holds every capability, its password is shown once, and no administrator can change it. Use it when no named account is available, and store its credential offline. See Signing In for the First Time.

C

- **CA fingerprint**—The SHA-256 hash of the Manager's certificate authority, shown as 16 groups of four hexadecimal digits. Agents and the `cdr` tool use it to recognize the Manager. It is public, not secret. See Certificates and Trust.
- **cancel**—To stop a backup job, or a restore before it starts writing to the machine. A test restore can be cancelled at any point. See also **discard** and **abandon**.
- **candidate machine**—A bare machine, not a domain controller, with the DC agent enrolled, other than the storage machine or an Isolated Recovery Host, that can receive a full system recovery when the original domain controller will not start. It is listed on the Agents screen, not the Domain Controllers screen. See Full System Recovery onto a Candidate.
- **capability**—A single permission, such as starting backups or approving restores. Roles grant capabilities, and the Manager checks them on every request. See Managing Operators and Roles.
- **cdr**—The command-line tool installed with the Manager, for scripting, monitoring, and tasks the console does not perform. See Command-Line Reference.
- **change journal**—The record an NTFS or ReFS volume keeps of every change to its files. The storage agent reads the vault volume's change journal to detect changes to committed backups that retention did not make. See Detecting changes to committed backups.
- **configuration backup**—An encrypted copy of the Manager's database, enrollment token, and secret settings that the Manager sends to the storage machine on a schedule, daily by default. Only the recovery passphrase that was current when it was written opens it. See Manager Configuration Backups.
- **confirm gate**—A preflight check result on a restore run that blocks approval until the approver accepts it by its code, or the run is cancelled. It reports something that failed from where the Manager looked but may not fail from where the recovery environment runs. See Preflight Check Reference.

- **console**—The web application the Manager serves, where operators monitor protection and run backups and recoveries. See Using the Console.
- **covered**—The best domain coverage verdict: two or more writable domain controllers hold full system backups, and the newest backup is fresh and inside the tombstone lifetime. A domain with a warning verdict can also be recovered today, but is at risk. See Coverage and Readiness.

D

- **DC agent**—The service installed on every domain controller, and on candidate machines. It reports topology, runs backups, and carries out restores and recovery steps. It only connects out to the Manager and listens on nothing. See Components and Architecture.
- **DC state backup**—A smaller, faster backup that captures Active Directory and the domain controller's Windows system state, including the registry. It supports DC state recovery on a domain controller that still boots, but not full system recovery. See Backup Types.
- **DC state recovery**—A restore of a domain controller whose Windows installation still starts. The DC agent restores DC state from a backup in Directory Services Restore Mode, then restarts normally. Planned for one domain controller, it marks the subtrees and objects you name authoritative, so they go back to the backup on every domain controller of the domain; in a domain or forest plan it is non-authoritative. It works from either backup type. See Restoring a DC with DC State Recovery.
- **discard**—To remove a planned or approved restore that was never executed, such as one whose preflight failed.
- **domain recovery**—Recovery of one domain while the rest of the forest stays live. See Recovering a Domain.
- **DSRM password**—The Directory Services Restore Mode administrator password of a domain controller. A domain or forest recovery sets a new one, which you supply, on every domain controller it recovers. The Manager holds it in memory only and never shows it again.

E

- **enrollment token**—The shared secret an agent presents to enroll with the Manager. A valid token enrolls an agent immediately, with no approval step, so keep it secret. See Certificates and Trust.
- **exclude (recovery plan)**—In a domain or forest recovery plan, to neither restore nor rebuild a domain controller. Its metadata is removed from the directory along with the

domain's other domain controllers, and it is not brought back; keep the machine off the network. Not the same as an **exclusion**.

- **exclusion**–An operator's decision to put an object out of scope, with a mandatory reason and an optional review date. An excluded object stops counting toward alerts but is still discovered and shown: an exclusion changes what is counted, never what is shown. See Excluding and Retiring Objects.
- **execute**–To start an approved restore or recovery plan. Executing needs its own capability, separate from approving.

