Red Hat Enterprise Linux 9

Accessing Identity Management services

A guide for logging in to IdM and managing its services

Last Updated: 2022-08-30

Red Hat Enterprise Linux 9 Accessing Identity Management services

A guide for logging in to IdM and managing its services

Legal Notice

Abstract

This documentation collection provides instructions on how to log in as an Identity Management (IdM) user on the console and WebUI, and how to start and stop RHEL 9 IdM services.

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.

In Identity Management, planned terminology replacements include:

- ***block list*** replaces *blacklist*
- ***allow list*** replaces *whitelist*
- ***secondary*** replaces *slave*
- The word *master* is being replaced with more precise language, depending on the context:
  - ***IdM server*** replaces *IdM master*
  - ***CA renewal server*** replaces *CA renewal master*
  - ***CRL publisher server*** replaces *CRL master*
  - ***multi-supplier*** replaces *multi-master*

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

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

Submitting comments on specific passages

1. View the documentation in the **Multi-page HTML** format and ensure that you see the **Feedback** button in the upper right corner after the page fully loads.
2. Use your cursor to highlight the part of the text that you want to comment on.
3. Click the **Add Feedback** button that appears near the highlighted text.
4. Add your feedback and click **Submit**.

Submitting feedback through Bugzilla (account required)

1. Log in to the Bugzilla website.
2. Select the correct version from the **Version** menu.
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 **Submit Bug**.

CHAPTER 1. LOGGING IN TO IDENTITY MANAGEMENT FROM THE COMMAND LINE

Identity Management (IdM) uses the Kerberos protocol to support single sign-on. Single sign-on means that the user enters the correct user name and password only once, and then accesses IdM services without the system prompting for the credentials again.

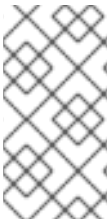
IMPORTANT

In IdM, the System Security Services Daemon (SSSD) automatically obtains a ticket-granting ticket (TGT) for a user after the user successfully logs in to the desktop environment on an IdM client machine with the corresponding Kerberos principal name. This means that after logging in, the user is not required to use the **kinit** utility to access IdM resources.

If you have cleared your Kerberos credential cache or your Kerberos TGT has expired, you need to request a Kerberos ticket manually to access IdM resources. The following sections present basic user operations when using Kerberos in IdM.

1.1. USING KINIT TO LOG IN TO IDM MANUALLY

This procedure describes using the **kinit** utility to authenticate to an Identity Management (IdM) environment manually. The **kinit** utility obtains and caches a Kerberos ticket-granting ticket (TGT) on behalf of an IdM user.



NOTE

Only use this procedure if you have destroyed your initial Kerberos TGT or if it has expired. As an IdM user, when logging onto your local machine you are also automatically logging in to IdM. This means that after logging in, you are not required to use the **kinit** utility to access IdM resources.

Procedure

1. To log in to IdM

   - under the user name of the user who is currently logged in on the local system, use **kinit** without specifying a user name. For example, if you are logged in as **example_user** on the local system:

     ```
     [example_user@server ~]$ kinit
     Password for example_user@EXAMPLE.COM:
     [example_user@server ~]$
     ```

     If the user name of the local user does not match any user entry in IdM, the authentication attempt fails:

     ```
     [example_user@server ~]$ kinit
     kinit: Client 'example_user@EXAMPLE.COM' not found in Kerberos database while
     getting initial credentials
     ```

   - using a Kerberos principal that does not correspond to your local user name, pass the required user name to the **kinit** utility. For example, to log in as the **admin** user:

```
[example_user@server ~]$ kinit admin
Password for admin@EXAMPLE.COM:
[example_user@server ~]$
```

2. Optionally, to verify that the login was successful, use the **klist** utility to display the cached TGT. In the following example, the cache contains a ticket for the **example_user** principal, which means that on this particular host, only **example_user** is currently allowed to access IdM services:

```
$ klist
Ticket cache: KEYRING:persistent:0:0
Default principal: example_user@EXAMPLE.COM

Valid starting       Expires              Service principal
11/10/2019 08:35:45  11/10/2019 18:35:45  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

1.2. DESTROYING A USER'S ACTIVE KERBEROS TICKET

This section describes how to clear the credentials cache that contains the user's active Kerberos ticket.

Procedure

1. To destroy your Kerberos ticket:

```
[example_user@server ~]$ kdestroy
```

2. Optionally, to check that the Kerberos ticket has been destroyed:

```
[example_user@server ~]$ klist
klist: Credentials cache keyring 'persistent:0:0' not found
```

1.3. CONFIGURING AN EXTERNAL SYSTEM FOR KERBEROS AUTHENTICATION

This section describes how to configure an external system so that Identity Management (IdM) users can log in to IdM from the external system using their Kerberos credentials.

Enabling Kerberos authentication on external systems is especially useful when your infrastructure includes multiple realms or overlapping domains. It is also useful if the system has not been enrolled into any IdM domain through **ipa-client-install**.

To enable Kerberos authentication to IdM from a system that is not a member of the IdM domain, define an IdM-specific Kerberos configuration file on the external system.

Prerequisites

- The **krb5-workstation** package is installed on the external system.
  To find out whether the package is installed, use the following CLI command:

```
# dnf list installed krb5-workstation
Installed Packages
krb5-workstation.x86_64    1.16.1-19.el8    @BaseOS
```

Procedure

1. Copy the **/etc/krb5.conf** file from the IdM server to the external system. For example:

```
# scp /etc/krb5.conf root@externalsystem.example.com:/etc/krb5_ipa.conf
```

WARNING

Do not overwrite the existing **krb5.conf** file on the external system.

2. On the external system, set the terminal session to use the copied IdM Kerberos configuration file:

```
$ export KRB5_CONFIG=/etc/krb5_ipa.conf
```

The **KRB5_CONFIG** variable exists only temporarily until you log out. To prevent this loss, export the variable with a different file name.

3. Copy the Kerberos configuration snippets from the **/etc/krb5.conf.d/** directory to the external system.

Users on the external system can now use the **kinit** utility to authenticate against the IdM server.

1.4. ADDITIONAL RESOURCES

- The **krb5.conf(5)** man page.
- The **kinit(1)** man page.
- The **klist(1)** man page.
- The **kdestroy(1)** man page.

CHAPTER 2. VIEWING, STARTING AND STOPPING THE IDENTITY MANAGEMENT SERVICES

Identity Management (IdM) servers are Red Hat Enterprise Linux systems that work as domain controllers (DCs). A number of different services are running on IdM servers, most notably the Directory Server, Certificate Authority (CA), DNS, and Kerberos.

2.1. THE IDM SERVICES

This section describes the services that can be installed and run on the IdM servers and clients.

List of services hosted by IdM servers

Most of the following services are not strictly required to be installed on the IdM server. For example, you can install services such as a certificate authority (CA) or DNS server on an external server outside the IdM domain.

Kerberos

the **krb5kdc** and **kadmin** services

IdM uses the **Kerberos** protocol to support single sign-on. With Kerberos, users only need to present the correct username and password once and can access IdM services without the system prompting for credentials again.

Kerberos is divided into two parts:

- The **krb5kdc** service is the Kerberos Authentication service and Key Distribution Center (KDC) daemon.
- The **kadmin** service is the Kerberos database administration program.

For information on how to authenticate using Kerberos in IdM, see Logging in to Identity Management from the command line and Logging in to IdM in the Web UI: Using a Kerberos ticket .

LDAP directory server

the **dirsrv** service

The IdM **LDAP directory server** instance stores all IdM information, such as information related to Kerberos, user accounts, host entries, services, policies, DNS, and others. The LDAP directory server instance is based on the same technology as Red Hat Directory Server . However, it is tuned to IdM-specific tasks.

Certificate Authority

the **pki-tomcatd** service

The integrated **certificate authority (CA)** is based on the same technology as Red Hat Certificate System. **pki** is the Command-Line Interface for accessing Certificate System services.

You can also install the server without the integrated CA if you create and provide all required certificates independently.

For more information, see Planning your CA services .

Domain Name System (DNS)

the **named** service

IdM uses **DNS** for dynamic service discovery. The IdM client installation utility can use information from DNS to automatically configure the client machine. After the client is enrolled in the IdM domain, it uses DNS to locate IdM servers and services within the domain. The **BIND** (Berkeley Internet Name Domain) implementation of the DNS (Domain Name System) protocols in Red Hat Enterprise Linux includes the **named** DNS server. **named-pkcs11** is a version of the BIND DNS server built with native support for the PKCS#11 cryptographic standard.

For information, see Planning your DNS services and host names .

Apache HTTP Server

the **httpd** service

The **Apache HTTP web server** provides the IdM Web UI, and also manages communication between the Certificate Authority and other IdM services.

Samba / Winbind

smb and **winbind** services

Samba implements the Server Message Block (SMB) protocol, also known as the Common Internet File System (CIFS) protocol, in Red Hat Enterprise Linux. Via the smb service, the SMB protocol enables you to access resources on a server, such as file shares and shared printers. If you have configured a Trust with an Active Directory (AD) environment, the `Winbind` service manages communication between IdM servers and AD servers.

One-time password (OTP) authentication

the **ipa-otpd** services

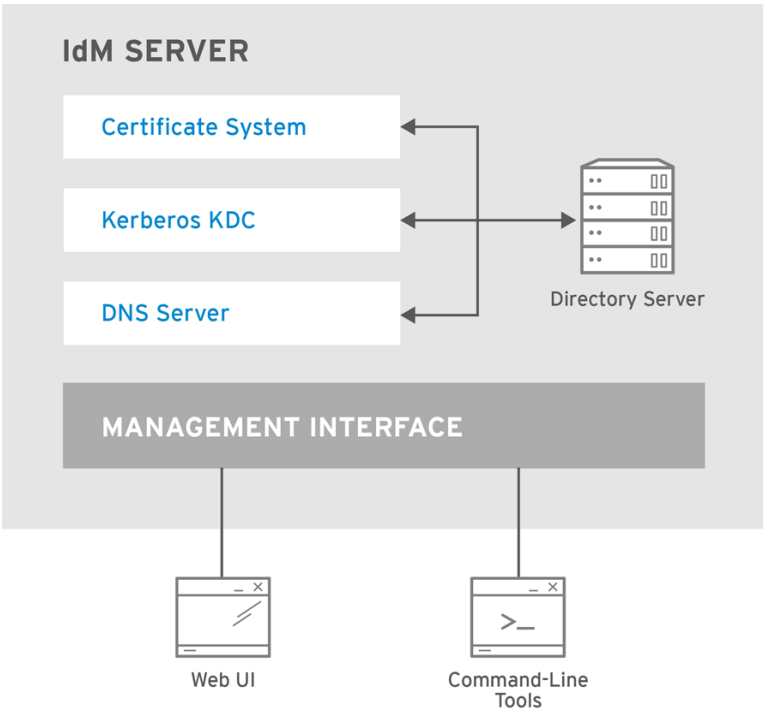
One-time passwords (OTP) are passwords that are generated by an authentication token for only one session, as part of two-factor authentication. OTP authentication is implemented in Red Hat Enterprise Linux via the **ipa-otpd** service.

For more information, see Logging in to the Identity Management Web UI using one time passwords .

OpenDNSSEC

the **ipa-dnskeysyncd** service

OpenDNSSEC is a DNS manager that automates the process of keeping track of DNS security extensions (DNSSEC) keys and the signing of zones. The **ipa-dnskeysyncd** service manages synchronization between the IdM Directory Server and OpenDNSSEC.



List of services hosted by IdM clients

- **System Security Services Daemon**: the **sssd** service

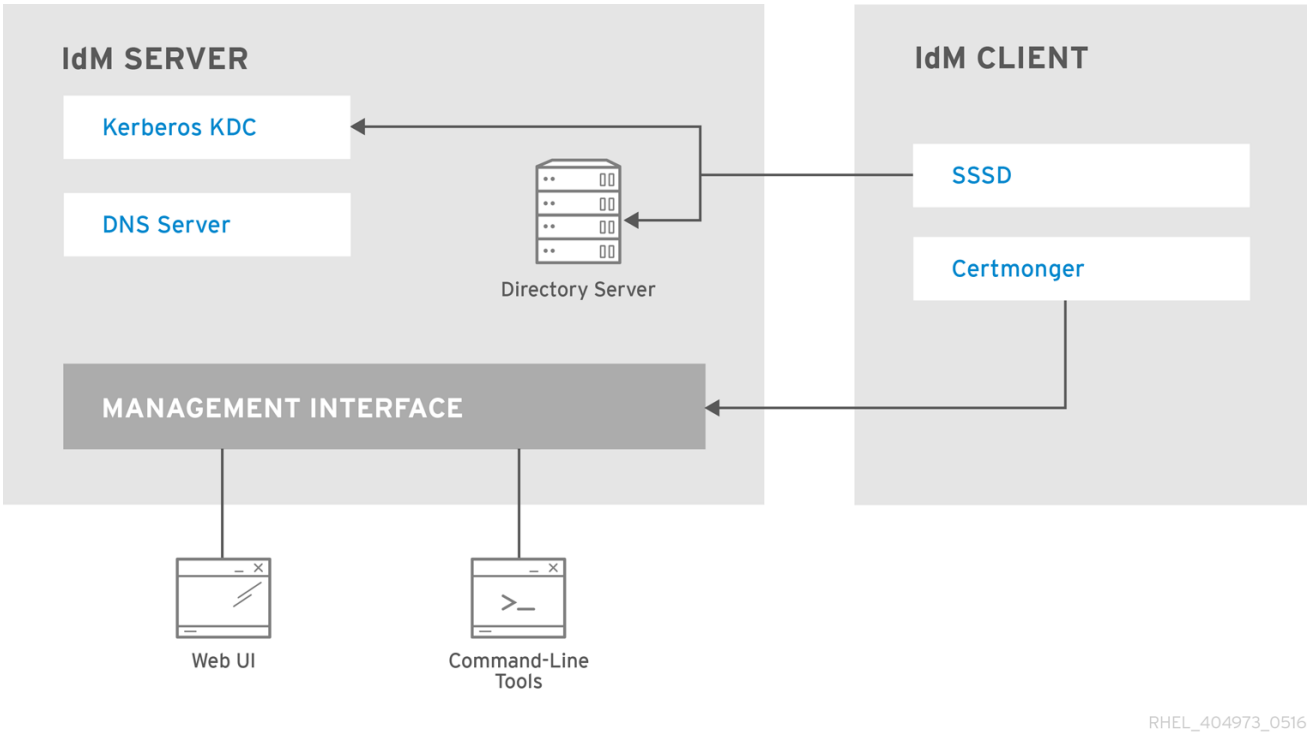
The **System Security Services Daemon** (SSSD) is the client-side application that manages user authentication and caching credentials. Caching enables the local system to continue normal authentication operations if the IdM server becomes unavailable or if the client goes offline.

For more information, see Understanding SSSD and its benefits .

- **Certmonger**: the **certmonger** service

The **certmonger** service monitors and renews the certificates on the client. It can request new certificates for the services on the system.

For more information, see Obtaining an IdM certificate for a service using certmonger .

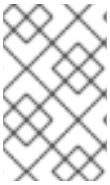


2.2. VIEWING THE STATUS OF IDM SERVICES

To view the status of the IdM services that are configured on your IdM server, run the **ipactl status** command:

```
[root@server ~]# ipactl status
Directory Service: RUNNING
krb5kdc Service: RUNNING
kadmin Service: RUNNING
named Service: RUNNING
httpd Service: RUNNING
pki-tomcatd Service: RUNNING
smb Service: RUNNING
winbind Service: RUNNING
ipa-otpd Service: RUNNING
ipa-dnskeysyncd Service: RUNNING
ipa: INFO: The ipactl command was successful
```

The output of the **ipactl status** command on your server depends on your IdM configuration. For example, if an IdM deployment does not include a DNS server, the **named** service is not present in the list.



NOTE

You cannot use the IdM web UI to view the status of all the IdM services running on a particular IdM server. Kerberized services running on different servers can be viewed in the **Identity → Services** tab of the IdM web UI.

You can start or stop the entire server, or an individual service only.

To start, stop, or restart the entire IdM server, see:

- Starting and stopping the entire Identity Management server: the ipactl utility

To start, stop, or restart an individual IdM service, see:

- Starting and stopping an individual Identity Management service: the systemctl utility

To display the version of IdM software, see:

- Methods for displaying IdM software version

2.3. STARTING AND STOPPING THE ENTIRE IDENTITY MANAGEMENT SERVER: THE IPACTL UTILITY

Use the **ipactl** utility to stop, start, or restart the entire IdM server along with all the installed services. Using the **ipactl** utility ensures all services are stopped, started, or restarted in the appropriate order. You do not need to have a valid Kerberos ticket to run the **ipactl** commands.

ipactl commands

To start the entire IdM server:

```
# ipactl start
```

To stop the entire IdM server:

```
# ipactl stop
```

To restart the entire IdM server:

```
# ipactl restart
```

To show the status of all the services that make up IdM:

```
# ipactl status
```

IMPORTANT

You cannot use the IdM web UI to perform the **ipactl** commands.

2.4. STARTING AND STOPPING AN INDIVIDUAL IDENTITY MANAGEMENT SERVICE: THE SYSTEMCTL UTILITY

Changing IdM configuration files manually is generally not recommended. However, certain situations require that an administrator performs a manual configuration of specific services. In such situations, use the **systemctl** utility to stop, start, or restart an individual IdM service.

For example, use **systemctl** after customizing the Directory Server behavior, without modifying the other IdM services:

```
# systemctl restart dirsrv@REALM-NAME.service
```

Also, when initially deploying an IdM trust with Active Directory, modify the **/etc/sssd/sssd.conf** file, adding:

- specific parameters to tune the timeout configuration options in an environment where remote servers have a high latency
- specific parameters to tune the Active Directory site affinity
- overrides for certain configuration options that are not provided by the global IdM settings

To apply the changes you have made in the **/etc/sssd/sssd.conf** file:

```
# systemctl restart sssd.service
```

Running **systemctl restart sssd.service** is required because the System Security Services Daemon (SSSD) does not automatically re-read or re-apply its configuration.

Note that for changes that affect IdM identity ranges, a complete server reboot is recommended.

IMPORTANT

To restart multiple IdM domain services, always use **ipactl**. Because of dependencies between the services installed with the IdM server, the order in which they are started and stopped is critical. The **ipactl** utility ensures that the services are started and stopped in the appropriate order.

Useful systemctl commands

To start a particular IdM service:

```
# systemctl start name.service
```

To stop a particular IdM service:

```
# systemctl stop name.service
```

To restart a particular IdM service:

```
# systemctl restart name.service
```

To view the status of a particular IdM service:

```
# systemctl status name.service
```

IMPORTANT

You cannot use the IdM web UI to start or stop the individual services running on IdM servers. You can only use the web UI to modify the settings of a Kerberized service by navigating to **Identity → Services** and selecting the service.

2.5. METHODS FOR DISPLAYING IDM SOFTWARE VERSION

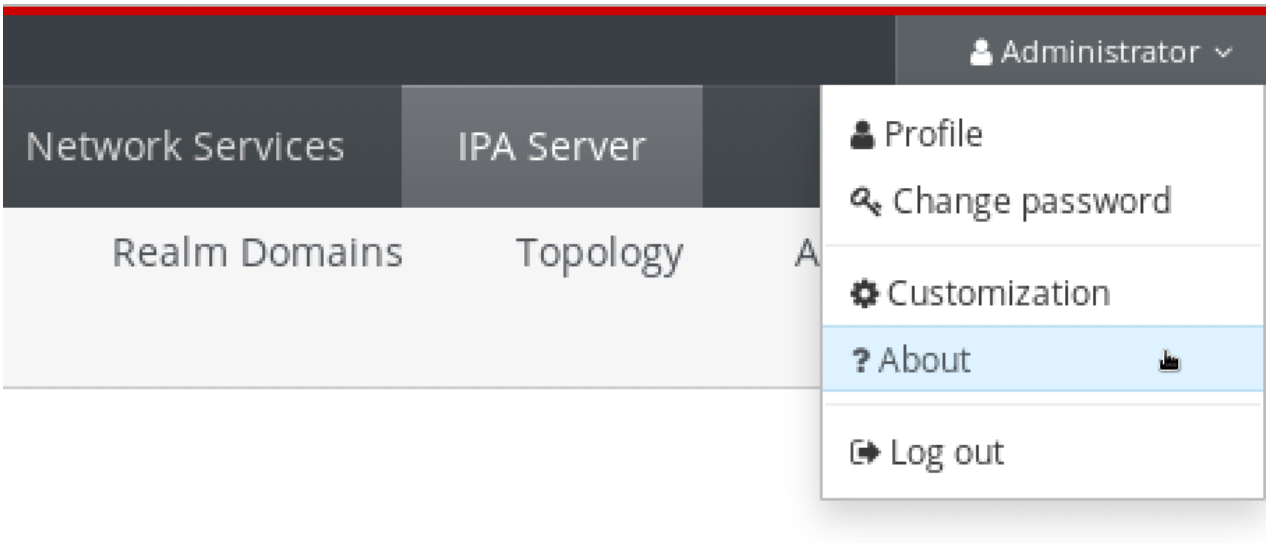
You can display the IdM version number with:

- the IdM WebUI

- **ipa** commands
- **rpm** commands

Displaying version through the WebUI

In the IdM WebUI, the software version can be displayed by choosing **About** from the username menu at the top-right.



Displaying version with ipa commands

From the command line, use the **ipa --version** command.

```
[root@server ~]# ipa --version
VERSION: 4.8.0, API_VERSION: 2.233
```

Displaying version with rpm commands

If IdM services are not operating properly, you can use the **rpm** utility to determine the version number of the **ipa-server** package that is currently installed.

```
[root@server ~]# rpm -q ipa-server
ipa-server-4.8.0-11.module+el8.1.0+4247+9f3fd721.x86_64
```

CHAPTER 3. INTRODUCTION TO THE IDM COMMAND-LINE UTILITIES

The following sections describe the basics of using the Identity Management (IdM) command-line utilities.

Prerequisites

- Installed and accessible IdM server.
  For details, see Installing Identity Management.
- To use the IPA command-line interface, authenticate to IdM with a valid Kerberos ticket.

3.1. WHAT IS THE IPA COMMAND LINE INTERFACE

The IPA command-line interface (CLI) is the basic command-line interface for Identity Management (IdM) administration.

It supports a lot of subcommands for managing IdM, such as the **ipa user-add** command to add a new user.

IPA CLI allows you to:

- Add, manage, or remove users, groups, hosts and other objects in the network.
- Manage certificates.
- Search entries.
- Display and list objects.
- Set access rights.
- Get help with the correct command syntax.

3.2. WHAT IS THE IPA HELP

The IPA help is a built-in documentation system for the IdM server.

The IPA command-line interface (CLI) generates available help topics from loaded IdM plugin modules. To use the IPA help utility, you must:

- Have an IdM server installed and running.
- Be authenticated with a valid Kerberos ticket.

Entering the **ipa help** command without options displays information about basic help usage and the most common command examples.

You can use the following options for different **ipa help** use cases:

```
$ ipa help [TOPIC | COMMAND | topics | commands]
```

- **[]** – Brackets mean that all parameters are optional and you can write just **ipa help** and the command will be executed.
- | – The pipe character means **or**. Therefore, you can specify a **TOPIC**, a **COMMAND**, or **topics**, or **commands**, with the basic **ipa help** command:
  - **topics** – You can run the command **ipa help topics** to display a list of topics that are covered by the IPA help, such as **user**, **cert**, **server** and many others.
  - **TOPIC** – The **TOPIC** with capital letters is a variable. Therefore, you can specify a particular topic, for example, **ipa help user**.
  - **commands** – You can enter the command **ipa help commands** to display a list of commands which are covered by the IPA help, for example, **user-add**, **ca-enable**, **server-show** and many others.
  - **COMMAND** – The **COMMAND** with capital letters is a variable. Therefore, you can specify a particular command, for example, **ipa help user-add**.

3.3. USING IPA HELP TOPICS

The following procedure describes how to use the IPA help in the command-line interface.

Procedure

1. Open a terminal and connect to the IdM server.
2. Enter **ipa help topics** to display a list of topics covered by help.

