Red Hat Enterprise Linux 8

Preparing for disaster recovery with Identity Management

Documentation for mitigating disasters affecting an Identity Management deployment

Last Updated: 2020-11-23

Red Hat Enterprise Linux 8 Preparing for disaster recovery with Identity Management

Documentation for mitigating disasters affecting an Identity Management deployment

Legal Notice

Abstract

This document describes common disaster scenarios that threaten an IdM deployment, along with methods to mitigate those situations through replication, Virtual Machine snapshots, and backups.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Please let us know how we could make it better. To do so:

- For simple comments on specific passages:
  1. Make sure you are viewing the documentation in the *Multi-page HTML* format. In addition, ensure you see the **Feedback** button in the upper right corner of the document.
  2. Use your mouse cursor to highlight the part of text that you want to comment on.
  3. Click the **Add Feedback** pop-up that appears below the highlighted text.
  4. Follow the displayed instructions.
- For submitting more complex feedback, create a Bugzilla ticket:
  1. Go to the Bugzilla website.
  2. As the Component, use **Documentation**.
  3. Fill in the **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
  4. Click **Submit Bug**.

CHAPTER 1. DISASTER RECOVERY TOOLS IN IDM

A good disaster recovery strategy combines the following tools in order to recover from a disaster as soon as possible with minimal data loss:

Replication

Replication copies database contents between IdM servers. If an IdM server fails, you can replace the lost server by creating a new replica based on one of the remaining servers.

Virtual machine (VM) snapshots

A snapshot is a view of a VM's operating system and applications on any or all available disks at a given point in time. After taking a VM snapshot, you can use it to return a VM and its IdM data to a previous state.

IdM backups

The **ipa-backup** utility allows you to take a backup of an IdM server's configuration files and its data. You can later use a backup to restore an IdM server to a previous state.

CHAPTER 2. DISASTER SCENARIOS IN IDM

There are two main classes of disaster scenarios: *server loss* and *data loss*.

Table 2.1. Server loss vs. data loss

Disaster type	Example causes	How to prepare
Server loss: The IdM deployment loses one or several servers.	• Hardware malfunction	• Chapter 3, *Preparing for server loss with replication*
Data loss: IdM data is unexpectedly modified on a server, and the change is propagated to other servers.	• A user accidentally deletes data • A software bug modifies data	• Chapter 4, *Preparing for data loss with VM snapshots* • Chapter 5, *Preparing for data loss with IdM backups*

CHAPTER 3. PREPARING FOR SERVER LOSS WITH REPLICATION

Follow these guidelines to establish a replication topology that will allow you to respond to losing a server.

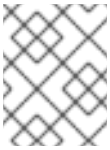
3.1. CONNECTING THE REPLICAS IN A TOPOLOGY

Connect each replica to at least two other replicas

Configuring additional replication agreements ensures that information is replicated not just between the initial replica and the master server, but between other replicas as well.

Connect a replica to a maximum of four other replicas (not a hard requirement)

A large number of replication agreements per server does not add significant benefits. A receiving replica can only be updated by one other replica at a time and meanwhile, the other replication agreements are idle. More than four replication agreements per replica typically means a waste of resources.



NOTE

This recommendation applies to both certificate replication and domain replication agreements.

There are two exceptions to the limit of four replication agreements per replica:

- You want failover paths if certain replicas are not online or responding.
- In larger deployments, you want additional direct links between specific nodes.

Configuring a high number of replication agreements can have a negative impact on overall performance: when multiple replication agreements in the topology are sending updates, certain replicas can experience a high contention on the changelog database file between incoming updates and the outgoing updates.

If you decide to use more replication agreements per replica, ensure that you do not experience replication issues and latency. However, note that large distances and high numbers of intermediate nodes can also cause latency problems.

Connect the replicas in a data center with each other

This ensures domain replication within the data center.

Connect each data center to at least two other data centers

This ensures domain replication between data centers.