F

- **forest recovery**–Recovery of a whole forest following the Microsoft forest recovery procedure: the forest root domain first, then each child domain after its parent. See Recovering a Forest.
- **full system backup**–A backup of every critical volume a domain controller needs to start (`wbadmin -allCritical`), including Windows, applications, and DC state. It supports both DC state recovery and full system recovery, and is the default backup type. See Backup Types.
- **full system recovery**–A restore of a whole server from a full system backup through the recovery image, in place, onto a candidate machine, or onto a new virtual machine booted from a recovery ISO. Use it when the operating system must be restored as well as the directory, or the hardware is replaced. A recovery in place needs the domain controller's Windows to start once; when it will not start, recover onto a candidate machine or a new virtual machine. See Restore Methods.

G

- **gate**–The result of one preflight check on a restore or recovery plan. A hard gate blocks approval until it is cleared; a confirm gate blocks approval of a restore run until you accept it; an advisory gate is for you to read. See Preflight Check Reference.

H

- **hard gate**–A preflight check result that blocks approval of a restore or recovery plan until it is cleared. See **gate**.
- **hash verification**–The storage agent's scheduled, read-only re-check that each stored backup still matches the content hash taken when it was committed. Restores also verify the hash before they read a backup. See Vault Protection and Integrity.

I

- **in place**—A full system recovery onto the domain controller's own hardware. Its Windows installation must start once so that its agent can stage the recovery image. See Full System Recovery in Place.
- **Isolated Recovery Host**—An optional Hyper-V host, not joined to any domain, on which test restores run. It runs the Isolated Recovery Host agent, which builds the test restore virtual machines and isolates them. A host is given test restores only once an operator approves it. See Setting Up an Isolated Recovery Host.
- **isolation**—Cutting the domain controllers being recovered off from the production network, so that recovered and damaged copies of the directory cannot replicate with each other. The targets keep only firewalled paths to the Manager and the storage machine. An operator attests it during a domain or forest recovery. A test restore is isolated differently: the restored system runs on a private switch on the Isolated Recovery Host that reaches nothing. See Isolating Recovery Targets.

L

- **license**—The Cygna license key that entitles an installation to back up and recover. Its count of people is shown beside the enabled user accounts. After its end date and a 14-day grace period (none for a trial), or with no key installed, the Manager refuses new backups, restores, recovery plans, and test restores. See Licensing.

M

- **Manager**—The central service, on a machine that is not joined to the domain. It holds the database, its own certificate authority, the scheduler, inventory, recovery plans, settings, operator accounts, and the audit trail, and serves the console. See Components and Architecture.
- **manifest**—The document stored with each backup that describes the machine at the time: identity, operating system, hardware, network, directory roles, and backup details. It is covered by the backup's content hash, and full system recovery is planned from it. See Restore Points.
- **multi-factor authentication (MFA)**—An optional second sign-in step that each operator turns on for their own account: after the password, a 6-digit code from an authenticator app on their phone. See Multi-Factor Authentication.

N

- **newest**—The default retention mode, which keeps the most recent versions of each domain controller's backups. See Retention.
- **notification addresses**—The email addresses a recovery plan tells when it halts, is aborted, or waits for a person, and sends its recovery report to when it completes.

O

- **operator**—A person who signs in to the console or the `cdr` tool with an account the Manager holds. Operator accounts are not Windows or Active Directory accounts, so they work while the directory is down. See Managing Operators and Roles.
- **OS media**—The Windows Server installation ISO files you supply, from which the storage agent builds recovery images. See Setting Up OS Media and Recovery Images.

P

- **point of no return**—A recovery step that cannot be undone, such as seizing operations master roles or cleaning up metadata. A plan stops before each one and runs it only after an operator acknowledges it by typing the step's code. Every acknowledgment is audited.
- **post-recovery directive**—A signed instruction that a restored domain controller's agent applies at its first start, such as the network settings chosen for the restored system.
- **preflight**—The checks the Manager makes on a restore or recovery plan to confirm that the chosen backup and target can succeed: when it is planned, and again before it is approved. A restore is checked once more when it executes. Each check produces a gate. See Preflight Check Reference.

