Red Hat Enterprise Linux 10

Performing disaster recovery with Identity Management

Recovering IdM after a server or data loss

Last Updated: 2025-05-22

Red Hat Enterprise Linux 10 Performing disaster recovery with Identity Management

Recovering IdM after a server or data loss

Legal Notice

Abstract

Server and data loss scenarios, for example due to a hardware failure, are the highest risks in IT environments. In case of such an event in a Red Hat Identity Management (IdM) environment, the recovery process depends on the type of problem, the IdM topology, and the actions that have been taken to mitigate such situations. For example, you can recover single and multiple servers in an IdM replication topology, and you can recover data by using IdM backups and snapshots. During or after the recovery, it can be necessary to adjust client settings, such as DNS servers and the Kerberos configuration.

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. Let us know how we can improve it.

Submitting feedback through Jira (account required)

1. Log in to the Jira website.
2. Click **Create** in the top navigation bar
3. Enter a descriptive title in the **Summary** field.
4. Enter your suggestion for improvement in the **Description** field. Include links to the relevant parts of the documentation.
5. Click **Create** at the bottom of the dialogue.

CHAPTER 1. DISASTER SCENARIOS IN IDM

Prepare and respond to various disaster scenarios in Identity Management (IdM) systems that affect servers, data, or entire infrastructures.

Table 1.1. Disaster scenarios in IdM

Disaster type	Example causes	How to prepare	How to respond
Server loss: The IdM deployment loses one or several servers.	• Hardware malfunction	• Preparing for server loss with replication	• Recovering a single server 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	• Preparing for data loss with VM snapshots • Preparing for data loss with IdM backups	• Recovering from data loss with VM snapshots • Recovering from data loss with IdM backups • Managing data loss
Total infrastructure loss: All IdM servers or Certificate Authority (CA) replicas are lost with no VM snapshots or data backups available.	• Lack of off-site backups or redundancy prevents recovery after a failure or disaster.	• Preparing for data loss with VM snapshots	This situation is a total loss.

WARNING

A total loss scenario occurs when all Certificate Authority (CA) replicas or all IdM servers are lost, and no virtual machine (VM) snapshots or backups are available for recovery. Without CA replicas, the IdM environment cannot deploy additional replicas or rebuild itself, making recovery impossible. To avoid such scenarios, ensure backups are stored off-site, maintain multiple geographically redundant CA replicas, and connect each replica to at least two others.

CHAPTER 2. RECOVERING A SINGLE SERVER WITH REPLICATION

If a single server is severely disrupted or lost, having multiple replicas ensures you can create a replacement replica and quickly restore the former level of redundancy.

If your IdM topology contains an integrated Certificate Authority (CA), the steps for removing and replacing a damaged replica differ for the CA renewal server and other replicas.

2.1. RECOVERING FROM LOSING THE CA RENEWAL SERVER

If the Certificate Authority (CA) renewal server is lost, you must first promote another CA replica to fulfill the CA renewal server role, and then deploy a replacement CA replica.

Prerequisites

- Your deployment uses IdM's internal Certificate Authority (CA).
- Another Replica in the environment has CA services installed.

WARNING

An IdM deployment is unrecoverable if:

1. The CA renewal server has been lost.
2. No other server has a CA installed.
3. No backup of a replica with the CA role exists.
   It is critical to make backups from a replica with the CA role so certificate data is protected. For more information about creating and restoring from backups, see Backing up and restoring IdM.

Procedure

1. From another replica in your environment, promote another CA replica in the environment to act as the new CA renewal server. See Changing and resetting IdM CA renewal server.
2. From another replica in your environment, remove replication agreements to the lost CA renewal server. See Removing server from topology using the CLI.
3. Install a new CA Replica to replace the lost CA replica. See Installing an IdM replica with a CA .
4. Update DNS to reflect changes in the replica topology. If IdM DNS is used, DNS service records are updated automatically.
5. Verify IdM clients can reach IdM servers. See Adjusting IdM clients during recovery.