```
$ ipa help topics
```

3. Select one of the topics and create a command according to the following pattern: **ipa help *[topic_name]***. Instead of the **topic_name** string, add one of the topics you listed in the previous step.
   In the example, we use the following topic: **user**

```
$ ipa help user
```

4. If the IPA help output is too long and you cannot see the whole text, use the following syntax:

```
$ ipa help user | less
```

   You can then scroll down and read the whole help.

The IPA CLI displays a help page for the **user** topic. After reading the overview, you can see many examples with patterns for working with topic commands.

3.4. USING IPA HELP COMMANDS

The following procedure describes how to create IPA help commands in the command-line interface.

Procedure

1. Open a terminal and connect to the IdM server.

2. Enter **ipa help commands** to display a list of commands covered by help.

```
$ ipa help commands
```

3. Select one of the commands and create a help command according to the following pattern: **ipa help *<COMMAND>***. Instead of the **<COMMAND>** string, add one of the commands you listed in the previous step.

```
$ ipa help user-add
```

Additional resources

- The **ipa** man page.

3.5. STRUCTURE OF IPA COMMANDS

The IPA CLI distinguishes the following types of commands:

- **Built-in commands** – Built-in commands are all available in the IdM server.
- **Plug-in provided commands**

The structure of IPA commands allows you to manage various types of objects. For example:

- Users,
- Hosts,
- DNS records,
- Certificates,

and many others.

For most of these objects, the IPA CLI includes commands to:

- Add (**add**)
- Modify (**mod**)
- Delete (**del**)
- Search (**find**)
- Display (**show**)

Commands have the following structure:

ipa user-add, **ipa user-mod**, **ipa user-del**, **ipa user-find**, **ipa user-show**

ipa host-add, **ipa host-mod**, **ipa host-del**, **ipa host-find**, **ipa host-show**

ipa dnsrecord-add, **ipa dnsrecord-mod**, **ipa dnsrecord-del**, **ipa dnsrecord-find**, **ipa dnrecord-show**

You can create a user with the **ipa user-add [options]**, where **[options]** are optional. If you use just the **ipa user-add** command, the script asks you for details one by one.

To change an existing object, you need to define the object, therefore the command also includes an object: **ipa user-mod USER_NAME [options]**.

3.6. USING AN IPA COMMAND TO ADD A USER ACCOUNT TO IDM

The following procedure describes how to add a new user to the Identity Management (IdM) database using the command line.

Prerequisites

- You need to have administrator privileges to add user accounts to the IdM server.

Procedure

1. Open a terminal and connect to the IdM server.
2. Enter the command for adding a new user:

   ```
   $ ipa user-add
   ```

   The command runs a script that prompts you to provide basic data necessary for creating a user account.
3. In the **First name:** field, enter the first name of the new user and press the **Enter** key.
4. In the **Last name:** field, enter the last name of the new user and press the **Enter** key.
5. In the **User login [suggested user name]:**enter the user name, or just press the **Enter** key to accept the suggested user name.
   The user name must be unique for the whole IdM database. If an error occurs because that user name already exists, repeat the process with the **ipa user-add** command and use a different, unique user name.

After you add the user name, the user account is added to the IdM database and the IPA command line interface (CLI) prints the following output:

```
----------------------
Added user "euser"
----------------------
User login: euser
First name: Example
Last name: User
Full name: Example User
Display name: Example User
Initials: EU
Home directory: /home/euser
GECOS: Example User
Login shell: /bin/sh
Principal name: euser@IDM.EXAMPLE.COM
Principal alias: euser@IDM.EXAMPLE.COM
Email address: euser@idm.example.com
UID: 427200006
GID: 427200006
```

```
Password: False
Member of groups: ipausers
Kerberos keys available: False
```

NOTE

By default, a user password is not set for the user account. To add a password while creating a user account, use the **ipa user-add** command with the following syntax:

```
$ ipa user-add --first=Example --last=User --password
```

The IPA CLI then prompts you to add or confirm a user name and password.

If the user has been created already, you can add the password with the **ipa user-mod** command.

Additional resources

- Run the **ipa help user-add** command for more information about parameters.

3.7. USING AN IPA COMMAND TO MODIFY A USER ACCOUNT IN IDM

You can change many parameters for each user account. For example, you can add a new password to the user.

Basic command syntax is different from the **user-add** syntax because you need to define the existing user account for which you want to perform changes, for example, add a password.

Prerequisites

- You need to have administrator privileges to modify user accounts.

Procedure

1. Open a terminal and connect to the IdM server.
2. Enter the **ipa user-mod** command, specify the user to modify, and any options, such as **--password** for adding a password:

   ```
   $ ipa user-mod euser --password
   ```

   The command runs a script where you can add the new password.
3. Enter the new password and press the **Enter** key.

The IPA CLI prints the following output:

```
----------------------
Modified user "euser"
----------------------
User login: euser
First name: Example
Last name: User
Home directory: /home/euser
```

```
Principal name: euser@IDM.EXAMPLE.COM
Principal alias: euser@IDM.EXAMPLE.COM
Email address: euser@idm.example.com
UID: 427200006
GID: 427200006
Password: True
Member of groups: ipausers
Kerberos keys available: True
```

The user password is now set for the account and the user can log into IdM.

Additional resources

- Run the **ipa help user-mod** command for more information about parameters.

3.8. HOW TO SUPPLY A LIST OF VALUES TO THE IDM UTILITIES

Identity Management (IdM) stores values for multi-valued attributes in lists.

IdM supports the following methods of supplying multi-valued lists:

- Using the same command-line argument multiple times within the same command invocation:

```
$ ipa permission-add --right=read --permissions=write --permissions=delete ...
```

- Alternatively, you can enclose the list in curly braces, in which case the shell performs the expansion:

```
$ ipa permission-add --right={read,write,delete} ...
```

The examples above show a command **permission-add** which adds permissions to an object. The object is not mentioned in the example. Instead of **...** you need to add the object for which you want to add permissions.

When you update such multi-valued attributes from the command line, IdM completely overwrites the previous list of values with a new list. Therefore, when updating a multi-valued attribute, you must specify the whole new list, not just a single value you want to add.

For example, in the command above, the list of permissions includes reading, writing and deleting. When you decide to update the list with the **permission-mod** command, you must add all values, otherwise those not mentioned will be deleted.

Example 1: – The **ipa permission-mod** command updates all previously added permissions.

```
$ ipa permission-mod --right=read --right=write --right=delete ...
```

or

```
$ ipa permission-mod --right={read,write,delete} ...
```

Example 2 – The **ipa permission-mod** command deletes the **--right=delete** argument because it is not included in the command:

```
$ ipa permission-mod --right=read --right=write ...
```

-

or

```
$ ipa permission-mod --right={read,write} ...
```

3.9. HOW TO USE SPECIAL CHARACTERS WITH THE IDM UTILITIES

When passing command-line arguments that include special characters to the **ipa** commands, escape these characters with a backslash (\). For example, common special characters include angle brackets (< and >), ampersand (&), asterisk (*), or vertical bar (|).

For example, to escape an asterisk (*):

```
$ ipa certprofile-show certificate_profile --out=exported\*profile.cfg
```

Commands containing unescaped special characters do not work as expected because the shell cannot properly parse such characters.

CHAPTER 4. SEARCHING IDENTITY MANAGEMENT ENTRIES FROM THE COMMAND LINE

The following sections describe how to use IPA commands, which helps you to find or show objects.

4.1. OVERVIEW OF LISTING IDM ENTRIES

This section describes the **ipa *-find** commands, which can help you to search for a particular type of IdM entries.

To list all the **find** commands, use the following ipa help command:

```
$ ipa help commands | grep find
```

You may need to check if a particular user is included in the IdM database. You can then list all users with the following command:

```
$ ipa user-find
```

To list user groups whose specified attributes contain a keyword:

```
$ ipa group-find keyword
```

For example the **ipa group-find *admin*** command lists all groups whose names or descriptions include string **admin**:

```
----------------
3 groups matched
----------------
  Group name: admins
  Description: Account administrators group
  GID: 427200002

  Group name: editors
  Description: Limited admins who can edit other users
  GID: 427200002

  Group name: trust admins
  Description: Trusts administrators group
```

When searching user groups, you can also limit the search results to groups that contain a particular user:

```
$ ipa group-find --user=user_name
```

To search for groups that do not contain a particular user:

```
$ ipa group-find --no-user=user_name
```

4.2. SHOWING DETAILS FOR A PARTICULAR ENTRY

Use the **ipa *-show** command to display details about a particular IdM entry.

Procedure

- To display details about a host named *server.example.com*:

```
$ ipa host-show server.example.com

Host name: server.example.com
Principal name: host/server.example.com@EXAMPLE.COM
...
```

4.3. ADJUSTING THE SEARCH SIZE AND TIME LIMIT

Some queries, such as requesting a list of IdM users, can return a very large number of entries. By tuning these search operations, you can improve the overall server performance when running the **ipa *-find** commands, such as **ipa user-find**, and when displaying corresponding lists in the Web UI.

Search size limit

Defines the maximum number of entries returned for a request sent to the server from a client's CLI or from a browser accessing the IdM Web UI.
Default: 100 entries.

Search time limit

Defines the maximum time (in seconds) that the server waits for searches to run. Once the search reaches this limit, the server stops the search and returns the entries discovered in that time.
Default: 2 seconds.

If you set the values to **-1**, IdM will not apply any limits when searching.

IMPORTANT

Setting search size or time limits too high can negatively affect server performance.

4.3.1. Adjusting the search size and time limit in the command line

The following procedure describes adjusting search size and time limits in the command line:

- Globally
- For a specific entry

Procedure

1. To display current search time and size limits in CLI, use the **ipa config-show** command:

```
$ ipa config-show

Search time limit: 2
Search size limit: 100
```

2. To adjust the limits **globally** for all queries, use the **ipa config-mod** command and add the **--searchrecordslimit** and **--searchtimelimit** options. For example:

```
$ ipa config-mod --searchrecordslimit=500 --searchtimelimit=5
```

3. To **temporarily** adjust the limits only for a specific query, add the **--sizelimit** or **--timelimit** options to the command. For example:

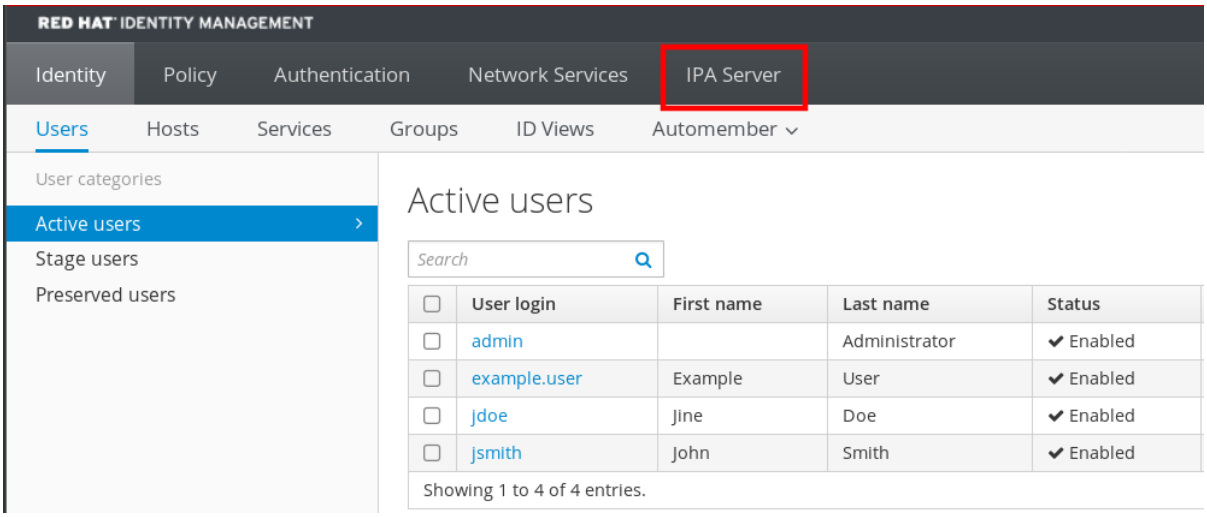
```
$ ipa user-find --sizelimit=200 --timelimit=120
```

4.3.2. Adjusting the search size and time limit in the Web UI

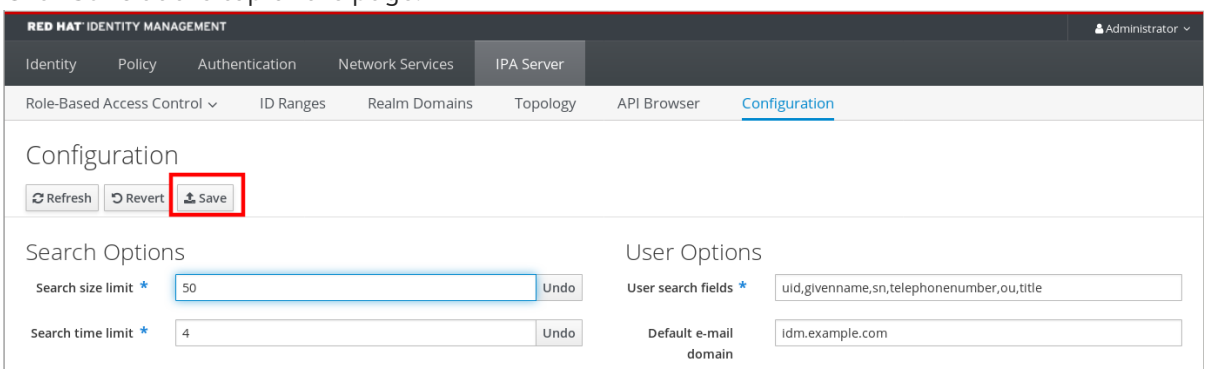
The following procedure describes adjusting global search size and time limits in the IdM Web UI.

Procedure

1. Log in to the IdM Web UI.
2. Click **IPA Server**.



3. On the **IPA Server** tab, click **Configuration**.
4. Set the required values in the **Search Options** area.
   Default values are:
   - Search size limit: 100 entries
   - Search time limit: 2 seconds
5. Click **Save** at the top of the page.



CHAPTER 5. ACCESSING THE IDM WEB UI IN A WEB BROWSER

The following sections provide an overview of the IdM (Identity Management) Web UI and describe how to access it.

5.1. WHAT IS THE IDM WEB UI

The IdM (Identity Management) Web UI is a web application for IdM administration, a graphical alternative to the IdM command line tools.

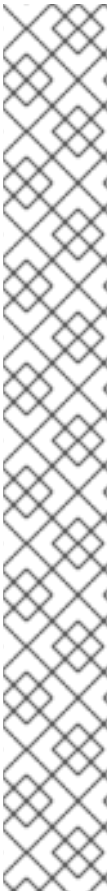
You can access the IdM Web UI as:

- **IdM users:** A limited set of operations depending on permissions granted to the user in the IdM server. Basically, active IdM users can log in to the IdM server and configure their own account. They cannot change settings of other users or the IdM server settings.
- **Administrators:** Full access rights to the IdM server.
- **Active Directory users:** A set of operations depending on permissions granted to the user. Active Directory users can now be administrators for Identity Management. For details, see Enabling AD users to administer IdM .

5.2. WEB BROWSERS SUPPORTED FOR ACCESSING THE WEB UI

Identity Management (IdM) supports the following browsers for connecting to the Web UI:

- Mozilla Firefox 38 and later
- Google Chrome 46 and later



NOTE

You might experience problems accessing the IdM Web UI with a smart card if your browser attempts to use TLS v1.3:

```
[ssl:error] [pid 125757:tid 140436077168384] [client 999.999.999.999:99999] AH:
verify client post handshake
[ssl:error] [pid 125757:tid 140436077168384] [client 999.999.999.999:99999]
AH10158: cannot perform post-handshake authentication
[ssl:error] [pid 125757:tid 140436077168384] SSL Library Error: error:14268117:SSL
routines:SSL_verify_client_post_handshake:extension not received
```

This is because the most recent versions of browsers do not have TLS Post-Handshake Authentication (PHA) enabled by default, or they do not support PHA. PHA is necessary to require a TLS client certificate for only a portion of a web site, such as when accessing the IdM Web UI with smart card authentication.

To resolve this issue, disable TLS v1.3 by adding **-TLSv1.3** to the **SSLProtocol** option in the **/etc/httpd/conf.d/ssl.conf** configuration file:

```
SSLProtocol all -TLSv1 -TLSv1.1 -TLSv1.3
```

Note that IdM manages the **ssl.conf** file and might overwrite its contents during package updates. Verify custom settings after updating IdM packages.

5.3. ACCESSING THE WEB UI

The following procedure describes the first logging in to the IdM (Identity Management) Web UI with a password.

After the first login you can configure your IdM server to authenticate with:

- Kerberos ticket
  For details, see Kerberos authentication in Identity Management .
- Smart card
  For details, see Configuring the IdM server for smart card authentication .
- One time password (OTP) – this can be combined with password and Kerberos authentication.
  For details, see One time password (OTP) authentication in Identity Management .

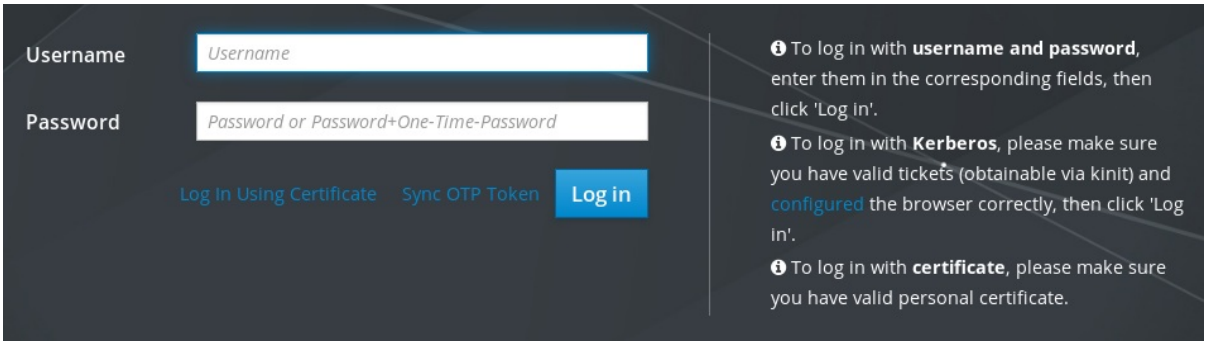
Procedure

1. Type an IdM server URL into the browser address bar. The name will look similarly to the following example:

   ```
   https://server.example.com
   ```

   You just need to change **server.example.com** with a DNS name of your IdM server.

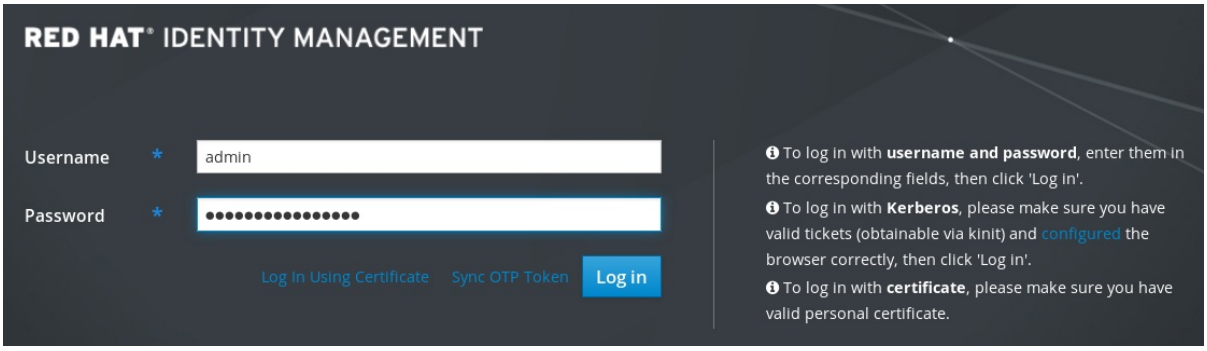
   This opens the IdM Web UI login screen in your browser.



- If the server does not respond or the login screen does not open, check the DNS settings on the IdM server to which you are connecting.
- If you use a self-signed certificate, the browser issues a warning. Check the certificate and accept the security exception to proceed with the login.
  To avoid security exceptions, install a certificate signed by a certificate authority.

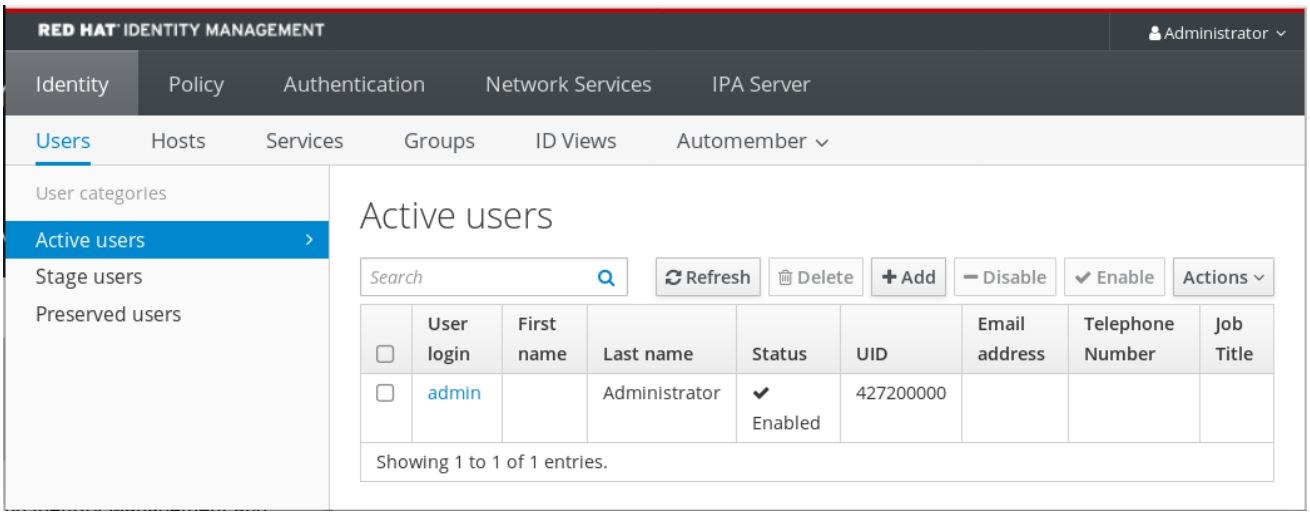
2. On the Web UI login screen, enter the administrator account credentials you added during the IdM server installation.
   For details, see Installing an Identity Management server: With integrated DNS, with an integrated CA.

   You can enter your personal account credentials as well if they are already entered in the IdM server.



3. Click **Log in**.

After the successful login, you can start configuring the IdM server.



CHAPTER 6. LOGGING IN TO IDM IN THE WEB UI: USING A KERBEROS TICKET

The following sections describe the initial configuration of your environment to enable Kerberos login to the IdM Web UI and accessing IdM using Kerberos authentication.

Prerequisites

- Installed IdM server in your network environment
  For details, see Installing Identity Management in Red Hat Enterprise Linux 9

6.1. KERBEROS AUTHENTICATION IN IDENTITY MANAGEMENT

Identity Management (IdM) uses the Kerberos protocol to support single sign-on. Single sign-on authentication allows you to provide the correct user name and password only once, and you can then access Identity Management services without the system prompting for credentials again.

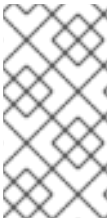
The IdM server provides Kerberos authentication immediately after the installation if the DNS and certificate settings have been configured properly. For details, see Installing Identity Management.

To use Kerberos authentication on hosts, install:

- the IdM client
  For details, see Preparing the system for Identity Management client installation .
- the krb5conf package

6.2. USING KINIT TO LOG IN TO IDM MANUALLY

This procedure describes using the **kinit** utility to authenticate to an Identity Management (IdM) environment manually. The **kinit** utility obtains and caches a Kerberos ticket-granting ticket (TGT) on behalf of an IdM user.



NOTE

Only use this procedure if you have destroyed your initial Kerberos TGT or if it has expired. As an IdM user, when logging onto your local machine you are also automatically logging in to IdM. This means that after logging in, you are not required to use the **kinit** utility to access IdM resources.

Procedure

1. To log in to IdM
   - under the user name of the user who is currently logged in on the local system, use **kinit** without specifying a user name. For example, if you are logged in as **example_user** on the local system:

```
[example_user@server ~]$ kinit
Password for example_user@EXAMPLE.COM:
[example_user@server ~]$
```

If the user name of the local user does not match any user entry in IdM, the authentication attempt fails:

```
[example_user@server ~]$ kinit
kinit: Client 'example_user@EXAMPLE.COM' not found in Kerberos database while
getting initial credentials
```

- using a Kerberos principal that does not correspond to your local user name, pass the required user name to the **kinit** utility. For example, to log in as the **admin** user:

```
[example_user@server ~]$ kinit admin
Password for admin@EXAMPLE.COM:
[example_user@server ~]$
```

2. Optionally, to verify that the login was successful, use the **klist** utility to display the cached TGT. In the following example, the cache contains a ticket for the **example_user** principal, which means that on this particular host, only **example_user** is currently allowed to access IdM services:

```
$ klist
Ticket cache: KEYRING:persistent:0:0
Default principal: example_user@EXAMPLE.COM

Valid starting      Expires             Service principal
11/10/2019 08:35:45   11/10/2019 18:35:45   krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

6.3. CONFIGURING THE BROWSER FOR KERBEROS AUTHENTICATION

To enable authentication with a Kerberos ticket, you may need a browser configuration.

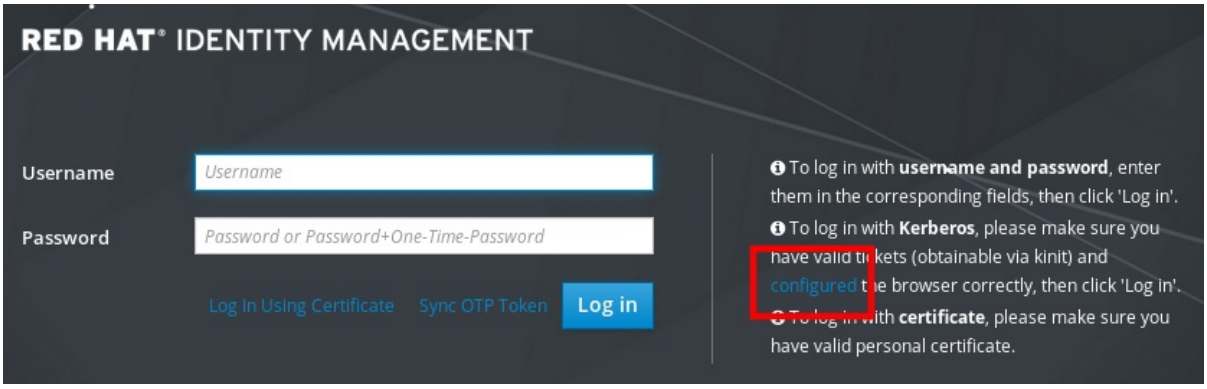
The following steps help you to support Kerberos negotiation for accessing the IdM domain.

Each browser supports Kerberos in a different way and needs different set up. The IdM Web UI includes guidelines for the following browsers:

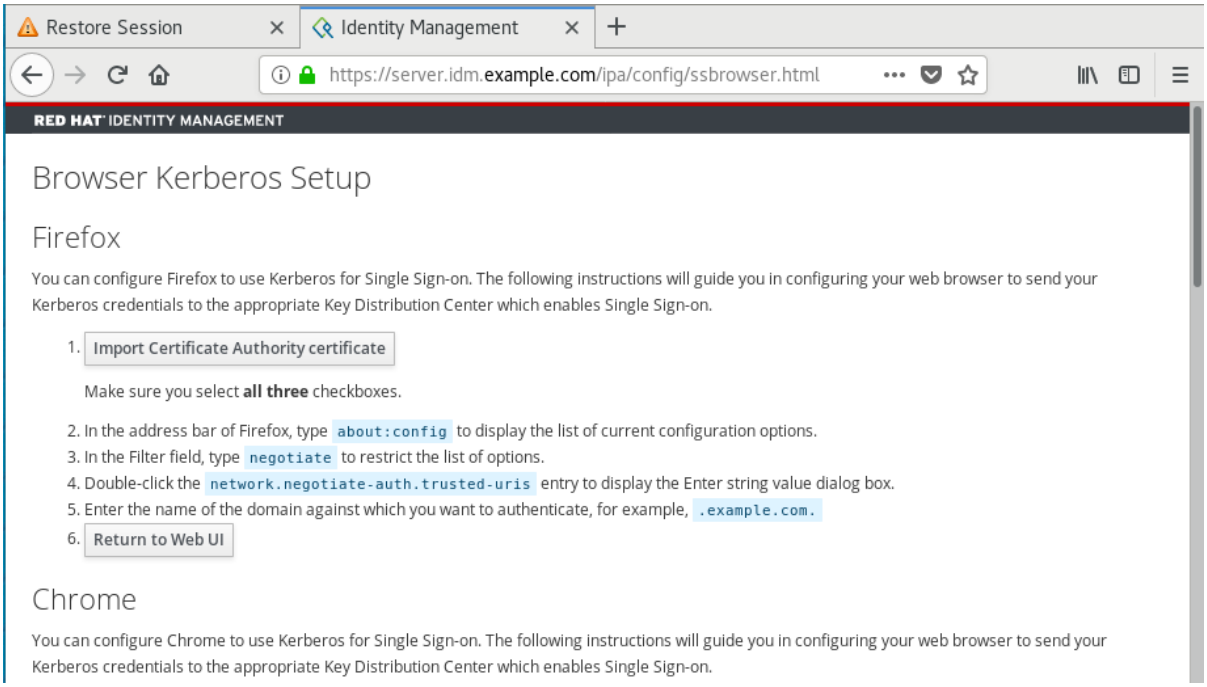
- Firefox
- Chrome

Procedure

1. Open the IdM Web UI login dialog in your web browser.
2. Click the link for browser configuration on the Web UI login screen.



3. Follow the steps on the configuration page.



After the setup, turn back to the IdM Web UI and click **Log in**.

6.4. LOGGING IN TO THE WEB UI USING A KERBEROS TICKET

This procedure describes logging in to the IdM Web UI using a Kerberos ticket-granting ticket (TGT).

The TGT expires at a predefined time. The default time interval is 24 hours and you can change it in the IdM Web UI.

After the time interval expires, you need to renew the ticket:

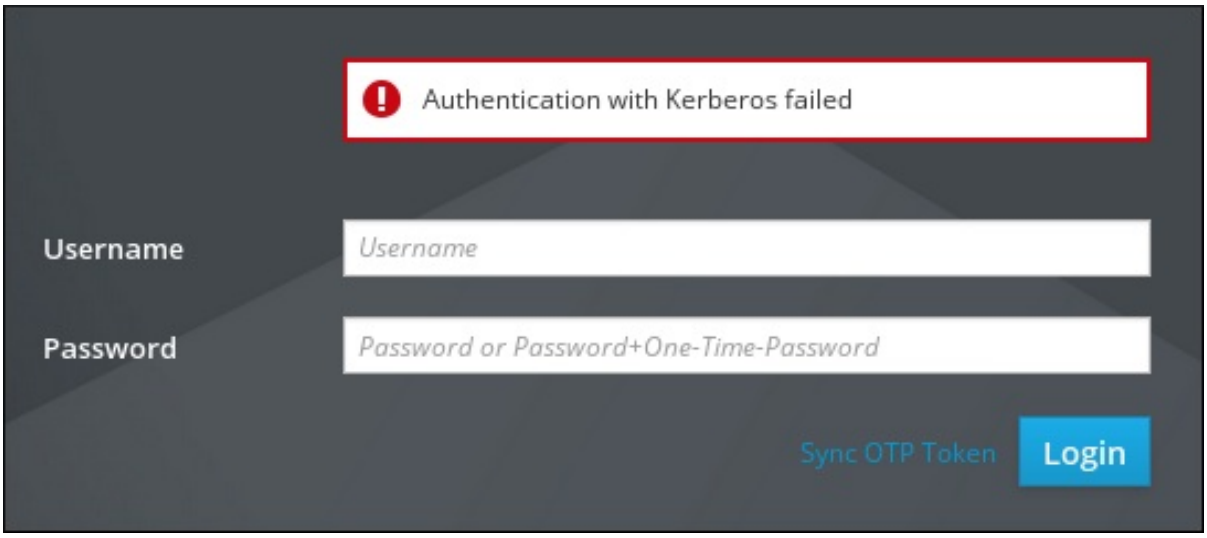
- Using the kinit command.
- Using IdM login credentials in the Web UI login dialog.

Procedure

- Open the IdM Web UI.
  If Kerberos authentication works correctly and you have a valid ticket, you will be automatically authenticated and the Web UI opens.

  If the ticket is expired, it is necessary to authenticate yourself with credentials first. However, next time the IdM Web UI will open automatically without opening the login dialog.

If you see an error message **Authentication with Kerberos failed**, verify that your browser is configured for Kerberos authentication. See Configuring the browser for Kerberos authentication.



6.5. CONFIGURING AN EXTERNAL SYSTEM FOR KERBEROS AUTHENTICATION

This section describes how to configure an external system so that Identity Management (IdM) users can log in to IdM from the external system using their Kerberos credentials.

Enabling Kerberos authentication on external systems is especially useful when your infrastructure includes multiple realms or overlapping domains. It is also useful if the system has not been enrolled into any IdM domain through **ipa-client-install**.

To enable Kerberos authentication to IdM from a system that is not a member of the IdM domain, define an IdM-specific Kerberos configuration file on the external system.

Prerequisites

- The **krb5-workstation** package is installed on the external system.
  To find out whether the package is installed, use the following CLI command:

```
# dnf list installed krb5-workstation
Installed Packages
krb5-workstation.x86_64    1.16.1-19.el8    @BaseOS
```

Procedure

1. Copy the **/etc/krb5.conf** file from the IdM server to the external system. For example:

```
# scp /etc/krb5.conf root@externalsystem.example.com:/etc/krb5_ipa.conf
```

WARNING

Do not overwrite the existing **krb5.conf** file on the external system.

2. On the external system, set the terminal session to use the copied IdM Kerberos configuration file:

```
$ export KRB5_CONFIG=/etc/krb5_ipa.conf
```

   The **KRB5_CONFIG** variable exists only temporarily until you log out. To prevent this loss, export the variable with a different file name.