Connect data centers using at least a pair of replication agreements

If data centers A and B have a replication agreement from A1 to B1, having a replication agreement from A2 to B2 ensures that if one of the servers is down, the replication can continue between the two data centers.

3.2. REPLICA TOPOLOGY EXAMPLES

The figures below show examples of Identity Management (IdM) topologies based on the guidelines for creating a reliable topology.

Figure 3.1, “Replica Topology Example 1” shows four data centers, each with four servers. The servers are connected with replication agreements.

Figure 3.1. Replica Topology Example 1

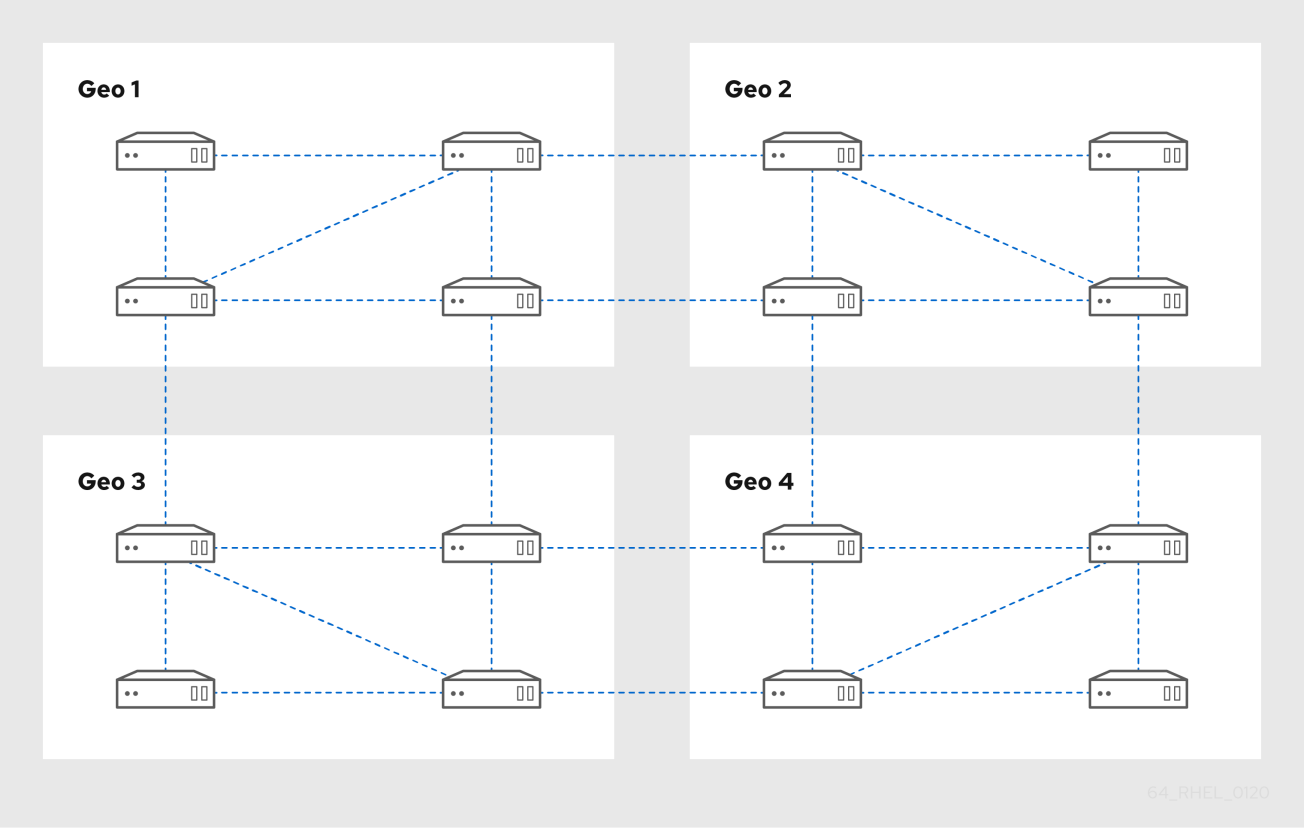
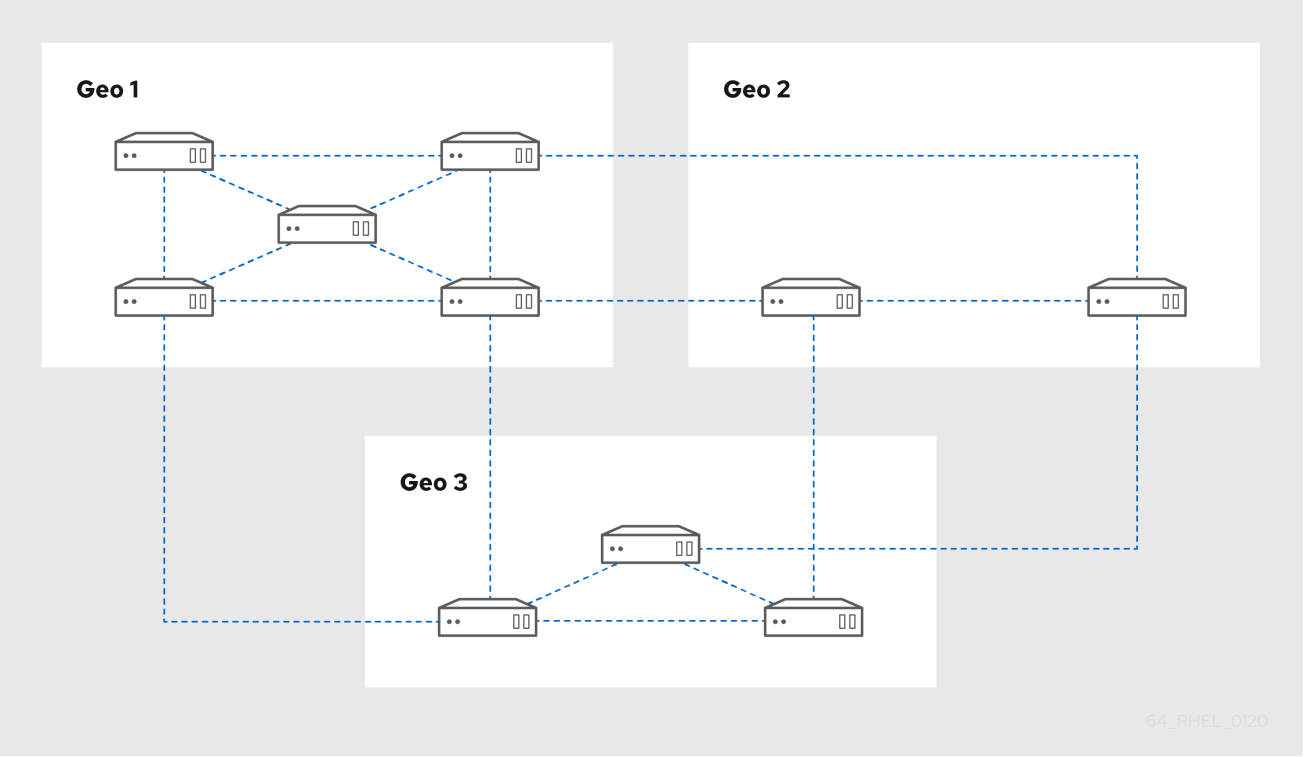


Figure 3.2, “Replica Topology Example 2” shows three data centers, each with a different number of servers. The servers are connected with replication agreements.

Figure 3.2. Replica Topology Example 2



3.3. PROTECTING IDM CA DATA

If your deployment contains the integrated IdM Certificate Authority (CA), install several CA replicas so you can create additional CA replicas if one is lost.

Procedure

1. Configure three or more replicas to provide CA services.

   a. To install a new replica with CA services, run **ipa-replica-install** with the **--setup-ca** option.