Verification

1. Test the Kerberos server on the new replica by successfully retrieving a Kerberos Ticket-Granting-Ticket as an IdM user.

```
[root@server ~]# kinit admin
Password for admin@EXAMPLE.COM:

[root@server ~]# klist
Ticket cache: KCM:0
Default principal: admin@EXAMPLE.COM

Valid starting       Expires              Service principal
10/31/2019 15:51:37  11/01/2019 15:51:02  HTTP/server.example.com@EXAMPLE.COM
10/31/2019 15:51:08  11/01/2019 15:51:02  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

2. Test the Directory Server and SSSD configuration by retrieving user information.

```
[root@server ~]# ipa user-show admin
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  Principal alias: admin@EXAMPLE.COM
  UID: 1965200000
  GID: 1965200000
  Account disabled: False
  Password: True
  Member of groups: admins, trust admins
  Kerberos keys available: True
```

3. Test the CA configuration with the **ipa cert-show** command.

```
[root@server ~]# ipa cert-show 1
  Issuing CA: ipa
  Certificate: MIIEgjCCAuqgAwIBAgIjoSIP...
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
  Issuer: CN=Certificate Authority,O=EXAMPLE.COM
  Not Before: Thu Oct 31 19:43:29 2019 UTC
  Not After: Mon Oct 31 19:43:29 2039 UTC
  Serial number: 1
  Serial number (hex): 0x1
  Revoked: False
```

Additional resources

- Using IdM CA renewal server

2.2. RECOVERING FROM LOSING A REGULAR REPLICA

To replace a replica that is not the Certificate Authority (CA) renewal server, remove the lost replica from the topology and install a new replica in its place.

Prerequisites

- The CA renewal server is operating properly. If the CA renewal server has been lost, see Recovering from losing the CA renewal server .

Procedure

1. Remove replication agreements to the lost server. See Uninstalling an IdM server .
2. Deploy a new replica with the corresponding services (CA, KRA, DNS). See Installing an IdM replica.
3. Update DNS to reflect changes in the replica topology. If IdM DNS is used, DNS service records are updated automatically.
4. Verify IdM clients can reach IdM servers. See Adjusting IdM clients during recovery .

Verification

1. Test the Kerberos server on the new replica by successfully retrieving a Kerberos Ticket-Granting-Ticket as an IdM user.

```
[root@newreplica ~]# kinit admin
Password for admin@EXAMPLE.COM:

[root@newreplica ~]# klist
Ticket cache: KCM:0
Default principal: admin@EXAMPLE.COM

Valid starting       Expires              Service principal
10/31/2019 15:51:37  11/01/2019 15:51:02  HTTP/server.example.com@EXAMPLE.COM
10/31/2019 15:51:08  11/01/2019 15:51:02  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

2. Test the Directory Server and SSSD configuration on the new replica by retrieving user information.

```
[root@newreplica ~]# ipa user-show admin
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  Principal alias: admin@EXAMPLE.COM
  UID: 1965200000
  GID: 1965200000
  Account disabled: False
  Password: True
  Member of groups: admins, trust admins
  Kerberos keys available: True
```

CHAPTER 3. RECOVERING MULTIPLE SERVERS WITH REPLICATION

If multiple servers are lost at the same time, determine if the environment can be rebuilt by seeing which one of the following five scenarios applies to your situation.

3.1. RECOVERING FROM LOSING MULTIPLE SERVERS IN A CA-LESS DEPLOYMENT

Servers in a CA-less deployment are all considered equal, so you can rebuild the environment by removing and replacing lost replicas in any order.

Prerequisites

- Your deployment uses an external Certificate Authority (CA).

Procedure

- See Recovering from losing a regular replica .

3.2. RECOVERING FROM LOSING MULTIPLE SERVERS WHEN THE CA RENEWAL SERVER IS UNHARMED

If the CA renewal server is intact, you can replace other servers in any order.

Prerequisites

- Your deployment uses the IdM internal Certificate Authority (CA).

Procedure

- See Recovering from losing a regular replica .

3.3. RECOVERING FROM LOSING THE CA RENEWAL SERVER AND OTHER SERVERS

If you lose the CA renewal server and other servers, promote another CA server to the CA renewal server role before replacing other replicas.

Prerequisites

- Your deployment uses the IdM internal Certificate Authority (CA).
- At least one CA replica is unharmed.