3. Copy the Kerberos configuration snippets from the **/etc/krb5.conf.d/** directory to the external system.

4. Configure the browser on the external system, as described in Configuring the browser for Kerberos authentication.

Users on the external system can now use the **kinit** utility to authenticate against the IdM server.

6.6. WEB UI LOGIN FOR ACTIVE DIRECTORY USERS

To enable Web UI login for Active Directory users, define an ID override for each Active Directory user in the **Default Trust View**. For example:

```
[admin@server ~]$ ipa idoverrideuser-add 'Default Trust View' ad_user@ad.example.com
```

Additional resources

- Using ID views for Active Directory users

CHAPTER 7. LOGGING IN TO THE IDENTITY MANAGEMENT WEB UI USING ONE TIME PASSWORDS

Access to IdM Web UI can be secured using several methods. The basic one is password authentication.

To increase the security of password authentication, you can add a second step and require automatically generated one-time passwords (OTPs). The most common usage is to combine password connected with the user account and a time limited one time password generated by a hardware or software token.

The following sections help you to:

- Understand how the OTP authentication works in IdM.
- Configure OTP authentication on the IdM server.
- Create OTP tokens and synchronize them with the FreeOTP app in your phone.
- Authenticate to the IdM Web UI with the combination of user password and one time password.
- Re-synchronize tokens in the Web UI.

7.1. PREREQUISITES

- Accessing the IdM Web UI in a web browser

7.2. ONE TIME PASSWORD (OTP) AUTHENTICATION IN IDENTITY MANAGEMENT

One-time passwords bring an additional step to your authentication security. The authentication uses your password + an automatically generated one time password.

To generate one time passwords, you can use a hardware or software token. IdM supports both software and hardware tokens.

Identity Management supports the following two standard OTP mechanisms:

- The HMAC-Based One-Time Password (HOTP) algorithm is based on a counter. HMAC stands for Hashed Message Authentication Code.
- The Time-Based One-Time Password (TOTP) algorithm is an extension of HOTP to support time-based moving factor.

IMPORTANT

IdM does not support OTP logins for Active Directory trust users.

7.3. ENABLING THE ONE TIME PASSWORD IN THE WEB UI

The IdM Web UI allows you to configure hardware or software device to generate one-time passwords.

The one time password is entered just after the usual password in the dedicated field in the login dialog.

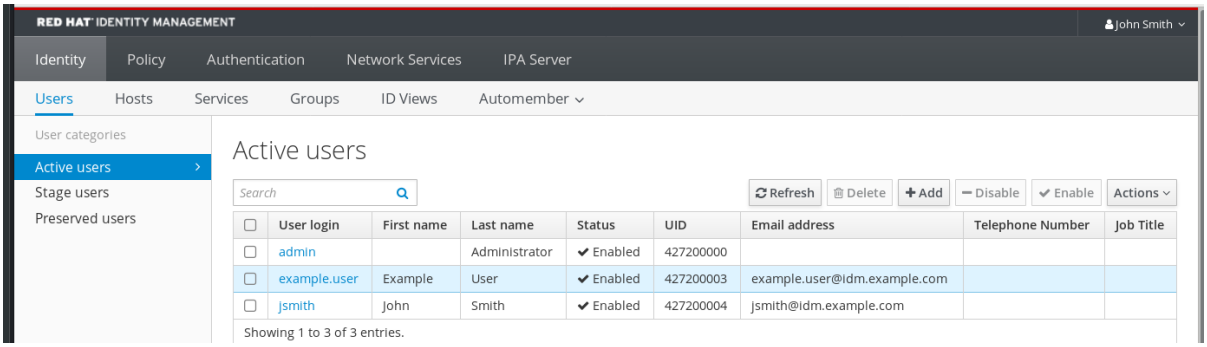
Only administrators can enable OTP authentication in the user settings.

Prerequisites

- Administration privileges

Procedure

1. Log in to the IdM Web UI with your username and password.
2. Open the **Identity → Users → Active users**tab.



3. Click your username to open the user settings.
4. In the **User authentication types**, select **Two factor authentication (password + OTP)**.
5. Click **Save**.

At this point, the OTP authentication is enabled on the IdM server.

Now you or users themselves need to assign a new token ID to the user account.

7.4. ADDING OTP TOKENS IN THE WEB UI

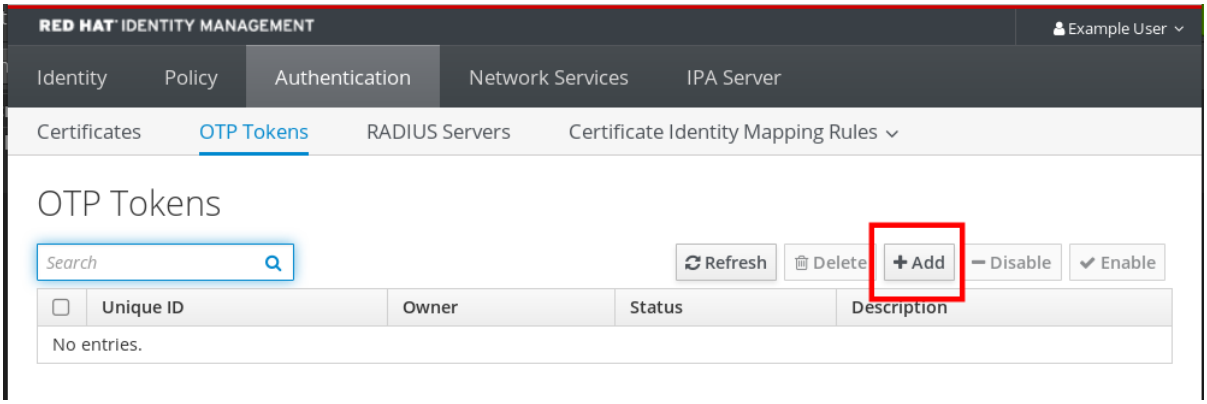
The following section helps you to add token to the IdM Web UI and to your software token generator.

Prerequisites

- Active user account on the IdM server.
- Administrator has enabled OTP for the particular user account in the IdM Web UI.
- A software device generating OTP tokens, for example FreeOTP.

Procedure

1. Log in to the IdM Web UI with your user name and password.
2. To create the token in your mobile phone, open the **Authentication → OTP Tokens**tab.
3. Click **Add**.

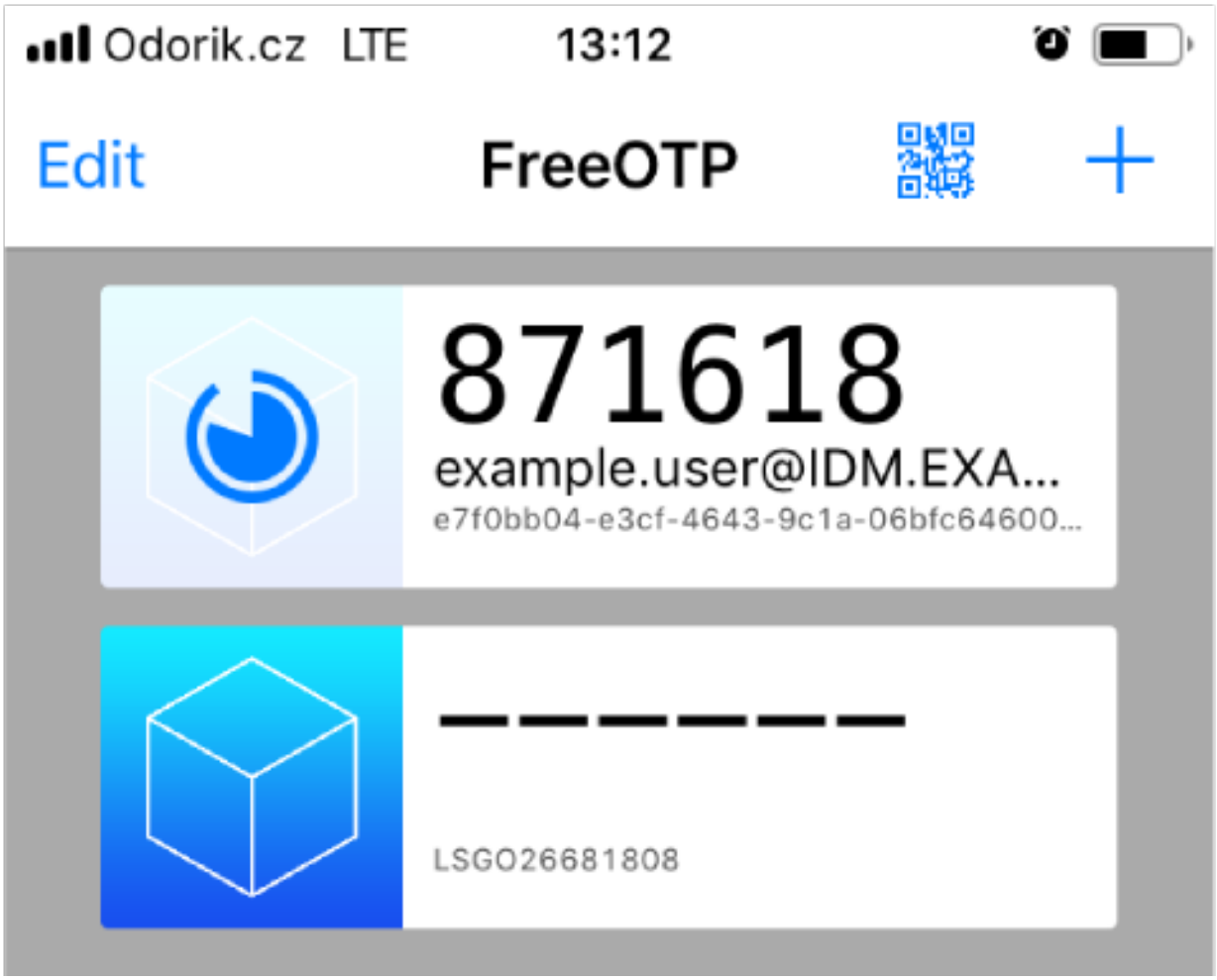


4. In the **Add OTP token** dialog box, leave everything unfilled and click **Add**.
   At this stage, the IdM server creates a token with default parameters at the server and opens a page with a QR code.

5. Copy the QR code into your mobile phone.

6. Click **OK** to close the QR code.

Now you can generate one time passwords and log in with them to the IdM Web UI.



7.5. LOGGING INTO THE WEB UI WITH A ONE TIME PASSWORD

This procedure describes the first login into the IdM Web UI using a one time password (OTP).

Prerequisites

Prerequisites

- OTP configuration enabled on the Identity Management server for the user account you are using for the OTP authentication. Administrators as well as users themselves can enable OTP. To enable the OTP configuration, see Enabling the one time password in the Web UI .
- A hardware or software device generating OTP tokens configured.