   ```
   [root@server ~]# ipa-replica-install --setup-ca
   ```

   b. To install CA services on a preexisting replica, run **ipa-ca-install**.

   ```
   [root@replica ~]# ipa-ca-install
   ```

2. Create CA replication agreements between your CA replicas.

```
[root@careplica1 ~]# ipa topologysegment-add
Suffix name: ca
Left node: careplica1.example.com
Right node: careplica2.example.com
Segment name [careplica1.example.com-to-careplica2.example.com]: new_segment
---------------------------
Added segment "new_segment"
---------------------------
  Segment name: new_segment
  Left node: careplica1.example.com
  Right node: careplica2.example.com
  Connectivity: both
```

WARNING

If only one server provides CA services and it is damaged, the entire environment will be lost. If you use the IdM CA, Red Hat **strongly recommends** having three or more replicas with CA services installed, with CA replication agreements between them.

Additional resources

- For more information on CA options in IdM, see Planning your CA services.
- For more information on installing IdM replicas, see Installing an IdM replica .

3.4. ADDITIONAL RESOURCES

- For additional information on replication, see Planning the replica topology.

CHAPTER 4. PREPARING FOR DATA LOSS WITH VM SNAPSHOTS

Virtual machine (VM) snapshots are an integral component of a data recovery strategy, since they preserve the full state of an IdM server:

- Operating system software and settings
- IdM software and settings
- IdM customer data

Preparing a VM snapshot of an IdM Certificate Authority (CA) replica allows you to rebuild an entire IdM deployment after a disaster.

WARNING

If your environment uses the integrated CA, a snapshot of a replica *without a CA* will not be sufficient for rebuilding a deployment, because certificate data will not be preserved.

Similarly, if your environment uses the IdM Key Recovery Authority (KRA), make sure you create snapshots of a KRA replica, or you may lose the storage key.

Red Hat recommends creating snapshots of a VM that has all of the IdM server roles installed which are in use in your deployment: CA, KRA, DNS.

Prerequisites

- A hypervisor capable of hosting RHEL VMs.

Procedure

1. Configure at least one **CA replica** in the deployment to run inside a VM.
   a. If IdM DNS or KRA are used in your environment, consider installing DNS and KRA services on this replica as well.
   b. Optionally, configure this VM replica as a hidden replica.
2. Periodically shutdown this VM, take a full snapshot of it, and bring it back online so it continues to receive replication updates. If the VM is a hidden replica, IdM Clients will not be disrupted during this procedure.

Additional resources

- For a list of certified hypervisors that have been tested and proven to run Red Hat Enterprise Linux as a guest, see Which hypervisors are certified to run Red Hat Enterprise Linux? .
- For more information on hidden replicas, see The hidden replica mode.

CHAPTER 5. PREPARING FOR DATA LOSS WITH IDM BACKUPS

IdM provides the **ipa-backup** utility to backup IdM data, and the **ipa-restore** utility to restore servers and data from those backups.

NOTE

Red Hat recommends running backups as often as necessary on a hidden replica with all server roles installed, especially the Certificate Authority (CA) role if the environment uses the integrated IdM CA.

5.1. IDM BACKUP TYPES

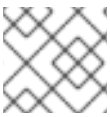
IdM provides two types of backups: a full-server backup, and a data-only backup.

Backup type	Backup contents	Performed Online or Offline	Suitable for
Full-server backup	• All server configuration files related to IdM • LDAP data in LDAP Data Interchange Format (LDIF)	Offline only. IdM services must be temporarily stopped.	Rebuilding an IdM deployment from scratch
Data-only backup	• LDAP data in LDAP Data Interchange Format (LDIF) • Replication Changelog	Online or Offline.	Restoring IdM data to a state in the past

5.2. NAMING CONVENTIONS FOR IDM BACKUP FILES

By default, IdM stores backups in the **/var/lib/ipa/backup/** directory, and the naming conventions for these subdirectories are:

- Full-server backup: **ipa-full-YEAR-MM-DD-HH-MM-SS** in GMT time
- Data-only backup: **ipa-data-YEAR-MM-DD-HH-MM-SS** in GMT time

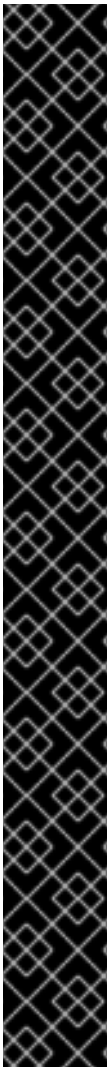


NOTE

Uninstalling an IdM server does not automatically remove any backup files.

5.3. CREATING A BACKUP

This section describes how to create a full-server and data-only backup in offline and online modes using the **ipa-backup** command.



IMPORTANT

- By default, **ipa-backup** runs in offline mode, which will stop all IdM services. The services will start automatically after the backup is finished.
- A full-server backup must **always** run with IdM services offline, but a data-only backup may be performed with services online.
- By default, backups are created on the file system containing the **/var/lib/ipa/backup/** directory. We recommend creating backups regularly on a file system separate from the production filesystem used by IdM, and archiving the backups to a fixed medium (tape or optical storage, for example).
- Consider performing backups on hidden replicas. IdM services can be shut down on hidden replicas without affecting IdM clients.
- Starting with RHEL 8.3.0, the **ipa-backup** utility checks if all of the services used in your IdM cluster, such as a Certificate Authority (CA), Domain Name System (DNS), and Key Recovery Agent (KRA), are installed on the server where you are running the backup. If the server does not have all these services installed, the **ipa-backup** utility exits with a warning, because backups taken on that host would not be sufficient for a full cluster restoration.
  For example, if your IdM deployment uses an integrated Certificate Authority (CA), a backup run on a non-CA replica will not capture CA data. Red Hat recommends verifying that the replica where you perform an **ipa-backup** has all of the IdM services used in the cluster installed.

  You can bypass the IdM server role check with the **ipa-backup --disable-role-check** command, but the resulting backup will not contain all the data necessary to restore IdM fully.

Examples of using the ipa-backup command

- To create a full-server backup in offline mode, use the **ipa-backup** utility without additional options.

```
[root@server ~]# ipa-backup
Preparing backup on server.example.com
Stopping IPA services
Backing up ipaca in EXAMPLE-COM to LDIF
Backing up userRoot in EXAMPLE-COM to LDIF
Backing up EXAMPLE-COM
Backing up files
Starting IPA service
Backed up to /var/lib/ipa/backup/ipa-full-2020-01-14-11-26-06
The ipa-backup command was successful
```

- To create an offline data-only backup, specify the **--data** option.

```
[root@server ~]# ipa-backup --data
```

- To create a full-server backup that includes IdM log files, use the **--logs** option.

```
[root@server ~]# ipa-backup --logs
```

- To create a data-only backup while IdM services are running, specify both **--data** and **--online** options.

```
[root@server ~]# ipa-backup --data --online
```

NOTE

If the backup fails due to insufficient space in the **/tmp** directory, use the **TMPDIR** environment variable to change the destination for temporary files created by the backup process:

```
[root@server ~]# TMPDIR=/new/location ipa-backup
```

For more details, see ipa-backup Command Fails to Finish .

Verification Steps

- The backup directory contains an archive with the backup.

```
[root@server ~]# ls /var/lib/ipa/backup/ipa-full-2020-01-14-11-26-06
header  ipa-full.tar
```

5.4. CREATING ENCRYPTED IDM BACKUPS

You can create encrypted backups using GNU Privacy Guard (GPG) encryption. To create encrypted IdM backups, you will first need to create a GPG2 key.

5.4.1. Creating a GPG2 key for encrypting IdM backups

The following procedure describes how to generate a GPG2 key for the **ipa-backup** utility.