Procedure

1. Promote another CA replica to fulfill the CA renewal server role. See Recovering from losing the CA renewal server.
2. Replace all other lost replicas. See Recovering from losing a regular replica .

CHAPTER 4. RECOVERING FROM DATA LOSS WITH VM SNAPSHOTS

If a data loss event occurs, you can restore a Virtual Machine (VM) snapshot of a Certificate Authority (CA) replica to repair the lost data, or deploy a new environment from it.

4.1. RECOVERING FROM ONLY A VM SNAPSHOT

If a disaster affects all IdM servers, and only a snapshot of an IdM CA replica virtual machine (VM) is left, you can recreate your deployment by removing all references to the lost servers and installing new replicas.

Prerequisites

- You have prepared a VM snapshot of a CA replica VM. See Preparing for data loss with VM snapshots.

Procedure

1. Boot the desired snapshot of the CA replica VM.
2. Remove replication agreements to any lost replicas.

```
[root@server ~]# ipa server-del lost-server1.example.com
[root@server ~]# ipa server-del lost-server2.example.com
...
```

3. Install a second CA replica. See Installing an IdM replica .
4. The VM CA replica is now the CA renewal server. Red Hat recommends promoting another CA replica in the environment to act as the CA renewal server. See Changing and resetting IdM CA renewal server.
5. Recreate the desired replica topology by deploying additional replicas with the desired services (CA, DNS). See Installing an IdM replica
6. Update DNS to reflect the new replica topology. If IdM DNS is used, DNS service records are updated automatically.
7. Verify that IdM clients can reach the IdM servers. See Adjusting IdM Clients during recovery .

Verification

1. Test the Kerberos server on every replica by successfully retrieving a Kerberos ticket-granting ticket as an IdM user.

```
[root@server ~]# kinit admin
Password for admin@EXAMPLE.COM:

[root@server ~]# klist
Ticket cache: KCM:0
Default principal: admin@EXAMPLE.COM
```

```
Valid starting       Expires              Service principal
10/31/2019 15:51:37  11/01/2019 15:51:02  HTTP/server.example.com@EXAMPLE.COM
10/31/2019 15:51:08  11/01/2019 15:51:02  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

2. Test the Directory Server and SSSD configuration on every replica by retrieving user information.

```
[root@server ~]# ipa user-show admin
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  Principal alias: admin@EXAMPLE.COM
  UID: 1965200000
  GID: 1965200000
  Account disabled: False
  Password: True
  Member of groups: admins, trust admins
  Kerberos keys available: True
```

3. Test the CA server on every CA replica with the **ipa cert-show** command.

```
[root@server ~]# ipa cert-show 1
  Issuing CA: ipa
  Certificate: MIIEgjCCAuqgAwIBAgIjoSIP...
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
  Issuer: CN=Certificate Authority,O=EXAMPLE.COM
  Not Before: Thu Oct 31 19:43:29 2019 UTC
  Not After: Mon Oct 31 19:43:29 2039 UTC
  Serial number: 1
  Serial number (hex): 0x1
  Revoked: False
```

Additional resources

- Planning the replica topology

4.2. RECOVERING FROM A VM SNAPSHOT AMONG A PARTIALLY-WORKING ENVIRONMENT

If a disaster affects some IdM servers while others are still operating properly, you may want to restore the deployment to the state captured in a Virtual Machine (VM) snapshot. For example, if all Certificate Authority (CA) Replicas are lost while other replicas are still in production, you will need to bring a CA Replica back into the environment.

In this scenario, remove references to the lost replicas, restore the CA replica from the snapshot, verify replication, and deploy new replicas.

Prerequisites

- You have prepared a VM snapshot of a CA replica VM. See Preparing for data loss with VM snapshots.

Procedure

1. Remove all replication agreements to the lost servers. See Uninstalling an IdM server .
2. Boot the desired snapshot of the CA replica VM.
3. Remove any replication agreements between the restored server and any lost servers.

```
[root@restored-CA-replica ~]# ipa server-del lost-server1.example.com
[root@restored-CA-replica ~]# ipa server-del lost-server2.example.com
...
```

4. If the restored server does not have replication agreements to any of the servers still in production, connect the restored server with one of the other servers to update the restored server.

