

Identity Management

Date published: 2019-08-22

Date modified:



Legal Notice

© Cloudera Inc. 2024. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

Contents

Managing FreeIPA.....	4
Creating an environment configured with FreeIPA.....	5
Show FreeIPA instance status.....	6
FreeIPA instance status options.....	8
Vertically scale FreeIPA instances.....	8
Upgrade FreeIPA.....	10
Repair FreeIPA.....	13
Check FreeIPA repair status.....	14
Resize FreeIPA.....	15
Configuring workload password policies.....	16
Check the current workload password policy.....	17
Set a password policy.....	18
Reset a password policy.....	20
Unset user's minimum password lifetime.....	21
Accessing a FreeIPA cluster via SSH.....	21
Recipes.....	22
Writing recipes.....	23
Register a recipe.....	24
Update a FreeIPA recipe.....	26
Managing recipes from CLI.....	26

Managing FreeIPA

FreeIPA is the backbone of the CDP Identity Management functionality. After you configure a CDP environment, FreeIPA works to provide user identities without the need for your attention. In case of problems, you may need to perform troubleshooting to ensure the health of the identity management system.

FreeIPA availability types

Historically, FreeIPA node count (also known as "availability types") can be one of the following:

- 3 nodes (HA)
- 2 nodes
- 1 node

When registering a CDP environment via web UI, you cannot select FreeIPA node count, but CDP adjusts FreeIPA based on the Data Lake scale that you select:

- For an Enterprise Data Lake, a FreeIPA cluster with 3 nodes (HA) is provisioned
- For a Medium Duty Data Lake, a FreeIPA cluster with 3 nodes (HA) is provisioned
- For a Light Duty Data Lake, a FreeIPA cluster with 2 nodes is provisioned



Attention: If you scale an existing Data Lake, for example scaling from light duty to medium duty, the FreeIPA node count is unaffected.

When registering a CDP environment via CDP CLI, you can select the node count manually. You can choose to create 1+ (up to a maximum of 3). If you do not specify the node count, CDP automatically configures FreeIPA with one node only.

When HA is configured, this mode allows automatic failover should one FreeIPA instance fail and a scripted manual process to recover the system with no downtime should it fail.

The CDP environment backs up the FreeIPA state periodically (by default, hourly). The backup data is stored on an attached volume (AWS) or managed disk (Azure). This backup allows the state to be recovered in the event of a failure. Without HA mode enabled, recovering from a FreeIPA failure requires a recovery process that is facilitated by Cloudera technical support.

For each running environment, the host and status of the FreeIPA instance is displayed in the environment's Summary tab in the Management Console.

FreeIPA HA

By default, CDP creates multiple FreeIPA instances and replicates identity management data across all of them. Should there be a conflict synchronizing across instances, the system maintains the "last in" content. If one of the FreeIPA instances fails to pass the environment's status checks, the overall status for FreeIPA turns gray. The FreeIPA clients switch to another FreeIPA instance and the system remains functional. After a week in this state, the identity management system may start to fail from certificates expiring and other problems.

When you see a status other than "Running", follow these steps to investigate and resolve problems:

1. Determine which FreeIPA instance or instances need attention.

To retrieve a detailed status, use the CDP command-line interface. See "Show FreeIPA instance status".

2. Repair one or more FreeIPA instances.

The CDP command-line interface includes a command to trigger a FreeIPA check and reboot repair process. The repair command should resolve most problems with the identity management system. For example, it checks to see if FreeIPA hosts are stopped and restarts them. If hosts are running, the repair process will restart them.

The repair command may cause cluster outage:

- If at least one instance is running, the repair command can avoid an outage.
- If all instances have failed, the repair command won't be able to avoid an outage.
- If the repair command is run with its "force" option against all instances, the command will cause an outage.

If the FreeIPA status for the environment returns to "Running", you can stop here.

3. Rebuild the identity management system.

When repairing a FreeIPA instance, the CDP command-line interface includes a command to trigger a rebuild process that destroys and rebuilds N-1 nodes in the identity management system. This process restores content from the most recent backup and does not require cluster downtime.