Procedure

1. Install and configure the **pinentry** utility.

```
[root@server ~]# dnf install pinentry
[root@server ~]# mkdir ~/.gnupg -m 700
[root@server ~]# echo "pinentry-program /usr/bin/pinentry-curses" >> ~/.gnupg/gpg-agent.conf
```

2. Create a **key-input** file used for generating a GPG keypair with your preferred details. For example:

```
[root@server ~]# cat >key-input <<EOF
%echo Generating a standard key
Key-Type: RSA
Key-Length: 2048
Name-Real: IPA Backup
Name-Comment: IPA Backup
Name-Email: root@example.com
```

```
Expire-Date: 0
%commit
%echo Finished creating standard key
EOF
```

3. By default, GPG2 stores its keyring in the **~/.gnupg** file. To use a custom keyring location, set the **GNUPGHOME** environment variable to a directory that is only accessible by root.

```
[root@server ~]# export GNUPGHOME=/root/backup

[root@server ~]# mkdir -p $GNUPGHOME -m 700
```

4. Begin generating a new GPG2 key based on the contents of **key-input**.

```
[root@server ~]# gpg2 --batch --gen-key key-input
```

a. Enter a passphrase to protect the GPG2 key.

```
┌──────────────────────────────────────────────────────┐
│ Please enter the passphrase to                       │
│ protect your new key                                 │
│                                                      │
│ Passphrase: SecretPassphrase42                       │
│                                                      │
│  <OK>                          <Cancel>              │
└──────────────────────────────────────────────────────┘
```

b. Confirm the correct passphrase by entering it again.

```
┌──────────────────────────────────────────────────────┐
│ Please re-enter this passphrase                      │
│                                                      │
│ Passphrase: SecretPassphrase42                       │
│                                                      │
│  <OK>                          <Cancel>              │
└──────────────────────────────────────────────────────┘
```

c. The new GPG2 key is now created.

```
gpg: keybox '/root/backup/pubring.kbx' created
gpg: Generating a standard key
gpg: /root/backup/trustdb.gpg: trustdb created
gpg: key BF28FFA302EF4557 marked as ultimately trusted
gpg: directory '/root/backup/openpgp-revocs.d' created
gpg: revocation certificate stored as '/root/backup/openpgp-
revocs.d/8F6FCF10C80359D5A05AED67BF28FFA302EF4557.rev'
gpg: Finished creating standard key
```

Verification Steps

- List the GPG keys on the server.

```
[root@server ~]# gpg2 --list-secret-keys
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   1  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 1u
/root/backup/pubring.kbx
------------------------
sec   rsa2048 2020-01-13 [SCEA]
      8F6FCF10C80359D5A05AED67BF28FFA302EF4557
uid           [ultimate] IPA Backup (IPA Backup) <root@example.com>
```

Additional resources

- For more information on GPG encryption and its uses, see the GNU Privacy Guard website.

5.4.2. Creating a GPG2-encrypted IdM backup

The following procedure creates an IdM backup and encrypts it using a GPG2 key.

Prerequisites

- You have created a GPG2 key. See Creating a GPG2 key for encrypting IdM backups .

Procedure

- Create a GPG-encrypted backup by specifying the **--gpg** option.

```
[root@server ~]# ipa-backup --gpg
Preparing backup on server.example.com
Stopping IPA services
Backing up ipaca in EXAMPLE-COM to LDIF
Backing up userRoot in EXAMPLE-COM to LDIF
Backing up EXAMPLE-COM
Backing up files
Starting IPA service
Encrypting /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00/ipa-full.tar
Backed up to /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00
The ipa-backup command was successful
```

Verification Steps

- Ensure that the backup directory contains an encrypted archive with a **.gpg** file extension.

```
[root@server ~]# ls /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00
header  ipa-full.tar.gpg
```

Additional resources

- For general information on creating a backup, see Creating a backup .

5.5. ADDITIONAL RESOURCES

- For more information on backing up and restoring IdM, see Backing up and restoring IdM.