```
[root@restored-CA-replica ~]# ipa topologysegment-add
Suffix name: domain
Left node: restored-CA-replica.example.com
Right node: server3.example.com
Segment name [restored-CA-replica.com-to-server3.example.com]: new_segment
---------------------------
Added segment "new_segment"
---------------------------
  Segment name: new_segment
  Left node: restored-CA-replica.example.com
  Right node: server3.example.com
  Connectivity: both
```

5. Review Directory Server error logs at **/var/log/dirsrv/slapd-YOUR-INSTANCE/errors** to see if the CA replica from the snapshot correctly synchronizes with the remaining IdM servers.
6. If replication on the restored server fails because its database is too outdated, reinitialize the restored server.

```
[root@restored-CA-replica ~]# ipa-replica-manage re-initialize --from
server2.example.com
```

7. If the database on the restored server is correctly synchronized, continue by deploying additional replicas with the desired services (CA, DNS) according to Installing an IdM replica .

Verification

1. Test the Kerberos server on every replica by successfully retrieving a Kerberos ticket-granting ticket as an IdM user.

```
[root@server ~]# kinit admin
Password for admin@EXAMPLE.COM:

[root@server ~]# klist
Ticket cache: KCM:0
Default principal: admin@EXAMPLE.COM
```

```
Valid starting       Expires              Service principal
10/31/2019 15:51:37  11/01/2019 15:51:02  HTTP/server.example.com@EXAMPLE.COM
10/31/2019 15:51:08  11/01/2019 15:51:02  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

2. Test the Directory Server and SSSD configuration on every replica by retrieving user information.

```
[root@server ~]# ipa user-show admin
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  Principal alias: admin@EXAMPLE.COM
  UID: 1965200000
  GID: 1965200000
  Account disabled: False
  Password: True
  Member of groups: admins, trust admins
  Kerberos keys available: True
```

3. Test the CA server on every CA replica with the **ipa cert-show** command.

```
[root@server ~]# ipa cert-show 1
  Issuing CA: ipa
  Certificate: MIIEgjCCAuqgAwIBAgIjoSIP...
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
  Issuer: CN=Certificate Authority,O=EXAMPLE.COM
  Not Before: Thu Oct 31 19:43:29 2019 UTC
  Not After: Mon Oct 31 19:43:29 2039 UTC
  Serial number: 1
  Serial number (hex): 0x1
  Revoked: False
```

Additional resources

- Recovering from a VM snapshot to establish a new IdM environment

4.3. RECOVERING FROM A VM SNAPSHOT TO ESTABLISH A NEW IDM ENVIRONMENT

If the Certificate Authority (CA) replica from a restored Virtual Machine (VM) snapshot is unable to replicate with other servers, create a new IdM environment from the VM snapshot.

To establish a new IdM environment, isolate the VM server, create additional replicas from it, and switch IdM clients to the new environment.

Prerequisites

- You have prepared a VM snapshot of a CA replica VM. See Preparing for data loss with VM snapshots.

Procedure

1. Boot the desired snapshot of the CA replica VM.

2. Isolate the restored server from the rest of the current deployment by removing all of its replication topology segments.

   a. First, display all **domain** replication topology segments.

```
[root@restored-CA-replica ~]# ipa topologysegment-find
Suffix name: domain
------------------
8 segments matched
------------------
  Segment name: new_segment
  Left node: restored-CA-replica.example.com
  Right node: server2.example.com
  Connectivity: both

...

----------------------------
Number of entries returned 8
----------------------------
```

   b. Next, delete every **domain** topology segment involving the restored server.

```
[root@restored-CA-replica ~]# ipa topologysegment-del
Suffix name: domain
Segment name: new_segment
----------------------------
Deleted segment "new_segment"
----------------------------
```

   c. Finally, perform the same actions with any **ca** topology segments.

```
[root@restored-CA-replica ~]# ipa topologysegment-find
Suffix name: ca
------------------
1 segments matched
------------------
  Segment name: ca_segment
  Left node: restored-CA-replica.example.com
  Right node: server4.example.com
  Connectivity: both
----------------------------
Number of entries returned 1
----------------------------

[root@restored-CA-replica ~]# ipa topologysegment-del
Suffix name: ca
Segment name: ca_segment
----------------------------
Deleted segment "ca_segment"
----------------------------
```

3. Install a sufficient number of IdM replicas from the restored server to handle the deployment load. There are now two disconnected IdM deployments running in parallel.