R

- **rebuild**—In a recovery plan, to clean up a domain controller's metadata and promote a new domain controller in its place, with no backup restored. The plan reinstalls the Windows roles and features the domain controller's last backup recorded before promoting it.
- **rebuilt on recovery**—The protection verdict of a domain controller that is not a backup target: no enabled policy covers it and nothing of it is in the vault. A recovery rebuilds it by promotion, so the verdict is information and raises no alert. See Backup Targets and Rebuilt Domain Controllers.

- **recovery agent**—The program that runs inside the recovery image. It reads the backup from the vault, restores the whole server, and reports progress to the Manager. It is not installed or enrolled separately. See Recovery Images.
- **recovery code**—One of ten one-time codes issued when an operator turns on multi-factor authentication. Each signs in once in place of an authenticator code, for when the phone is not at hand. See Multi-Factor Authentication.
- **recovery environment**—The Windows PE system a machine runs from the recovery image during a full system recovery.
- **recovery ISO**—A bootable ISO the storage agent builds for one restore run onto a new virtual machine. It carries the recovery image and the run's single-use enrollment token, and nothing else that grants access. See Full System Recovery onto a New VM.
- **recovery image**—A Windows PE image built from your own Windows Server media, containing the recovery agent. One is built for each operating system variant your domain controllers run. An image is **proven** once a restore that started from it completes, including a test restore. See Recovery Images.
- **recovery network**—The internal Hyper-V switch named CygnaDR Recovery, with Windows NAT behind it, that the Isolated Recovery Host agent creates. Test restore virtual machines reach the Manager and the storage machine through it while they restore, and nothing on your network can reach them.
- **recovery passphrase**—The passphrase, set at the first sign-in, that protects an escrow copy of the Manager's CA key. It also opens the Manager's configuration backups. You need it to recover the Manager on a new machine. See Recovering the Manager.
- **recovery plan**—An ordered, gated set of steps that recovers one domain or a whole forest. See also **restore run**. See How Recovery Works.
- **recovery report**—The report a domain or forest recovery plan produces at its last step and emails to its notification addresses: what happened to each domain controller, the manual follow-ups left, and the decisions made on the way. See After a Recovery.
- **repository**—The vault seen as storage: its capacity, how much history it holds per domain controller, retention, and hash verification coverage. The console's **Repository** screen shows it. See Repository.
- **restore point**—One stored backup of one domain controller that can be restored, with its manifest. See Restore Points.
- **restore verified**—The verdict of a test restore whose backup restored, started isolated, and passed every required check. The other verdict is **restore failed**, which names the step and the reason. A test restore that could not do its part leaves the backup unverified. See Test Restores.
- **restore run**—The restore of one domain controller, planned, approved, and executed as a unit. A domain or forest recovery plan runs restore runs as some of its steps.

- **retention floor**—An age within which no version is deleted to make room for an incoming backup. It does not limit the version-count or age ceilings; see Retention.
- **retention mode**—How retention chooses which backups to keep: **newest** or **thinned**. See Retention.
- **retired**—Recorded as removed from the directory, either by discovery or by an operator. Retired domain controllers are not counted in coverage. See Excluding and Retiring Objects.
- **role**—A named set of capabilities assigned to an operator: Viewer / Auditor, Backup Operator, Restore Operator, or System Admin. An operator's capabilities are the combination of their roles. See Managing Operators and Roles.

S

- **staging**—The folder on the storage machine that domain controllers write a backup into, through a share and with a single-use account, before the backup is verified and moved into the vault. See How Backup Works.
- **staging share**—The SMB share over the staging folder through which domain controllers write backups, each job with its own local account. The storage agent watches it and reports any other use as an unexpected SMB connection. See Vault Protection and Integrity.
- **start time**—The earliest time of day, in the Manager's local time, at which a backup policy's backups are due. See Managing Backup Policies.
- **storage agent**—The service on the storage machine that owns the staging folder and the vault, commits and verifies backups, applies retention, and builds recovery images. See Components and Architecture.
- **storage machine**—The machine, not joined to the domain, that holds the vault and runs the storage agent. An installation has one: the first storage agent to enroll. A storage agent that enrolls after it is a candidate and is given no work until the storage machine is replaced with it. A domain controller can never be the storage machine. See Setting Up the Storage Machine.
- **support package**—One zip file with the evidence for a support case: the Manager's log, a copy of the Manager database with every credential removed, host and configuration facts, and the logs of every connected agent. It is customer confidential and cannot restore a Manager. See Logs and Diagnostics.
- **survey**—A request to the target of a planned restore to describe itself again, so that the preflight checks reflect any change made to it since planning.