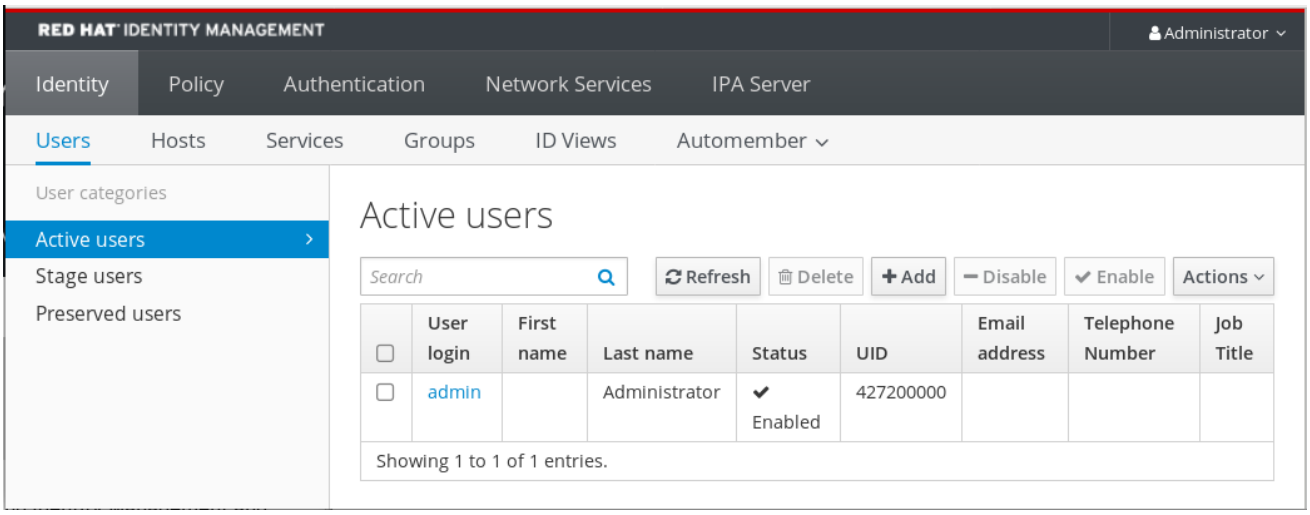
Procedure

1. In the Identity Management login screen, enter your user name or a user name of the IdM server administrator account.
2. Add the password for the user name entered above.
3. Generate a one time password on your device.
4. Enter the one time password right after the password (without space).
5. Click **Log in**.
   If the authentication fails, synchronize OTP tokens.

   If your CA uses a self-signed certificate, the browser issues a warning. Check the certificate and accept the security exception to proceed with the login.

   If the the IdM Web UI does not open, verify the DNS configuration of your Identity Management server.

After successful login, the IdM Web UI appears.



7.6. SYNCHRONIZING OTP TOKENS USING THE WEB UI

If the login with OTP (One Time Password) fails, OTP tokens are not synchronized correctly.

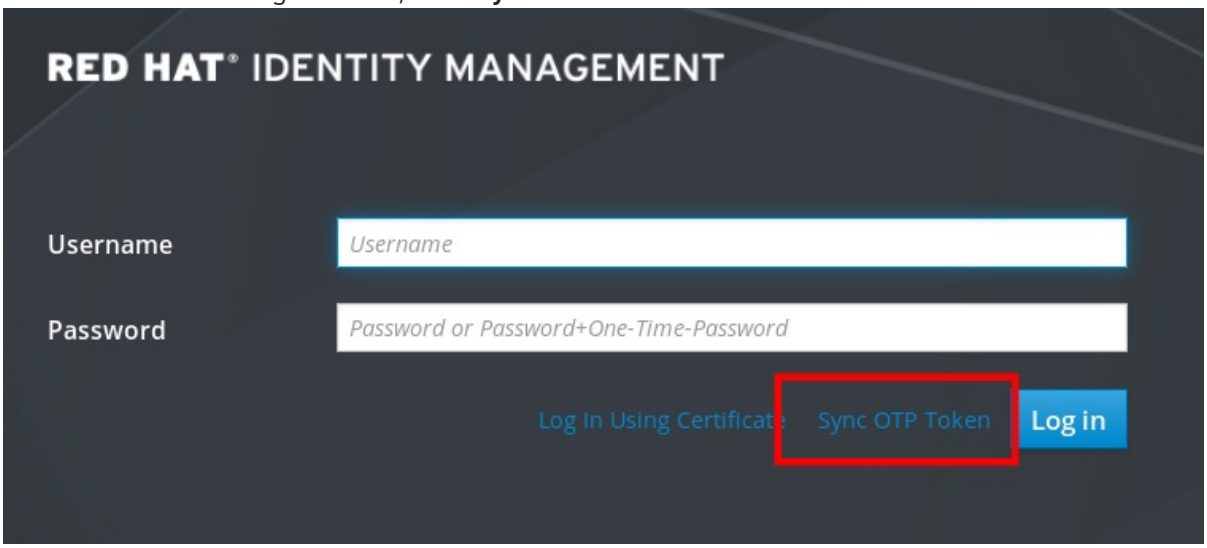
The following text describes token re-synchronization.

Prerequisites

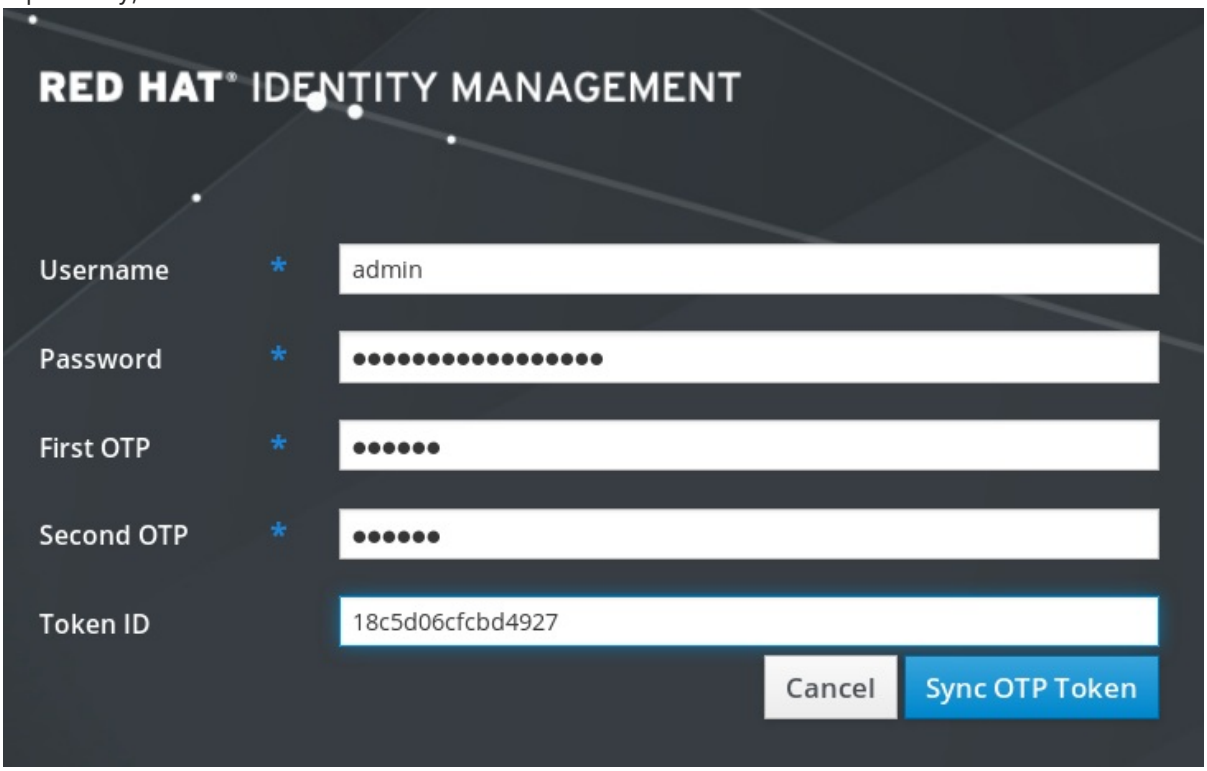
- A login screen opened.
- A device generating OTP tokens configured.

Procedure

1. On the IdM Web UI login screen, click **Sync OTP Token**.



2. In the login screen, enter your username and the Identity Management password.
3. Generate one time password and enter it in the **First OTP** field.
4. Generate another one time password and enter it in the **Second OTP** field.
5. Optionally, enter the token ID.



6. Click **Sync OTP Token**.

After the successful synchronization, you can log in to the IdM server.

7.7. CHANGING EXPIRED PASSWORDS

Administrators of Identity Management can enforce you having to change your password at the next login. It means that you cannot successfully log in to the IdM Web UI until you change the password.

Password expiration can happen during your first login to the Web UI.

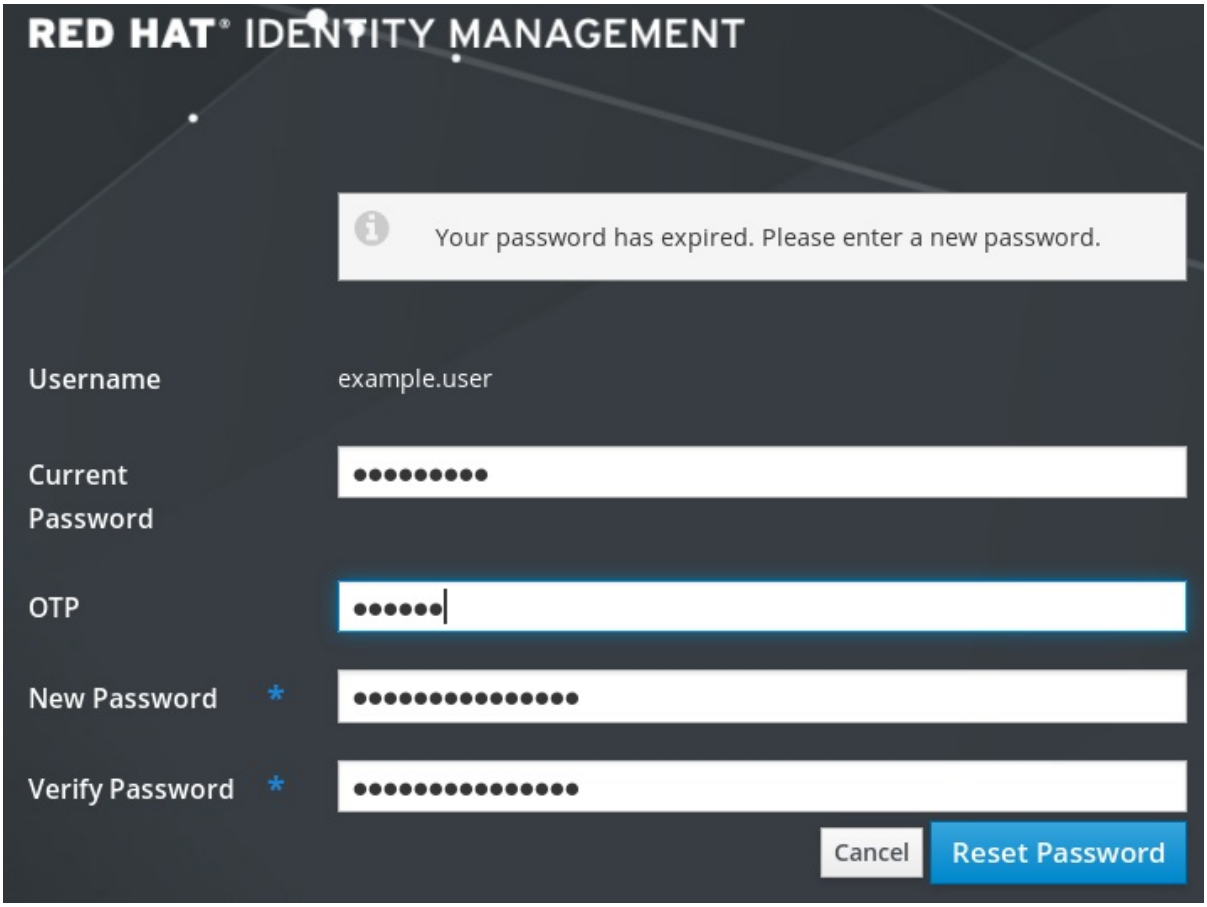
If the expiration password dialog appears, follow the instructions in the procedure.

Prerequisites

- A login screen opened.
- Active account to the IdM server.

Procedure

1. In the password expiration login screen, enter the user name.
2. Add the password for the user name entered above.
3. In the OTP field, generate a one time password, if you use the one time password authentication.
   If you do not have enabled the OTP authentication, leave the field empty.
4. Enter the new password twice for verification.
5. Click **Reset Password**.



After the successful password change, the usual login dialog displays. Log in with the new password.

CHAPTER 8. IDENTITY MANAGEMENT SECURITY SETTINGS

This section describes security-related features of Identity Management.