4. Switch the IdM clients to use the new deployment by hard-coding references to the new IdM replicas. See Adjusting IdM clients during recovery .

5. Stop and uninstall IdM servers from the previous deployment. See Uninstalling an IdM server .

Verification

1. Test the Kerberos server on every new replica by successfully retrieving a Kerberos ticket-granting ticket as an IdM user.

```
[root@server ~]# kinit admin
Password for admin@EXAMPLE.COM:

[root@server ~]# klist
Ticket cache: KCM:0
Default principal: admin@EXAMPLE.COM

Valid starting       Expires              Service principal
10/31/2019 15:51:37  11/01/2019 15:51:02  HTTP/server.example.com@EXAMPLE.COM
10/31/2019 15:51:08  11/01/2019 15:51:02  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

2. Test the Directory Server and SSSD configuration on every new replica by retrieving user information.

```
[root@server ~]# ipa user-show admin
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  Principal alias: admin@EXAMPLE.COM
  UID: 1965200000
  GID: 1965200000
  Account disabled: False
  Password: True
  Member of groups: admins, trust admins
  Kerberos keys available: True
```

3. Test the CA server on every new CA replica with the **ipa cert-show** command.

```
[root@server ~]# ipa cert-show 1
  Issuing CA: ipa
  Certificate: MIIEgjCCAuqgAwIBAgIjoSIP...
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
  Issuer: CN=Certificate Authority,O=EXAMPLE.COM
  Not Before: Thu Oct 31 19:43:29 2019 UTC
  Not After: Mon Oct 31 19:43:29 2039 UTC
  Serial number: 1
  Serial number (hex): 0x1
  Revoked: False
```

CHAPTER 5. RECOVERING FROM DATA LOSS WITH IDM BACKUPS

You can use the **ipa-restore** utility or Ansible playbooks to restore an IdM server to a previous state captured in an IdM backup.

For details how to perform IdM backups, see the following chapters in the Identity Management documentation:

- Backing up and restoring IdM
- Backing up and restoring IdM servers using Ansible playbooks

CHAPTER 6. MANAGING DATA LOSS

The proper response to a data loss event will depend on the number of replicas that have been affected and the type of lost data.

6.1. RESPONDING TO ISOLATED DATA LOSS

When a data loss event occurs, minimize replicating the data loss by immediately isolating the affected servers. Then create replacement replicas from the unaffected remainder of the environment.

Prerequisites

- A robust IdM replication topology with multiple replicas. See Preparing for server loss with replication.

Procedure

1. To limit replicating the data loss, disconnect all affected replicas from the rest of the topology by removing their replication topology segments.

   a. Display all **domain** replication topology segments in the deployment.

```
[root@server ~]# ipa topologysegment-find
Suffix name: domain
------------------
8 segments matched
------------------
  Segment name: segment1
  Left node: server.example.com
  Right node: server2.example.com
  Connectivity: both

...

----------------------------
Number of entries returned 8
----------------------------
```

   b. Delete all **domain** topology segments involving the affected servers.

```
[root@server ~]# ipa topologysegment-del
Suffix name: domain
Segment name: segment1
-----------------------------
Deleted segment "segment1"
-----------------------------
```

   c. Perform the same actions with any **ca** topology segments involving any affected servers.

```
[root@server ~]# ipa topologysegment-find
Suffix name: ca
------------------
1 segments matched
------------------
```

```
  Segment name: ca_segment
  Left node: server.example.com
  Right node: server2.example.com
  Connectivity: both
----------------------------
Number of entries returned 1
----------------------------