If all nodes need to be repaired or if the unsupervised rebuild process fails, Cloudera Technical Support can help you perform a rebuild of the identity management system and restore content from a backup. This process will require cluster downtime.

FreeIPA failure scenarios

Because FreeIPA is a background system, you are not likely to encounter any failures that include a specific reference to FreeIPA in the error text. Instead, problems with FreeIPA show up as DNS problems, user login problems that raise Kerberos errors, and authentication errors when provisioning workload clusters. If you encounter these general errors, consider checking the status of the FreeIPA system.

Related Information

[Creating an environment configured with FreeIPA](#)

[Show FreeIPA instance status](#)

[Upgrade FreeIPA](#)

[Repair FreeIPA](#)

[Resize FreeIPA](#)

[Configuring workload password policies](#)

[Accessing a FreeIPA cluster via SSH](#)

[CDP identity management](#)

Creating an environment configured with FreeIPA

High-availability FreeIPA is automatically enabled when you register an environment via CDP web UI or CLI. The CDP CLI provides environment creation commands that include an option for setting multiple FreeIPA instances, which triggers the system to set up the identity management cluster.

FreeIPA HA cannot be added to an existing environment, only configured during environment creation.

Configuring FreeIPA node count via UI

When registering a CDP environment via web UI, you cannot select FreeIPA node count, but CDP adjusts FreeIPA based on the Data Lake scale that you select:

- For an Enterprise Data Lake, a FreeIPA cluster with 3 nodes (HA) is provisioned
- For a Medium Duty Data Lake, a FreeIPA cluster with 3 nodes (HA) is provisioned
- For a Light Duty Data Lake, a FreeIPA cluster with 2 nodes is provisioned

Configuring FreeIPA node count via CLI

When registering an environment using CDP CLI, CDP configures FreeIPA with 1 node by default. For production, you should change this setting to 3 nodes. To create an environment with selected FreeIPA node count via CLI:

1. Run the CDP CLI command to create an environment and include an additional parameter in the JSON-formatted command input:

```
"freeIpa": { "instanceCountByGroup": <SPECIFY-NODE-COUNT> }
```

Or pass the following parameter:

```
--free-ipa instanceCountByGroup=<SPECIFY-NODE-COUNT>
```

replacing *<SPECIFY-NODE-COUNT>* with the number of instances of FreeIPA you want. The maximum number of instances is 3. Choose 2 or 3 based on the level of redundancy you want. The recommended setting is 3 nodes.

2. After successfully registering the environment, continue to create the Data Lake as described in the cloud-provider specific instructions.

Related Information

[Registering an AWS environment](#)

[Registering an Azure environment](#)

[Show FreeIPA instance status](#)

[Installing CDP client](#)

Show FreeIPA instance status

To see the status of each FreeIPA instance in an environment and determine which one(s) need to be repaired, run the CDP CLI `get-freeipa-status` command.

Steps

For CDP UI

The host and status of the FreeIPA instances is displayed in the environment's **FreeIPA Nodes** tab in the Management Console. If FreeIPA HA is enabled, tab will show status for all hosts:

The screenshot shows the 'FreeIPA' tab in the CDP UI. At the top, there are tabs for 'Data Hubs', 'Data Lake', 'FreeIPA' (selected), 'Cluster Definitions', and 'Summary'. Below the tabs, there are buttons for 'Get FreeIPA Certificate', 'Synchronize Users', and 'Retry'. The main content area is titled 'FreeIPA Details' and shows a 'STATUS' of 'Running' with a green checkmark, a 'STATUS REASON' of 'FreeIPA is Available', and a 'FREEIPA CRN' field. To the right, there are 'RECIPES' (0) and 'NODES' (2) counts, and an 'UPGRADE STATUS' of 'Upgrade Available'. Below this, there are tabs for 'Nodes', 'Recipes', and 'Upgrade'. The 'Nodes' tab is selected, showing a table of instances. A 'Vertical Scaling' button is visible on the right side of the table.