T

- **test restore**—A restore of one backup onto a new virtual machine on an Isolated Recovery Host, started where it cannot reach your network and checked as a working domain controller, then deleted. Nothing in production changes. See Test Restores.
- **thinned**—A retention mode that keeps each domain controller's newest and oldest versions within the retention floor and deletes from between them, so history is dense near today and sparse further back. See Retention.

U

- **unattended plan**—A domain or forest recovery plan that does not stop for a person at points of no return, attestations, the backup checkpoint, or restore confirmations. The operator who executes it confirms this and accepts each pause in their own name. It still halts on a failed step, waits for a missing credential, and stops at every trust reset that must be done by hand. See Recovering a Forest.

V

- **vault**—The folder on the storage machine that holds committed backups. It has no network share, and no credential that exists between a backup and a restore can reach it over the network. See Vault Protection and Integrity.
- **version identifier**—The identifier Windows Server Backup gives a backup, shown as recorded (for example `10/05/2026-02:30`, in the domain controller's date format). Restores are addressed by it.

Index

A

Accounts 36
Active Directory
  supported versions 18
Administration 426
Agents
  automatic updates 496, 638
  cdr command-line tool 634
  connectivity 600
  console screen 312
  deleting a record 636
  installing from the console 227
  maintaining 492
  overview 11
  packages 641
  re-pointing 492
  restarting 639
  uninstalling 640
  updating 492, 637
Agents screen 312
Alarms 509
  Alerts screen 299
Alerts
  cdr command-line tool 690
  email 527
  email settings 459
  email status 301, 691
  reference 509
Alerts screen 299
Architecture 11
Attestation 161
  recovery steps 685
Audit policy
  storage machine 526
Audit trail 528
  cdr command-line tool 690
  console screen 307
  exporting 310, 533, 694
Audit Trail screen 307
Auditing 528
Authenticator app 436
Authoritative restore 157, 366
  Microsoft guidance 152
  scopes 367
Automatic agent updates 496

B

Back up now 349
Backup jobs
  cancelling 352, 654
  errors 595
  listing 653
  monitoring 352
Backup Jobs screen 255
Backup policies
  cdr command-line tool 647
  first 93
  managing 342
  screen 239

Backup Policies screen 239
Backup storage
sizing 46
Backup targets 201
Backup tasks 341
Backup types 110
Backups
cdr command-line tool 647
first backup 93
how backup works 108
job history 255
not committed 656
restore points 112
running now 349
starting now 652
tasks 341
types 110
verifying 129, 355
which domain controllers 44
BitLocker
vault volume 116
Break-glass account 427
creating 66

C

CA fingerprint 41
verifying 70
ca recover 550
ca rekey 556
Cancelling backups 352
Candidate machines 153, 384
listing 679
cdr agent deploy 79
cdr command-line tool 613
agent auto-update 638
agent commands 634
agent deploy 642
agent forget 636
agent list 635
agent package download 641
agent restart 639
agent uninstall 640
agent update 637
alert list 690
alert notify status 691
alert notify test 692
audit export 694
audit list 692
backup commands 647
backup list 650
backup show 652
backup start 652
cleanup commands 706
cleanup dc 708
cleanup domain 709
cleanup forget 712
cleanup list 708
cleanup retire 710
cleanup trust 710
cleanup unretire 711