8.1. HOW IDENTITY MANAGEMENT APPLIES DEFAULT SECURITY SETTINGS

By default, Identity Management (IdM) uses the system-wide crypto policy. The benefit of this policy is that you do not need to harden individual IdM components manually.

IMPORTANT

Red Hat recommends that you use the system-wide crypto policy. Changing individual security settings can break components of IdM.

Additional resources

- See the **crypto-policies(7)** man page.

8.2. ANONYMOUS LDAP BINDS IN IDENTITY MANAGEMENT

By default, anonymous binds to the Identity Management (IdM) LDAP server are enabled. Anonymous binds can expose certain configuration settings or directory values. However, some utilities, such as **realmd**, or older RHEL clients require anonymous binds enabled to discover domain settings when enrolling a client.

Additional resources

- Disabling anonymous binds

8.3. DISABLING ANONYMOUS BINDS

You can disable anonymous binds on the Identity Management (IdM) 389 Directory Server instance by using LDAP tools to reset the **nsslapd-allow-anonymous-access** attribute.

These are the valid values for the **nsslapd-allow-anonymous-access** attribute:

- **on**: allows all anonymous binds (default)
- **rootdse**: allows anonymous binds only for root DSE information
- **off**: disallows any anonymous binds

Red Hat does not recommend completely disallowing anonymous binds by setting the attribute to **off**, because this also blocks external clients from checking the server configuration. LDAP and web clients are not necessarily domain clients, so they connect anonymously to read the root DSE file to get connection information.

By changing the value of the **nsslapd-allow-anonymous-access** attribute to **rootdse**, you allow access to the root DSE and server configuration without any access to the directory data.

WARNING

Certain clients rely on anonymous binds to discover IdM settings. Additionally, the compat tree can break for legacy clients that are not using authentication. Perform this procedure only if your clients do not require anonymous binds.

Prerequisites

- You can authenticate as the Directory Manager to write to the LDAP server.
- You can authenticate as the **root** user to restart IdM services.

Procedure

1. Change the **nsslapd-allow-anonymous-access** attribute to **rootdse**.

```
$ ldapmodify -x -D "cn=Directory Manager" -W -h server.example.com -p 389
Enter LDAP Password:
dn: cn=config
changetype: modify
replace: nsslapd-allow-anonymous-access
nsslapd-allow-anonymous-access: rootdse

modifying entry "cn=config"
```

2. Restart the 389 Directory Server instance to load the new setting.

```
# systemctl restart dirsrv.target
```

Verification

- Display the value of the **nsslapd-allow-anonymous-access** attribute.

```
$ ldapsearch -x -D "cn=Directory Manager" -b cn=config -W -h server.example.com  -p 389
nsslapd-allow-anonymous-access | grep nsslapd-allow-anonymous-access
Enter LDAP Password:
# requesting: nsslapd-allow-anonymous-access
nsslapd-allow-anonymous-access: rootdse
```

Additional resources

- nsslapd-allow-anonymous-access in Directory Server 11 documentation
- Anonymous LDAP binds in Identity Management

CHAPTER 9. IDM LOG FILES AND DIRECTORIES

Use the following sections to monitor, analyze, and troubleshoot the individual components of Identity Management (IdM):

- LDAP
- Apache web server
- Certificate system
- Kerberos
- DNS
- Custodia

Additionally, you can monitor, analyze, and troubleshoot the IdM server and client and enable audit logging on an IdM server.

9.1. IDM SERVER AND CLIENT LOG FILES AND DIRECTORIES

The following table presents directories and files that the Identity Management (IdM) server and client use to log information. You can use the files and directories for troubleshooting installation errors.

Directory or File	Description
/var/log/ipaserver-install.log	The installation log for the IdM server.
/var/log/ipareplica-install.log	The installation log for the IdM replica.
/var/log/ipaclient-install.log	The installation log for the IdM client.
/var/log/sssd/	Log files for SSSD. You can enable detailed logging for SSSD in the sssd.conf file or with the sssctl command.
~/.ipa/log/cli.log	The log file for errors returned by remote procedure calls (RPCs) and responses by the **ipa** utility. Created in the home directory for the **effective user** that runs the tools. This user might have a different user name than the IdM user principal, that is the IdM user whose ticket granting ticket (TGT) has been obtained before attempting to perform the failed **ipa** commands. For example, if you are logged in to the system as **root** and have obtained the TGT of IdM**admin**, then the errors are logged in to the **/root/.ipa/log/cli.log** file.
/etc/logrotate.d/	The log rotation policies for DNS, SSSD, Apache, Tomcat, and Kerberos.
/etc/pki/pki-tomcat/logging.properties	This link points to the default Certificate Authority logging configuration at **/usr/share/pki/server/conf/logging.properties**.

Additional resources

- Troubleshooting IdM server installation
- Troubleshooting IdM client installation
- Troubleshooting IdM replica installation
- Troubleshooting authentication with SSSD in IdM

9.2. DIRECTORY SERVER LOG FILES

The following table presents directories and files that the Identity Management (IdM) Directory Server (DS) instance uses to log information. You can use the files and directories for troubleshooting DS-related problems.

Table 9.1. Directory Server log files

Directory or file	Description
/var/log/dirsrv/slapd-*REALM_NAME*/	Log files associated with the DS instance used by the IdM server. Most operational data recorded here are related to server-replica interactions.
/var/log/dirsrv/slapd-*REALM_NAME*/audit	Contains audit trails of all DS operations when auditing is enabled in the DS configuration.
/var/log/dirsrv/slapd-*REALM_NAME*/access	Contain detailed information about attempted access and failed operations for the domain DS instance.
/var/log/dirsrv/slapd-*REALM_NAME*/errors	

Additional resources

- Monitoring Server and Database Activity
- Log File Reference

9.3. ENABLING AUDIT LOGGING ON AN IDM SERVER

This procedure describes how to enable logging on an Identity Management (IdM) server for audit purposes. Using detailed logs, you can monitor data, troubleshoot issues, and examine suspicious activity on the network.

NOTE

The LDAP service may become slower if there are many LDAP changes logged, especially if the values are large.

Prerequisites

- The Directory Manager password

Procedure

Procedure

1. Bind to the LDAP server:

```
$ ldapmodify -D "cn=Directory Manager" -W << EOF
```

2. Press [Enter].

3. Specify all the modifications you want to make, for example:

```
dn: cn=config
changetype: modify
replace: nsslapd-auditlog-logging-enabled
nsslapd-auditlog-logging-enabled: on
-
replace:nsslapd-auditlog
nsslapd-auditlog: /var/log/dirsrv/slapd-REALM_NAME/audit
-
replace:nsslapd-auditlog-mode
nsslapd-auditlog-mode: 600
-
replace:nsslapd-auditlog-maxlogsize
nsslapd-auditlog-maxlogsize: 100
-
replace:nsslapd-auditlog-logrotationtime
nsslapd-auditlog-logrotationtime: 1
-
replace:nsslapd-auditlog-logrotationtimeunit
nsslapd-auditlog-logrotationtimeunit: day
```

4. Indicate the end of the **ldapmodify** command by entering **EOF** on a new line.

5. Press [Enter] twice.

6. Repeat the previous steps on all the other IdM servers on which you want to enable audit logging.

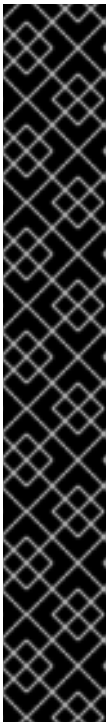
Verification

- Open the **/var/log/dirsrv/slapd-REALM_NAME/audit** file:

```
389-Directory/1.4.3.231 B2021.322.1803
server.idm.example.com:636 (/etc/dirsrv/slapd-IDM-EXAMPLE-COM)

time: 20220607102705
dn: cn=config
result: 0
changetype: modify
replace: nsslapd-auditlog-logging-enabled
nsslapd-auditlog-logging-enabled: on
[...]
```

  The fact that the file is not empty anymore confirms that auditing is enabled.



IMPORTANT

The system logs the bound LDAP distinguished name (DN) of the entry that makes a change. For this reason, you might have to post-process the log. For example, in the IdM Directory Server, it is an ID override DN that represents the identity of an AD user that modified a record:

```
$ modifiersName: ipaanchoruuid=:sid:s-1-5-21-19610888-1443184010-1631745340-279100,cn=default trust view,cn=views,cn=accounts,dc=idma,dc=idm,dc=example,dc=com
```

Use the **pysss_nss_idmap.getnamebysid** Python command to look up an AD user if you have the user SID:

```
>>> import pysss_nss_idmap
>>> pysss_nss_idmap.getnamebysid('S-1-5-21-1273159419-3736181166-4190138427-500'))
{'S-1-5-21-1273159419-3736181166-4190138427-500': {'name': 'administrator@ad.vm', 'type': 3}}
```

Additional resources

- Directory Server log files
- The How to enable Audit logging in IPA/IDM Server and Replica Servers KCS solution

9.4. THE IDM APACHE SERVER LOG FILES

The following table presents directories and files that the Identity Management (IdM) Apache Server uses to log information.

Table 9.2. Apache Server log files

Directory or File	Description
/var/log/httpd/	Log files for the Apache web server.
/var/log/httpd/access_log	Standard access and error logs for Apache servers. Messages specific to IdM are recorded along with the Apache messages because the IdM web UI and the RPC command-line interface use Apache. The access logs log mostly only the user principal and the URI used, which is often an RPC endpoint. The error logs contain the IdM server logs.
/var/log/httpd/error_log	

Additional resources

- Log Files in the Apache documentation

9.5. CERTIFICATE SYSTEM LOG FILES IN IDM

The following table presents directories and files that the Identity Management (IdM) Certificate System uses to log information.

Table 9.3. Certificate System log files

Directory or File	Description
/var/log/pki/pki-ca-spawn.*time_of_installation*.log	The installation log for the IdM certificate authority (CA).
/var/log/pki/pki-kra-spawn.*time_of_installation*.log	The installation log for the IdM Key Recovery Authority (KRA).
/var/log/pki/pki-tomcat/	The top level directory for PKI operation logs. Contains CA and KRA logs.
/var/log/pki/pki-tomcat/ca/	Directory with logs related to certificate operations. In IdM, these logs are used for service principals, hosts, and other entities which use certificates.
/var/log/pki/pki-tomcat/kra	Directory with logs related to KRA.
/var/log/messages	Includes certificate error messages among other system messages.

Additional resources

- Configuring subsystem logs in the Red Hat Certificate System *Administration Guide*

9.6. KERBEROS LOG FILES IN IDM

The following table presents directories and files that Kerberos uses to log information in Identity Management (IdM).

Table 9.4. Kerberos Log Files

Directory or File	Description
/var/log/krb5kdc.log	The primary log file for the Kerberos KDC server.
/var/log/kadmind.log	The primary log file for the Kerberos administration server.
Locations for these files are configured in the **krb5.conf** file. They can be different on some systems.	

9.7. DNS LOG FILES IN IDM

The following table presents directories and files that DNS uses to log information in Identity Management (IdM).

Table 9.5. DNS log files

Directory or File	Description
/var/log/messages	Includes DNS error messages and other system messages. DNS logging in this file is not enabled by default. To enable it, enter the **# /usr/sbin/rndc querylog** command. The command results in the following lines being added to **var/log/messages**: **Jun 26 17:37:33 r8server named-pkcs11[1445]: received control channel command 'querylog'** **Jun 26 17:37:33 r8server named-pkcs11[1445]: query logging is now on** To disable logging, run the command again.

9.8. CUSTODIA LOG FILES IN IDM

The following table presents directories and files that Custodia uses to log information in Identity Management (IdM).

Table 9.6. Custodia Log Files

Directory or File	Description
/var/log/custodia/	Log file directory for the Custodia service.

9.9. ADDITIONAL RESOURCES

- Viewing Log Files. You can use **journalctl** to view the logging output of **systemd** unit files.