[root@server ~]# ipa topologysegment-del
Suffix name: ca
Segment name: ca_segment
----------------------------
Deleted segment "ca_segment"
----------------------------
```

2. The servers affected by the data loss must be abandoned. To create replacement replicas, see Recovering multiple servers with replication.

6.2. RESPONDING TO LIMITED DATA LOSS AMONG ALL SERVERS

A data loss event can affect all replicas in the environment, such as replication carrying out an accidental deletion among all servers. If data loss is known and limited, manually re-add lost data.

Prerequisites

- A Virtual Machine (VM) snapshot or IdM backup of an IdM server that contains the lost data.

Procedure

1. If you need to review any lost data, restore the VM snapshot or backup to an isolated server on a separate network.
2. Add the missing information to the database using **ipa** or **ldapadd** commands.

Additional resources

- Recovering from data loss with VM snapshots
- Backing Up and Restoring IdM

6.3. RESPONDING TO UNDEFINED DATA LOSS AMONG ALL SERVERS

If data loss is severe or undefined, deploy a new environment from a Virtual Machine (VM) snapshot of a server.

Prerequisites

- A Virtual Machine (VM) snapshot contains the lost data.

Procedure

1. Restore an IdM Certificate Authority (CA) Replica from a VM snapshot to a known good state, and deploy a new IdM environment from it. See Recovering from only a VM snapshot .

2. Add any data created after the snapshot was taken using **ipa** or **ldapadd** commands.

Additional resources

- Recovering from data loss with VM snapshots

CHAPTER 7. ADJUSTING IDM CLIENTS DURING RECOVERY

While IdM servers are being restored, you may need to adjust IdM clients to reflect changes in the replica topology.

Procedure

1. Adjusting DNS configuration:

   a. If **/etc/hosts** contains any references to IdM servers, ensure that hard-coded IP-to-hostname mappings are valid.

   b. If IdM clients are using IdM DNS for name resolution, ensure that the **nameserver** entries in **/etc/resolv.conf** point to working IdM replicas providing DNS services.

2. Adjusting Kerberos configuration:

   a. By default, IdM clients look to DNS Service records for Kerberos servers, and will adjust to changes in the replica topology:

   ```
   [root@client ~]# grep dns_lookup_kdc /etc/krb5.conf
     dns_lookup_kdc = true
   ```

   b. If IdM clients have been hard-coded to use specific IdM servers in **/etc/krb5.conf**:

   ```
   [root@client ~]# grep dns_lookup_kdc /etc/krb5.conf
     dns_lookup_kdc = false
   ```

   make sure **kdc**, **master_kdc** and **admin_server** entries in **/etc/krb5.conf** are pointing to IdM servers that work properly:

   ```
   [realms]
    EXAMPLE.COM = {
     kdc = functional-server.example.com:88
     master_kdc = functional-server.example.com:88
     admin_server = functional-server.example.com:749
     default_domain = example.com
     pkinit_anchors = FILE:/var/lib/ipa-client/pki/kdc-ca-bundle.pem
     pkinit_pool = FILE:/var/lib/ipa-client/pki/ca-bundle.pem
   }
   ```

3. Adjusting SSSD configuration:

   a. By default, IdM clients look to DNS Service records for LDAP servers and adjust to changes in the replica topology:

   ```
   [root@client ~]# grep ipa_server /etc/sssd/sssd.conf
   ipa_server = _srv_, functional-server.example.com
   ```

   b. If IdM clients have been hard-coded to use specific IdM servers in **/etc/sssd/sssd.conf**, make sure the **ipa_server** entry points to IdM servers that are working properly:

   ```
   [root@client ~]# grep ipa_server /etc/sssd/sssd.conf
   ipa_server = functional-server.example.com
   ```

4. Clearing SSSD's cached information:

   - The SSSD cache may contain outdated information pertaining to lost servers. If users experience inconsistent authentication problems, purge the SSSD cache :

     ```
     [root@client ~]# sss_cache -E
     ```

Verification

1. Verify the Kerberos configuration by retrieving a Kerberos Ticket-Granting-Ticket as an IdM user.

   ```
   [root@client ~]# kinit admin
   Password for admin@EXAMPLE.COM:

   [root@client ~]# klist
   Ticket cache: KCM:0
   Default principal: admin@EXAMPLE.COM

   Valid starting       Expires              Service principal
   10/31/2019 18:44:58  11/25/2019 18:44:55  krbtgt/EXAMPLE.COM@EXAMPLE.COM
   ```

2. Verify the SSSD configuration by retrieving IdM user information.

   ```
   [root@client ~]# id admin
   uid=1965200000(admin) gid=1965200000(admins) groups=1965200000(admins)
   ```