config backup 699
config backup run 700
config set 697
config show 696
configuration commands 696
dc list 627
dc rebuild 628
domain list 631
exclusion add 704
exclusion commands 696
exclusion list 703
exclusion remove 705
forest list 632
inventory commands 627
jobs cancel 654
jobs list 653
license apply 702
license commands 696
license preview 701
license remove 702
license show 700
login 622
logout 623
monitoring commands 690
operator list 624
operator remove 625
operator restore 626
os-media list 663
policy list 647
policy remove 650
policy set 648
recovery-host approve 642
recovery-host drain 642
recovery-host list 641
recovery-host undrain 642
recovery abort 687
recovery ack 685
recovery approve 684
recovery attended 688
recovery attest 685
recovery candidates 679
recovery commands 677
recovery execute 684
recovery list 678
recovery notify 688
recovery override-wait 687
recovery plan 679
recovery report 689
recovery retry 686
recovery secrets 683
recovery show 682
report show 695
repository changes 657
repository changes review 658
repository estimate 659
repository status 655
repository uncommitted 656
request 662
restore abandon 676
restore approve 672

restore cancel 675
restore commands 664
restore discard 676
restore list 665
restore plan 667
restore reset-krbtgt 674
restore run 673
restore show 666
restore survey 672
restore validate 677
restore verify 355, 671
signing in 621
site list 633
storage commands 655
storage replace 661
support commands 714
support list 716
support package 714
trust commands 634
trust list 645
trust pin 646
trust remove 646
whoami 624
cdr dc rebuild 384, 389
cdr repository estimate 46
Certificates 41
agent 492
Manager CA 70
Colocated roles 178
Command line
cdr 613
Components 11
Configuration
settings reference 444
Configuration backups 545
card 488
cdr command-line tool 699
importing 550
Configuration screen 335
search 336
Console
Agents 312
Alerts 299
appearance 185
Audit Trail 307
Backup Jobs 255
Configuration 335
first sign-in 66
navigation 181
Operators & Roles 327
overview 181
Protection Summary 303
Recovery Plans 273
Repository 267
Restore Points 262
Conventions 19
Coverage 135
domain 213
site 221

Credentials
recovery 167, 683
required 36
Cygna Auditor 21
Cygna Forest Recovery
overview 10

D

Dashboard 195
Database snapshots 543
restoring 550
DC agent
after a DC state restore 610
deploying 79
updating 102, 312
DC agents
push install 642
DC state backup 110
DC state recovery 153
procedure 366
scopes 367
Defaults 444
Deployment
planning 23
Diagnostics 533
Disaster Readiness 195
Domain controllers
backup targets 201
choosing which to back up 44
metadata cleanup 708
removed from the directory 359
Domain Controllers screen 227
Domain recovery
cdr command-line tool 677
isolation 173
procedure 393
steps 144
Domains
removing an orphaned domain 709
Domains screen 213

E

Email
alerts 301, 459, 527
Enrollment
troubleshooting 600
Enrollment token 41, 62
storing 543
Error 1714 610
Error codes
backup 595
Event Log 533
Exclusions 359
about 703
cdr command-line tool 696
Exporting
audit trail 310, 533

F

Firewall rules 33
Forest recovery
cdr command-line tool 677
isolation 173

Microsoft guidance 149
phases 144
procedure 404
Forests screen 205
Full system backup 110
Full system recovery 153
in place 376
onto a candidate 384
onto a new VM 389
recovery images 125
requirements 167

G

Gates 563
Glossary 718
Grace period 482
Guide
using 19

H

Hardware requirements 27
High contrast 185
Hyper-V 96

I

Installation
DC agents 79
Manager 62
overview 52
quick install 54
troubleshooting 607
Installation media 246
Installer
messages 607
Integrity verification 116
Inventory
cdr command-line tool 627
ISO files 89
Isolated Recovery Host
approving 504, 642
cdr command-line tool 641
draining 504, 642
moving 504
setting up 96
test restores 129
Windows updates 101
Isolated Recovery Host agent
installing 96
Isolation
recovery targets 173

K

krbtgt
DC state recovery 368
reset after DC state recovery 674
reset after rollback 375
second reset 687
krbtgt reset 404

L

License key 482
applying 69
Licensing 482
cdr command-line tool 700

expiry warnings 482
Logs 533

M

Manager
administering 426
compromised 556
configuration backups 545
day-one checklist 543
installing 62
overview 11
protecting 540
re-keying 556
recovering 550
recovery 540
Manager CA 41
trusting 70
Manager recovery card 488
Manifest 112
Manifests
viewing 652
Metadata cleanup 706
MFA 436
Microsoft
authoritative restore 152
forest recovery guide 149
Monitoring
probe 690
msiexec 607
Multi-factor authentication 436
turning on 184

N

Network settings
full system recovery 376
Non-authoritative restore 157
Notifications
recovery plans 688

O

Operating systems
supported 27
Operators 427
activity 528
cdr command-line tool 624
console screen 327
removing 625
restoring 626
Operators & Roles screen 327
OS media
cdr command-line tool 655
coverage 663
recovery images 125
setting up 89
OS Media screen 246
Overview 10

P

Passwords
operator 427
Permissions 36
Planning 23
backup set 44

Point of no return 161
acknowledging 685
Points of no return 404
Ports 33
Post-recovery tasks 421
Preflight
checks 563
Preflight checks
cdr command-line tool 667
Problems
diagnosing 561
Protection
backup targets 201
domain controller 227
Protection Summary screen 303
Protocols 33
Push installation
failures 600

Q

Quick install 54

R

Ransomware recovery 158
Read-only domain controllers 18
Readiness 135
dashboard 195
forest 205
Rebuild 153
virtual machine from a backup 628
Rebuilt on recovery 201
Recovery
cancel, discard, abandon, abort 161
choosing an approach 158
follow-up tasks 421
how recovery works 141
limitations 178
procedures 364
Recovery agent 125
Recovery codes 436
Recovery for Active Directory 21
Recovery for Entra ID 21
Recovery images 125
proven 126
retirement 126, 252
setting up 89
status 246
Recovery lifecycle 161
Recovery media 389
Recovery network 173
Recovery passphrase 540
setting 66
Recovery plans 141
cdr command-line tool 677
console screen 273
domain 393
forest 404
retrying a step 686
Recovery Plans screen 273
Recovery procedures 364

Recovery report 178
cdr command-line tool 689
Recovery requirements 167
Recovery scopes 144
Related products 21
Reports
backup outcomes 303, 695
cdr command-line tool 690
CSV export 303
Repository
capacity 655
estimating size 46
Repository screen 267
Requests 504, 662
Requirements
system 27
Restore
full system recovery 376
preflight 563
single domain controller 366
Restore methods 153
Restore points 112
console screen 262
listing 650
Restore Points screen 262
Restore runs 141
cdr command-line tool 664
console screen 273
Restore validation 677
Restrict Administrators 116
Retention 121
Repository screen 267
Retention floor 121
Retention mode 121
Retired domain controllers 227
Retired objects 359
recording 710
Retirement
cdr command-line tool 706
Roles 427
console screen 327

S

Schedules 239
backup 342
Scripting
cdr command-line tool 613
Secure Boot 101
Security breach flag 158
Service accounts 36
Sessions
cdr command-line tool 621
Setting up 52
Settings 444
cdr command-line tool 696
Configuration screen 335
searching 336
Signing in
cdr command-line tool 621
console 181
Single-DC recovery 144

Single-DC restore
DC state recovery 366
Site links 221
Sites screen 221
Sizing 46
backup storage 659
Staging 108
Staging share 73
monitoring 116
Storage agent
installing 73
Storage machine
audit policy 526
manager-backups folder 545
moving 504
replacing 504, 661
setting up 73
Subnets 221
Support
collecting information 533
support packages 536
Support packages 536
cdr command-line tool 714
collecting 562
creating in the console 486
Supported directories 18
System requirements 27

T

Terminology 718
Test restores 129
cdr command-line tool 671
preflight 584
running 355
Themes 185
Tombstone lifetime 135
Domains screen 213
Topology 23
Troubleshooting 561
backups 595
Trusts
external 178
Forests screen 205
listing 632
removing 710

U

Unattended recovery plans
taking back 688
Uninstalling 102
Upgrading 102

V

Vault
backup flow 108
capacity 267
cdr command-line tool 655
change detection 116, 657
encryption 116
hash verification 116
protection 116
retention 121

Verdicts 135

Verified restore 129

Version identifier 112

W

Windows Server Backup 27