Instance ID	Status	FQDN	Private IP	Public IP
> i-06fe41577ebcb0fb5	Running	ipaserver1.go01-dem.ylcu-atmi.cloudera.site	10.10.134.104	
> i-0fba929a1b3c896bc	Running	ipaserver0.go01-dem.ylcu-atmi.cloudera.site	10.10.147.202	

The status shows the general health of the system and the cloud-provider ID of the host. If shown, the ID links to the cloud-provider's page for the host. A full list of status options can be found in [FreeIPA instance status options](#).

For CDP CLI

Run the FreeIPA status command. Run this command from a computer that has network access to the FreeIPA hosts

```
cdp environments get-freeipa-status --environment-name <value>
                                   [--cli-input-json <value>]
                                   [--generate-cli-skeleton]
```

where the options are the following:

Option	Description
--environment-name <value>	Specifies the FreeIPA's environment name or CRN. The environment CRN is listed in Environment Summary General .

This command can take 15 to 45 seconds to run as it gathers information in real-time. The output of the status command provides the status for each FreeIPA node associated with the environment (in JSON format):

```
{
  "environmentCrn": "crn:cdp:environments:us-west-1:12a0079b-1591-dd33-b
721-a446bda74e67:environment:36853fcc-2fef-4094-834c-557b4aea34ee",
  "environmentName": "finance-reporting-set4",
  "status": "AVAILABLE",
  "instances": {
    "i-078ba50f9feb6638f": {
      "hostname": "ipaserver1.datalake.xcu2-8y8x.cloudera.site",
      "status": "CREATED",
      "issues": []
    },
    "i-09e8b54a343b33d2": {
      "hostname": "ipaserver0.datalake.xcu2-8y8x.cloudera.site",
      "status": "CREATED",
      "issues": []
    }
  }
}
```

Element	Data Type	Description
environmentCrn	string	The CRN of the environment.
environmentName	string	The name of the environment.
status	string	The overall status of the FreeIPA cluster. The status values retrieved can reflect an earlier status from the cloud provider. For example, AWS may show the host as "Running," but the CDP CLI status command may show the status as "UNREACHABLE." When this occurs, it is a temporary status. Calling the CDP CLI status command again; eventually the status will change from "UNREACHABLE" to "CREATED," which is the CDP equivalent to AWS "Running." A full list of status values can be found in "FreeIPA instance status options".
instances	map	Status of individual nodes in the FreeIPA cluster.
key	string	Each entry includes the cloud-provider ID for the host. The corresponding value includes the status and issues of an individual FreeIPA node.
hostname	string	The hostname of the IPA instance.
issues	array	A list of issues the node is having. If there are no issues, the content is a blank string.

Related Information

[Repair FreeIPA](#)

[FreeIPA instance status options](#)

[Installing CDP client](#)

FreeIPA instance status options

A full list of FreeIPA instance status options for UI and CLI. This information is displayed in the environment's Summary tab in the Management Console, or via the `cdp environments get-freeipa-status` command.

Scenario	UI Status	CLI Overall Status	CLI Instance Status
Provisioning	Blue "Requested", then blue "Create in progress"	REQUESTED then CREATE_IN_PROGRESS then STACK_AVAILABLE then CREATE_IN_PROGRESS	CREATE_IN_PROGRESS
Provisioning failed	Red "Create failed"	CREATE_FAILED	
Running and all nodes have good health	Green "Running"	AVAILABLE	CREATED
Repairing	Blue "Update requested", then blue "Update in progress"	UPDATE_REQUESTED then UPDATE_IN_PROGRESS	At the end the new instance is CREATED and the old instance is TERMINATED
Repair Failed	Orange "Unhealthy"	UNHEALTHY	
Starting	Blue "Start in progress"	START_IN_PROGRESS	
Start Failed	Red "Start failed"	START_FAILED	
Stopping	Blue "Stop requested", then blue "Stop in progress"	STOP_REQUESTED then STOP_IN_PROGRESS	
Stop Failed	Red "Stop Failed"	STOP_FAILED	
Stopped	Gray "Stopped"	STOPPED	STOPPED
Deleted	Red "Terminated"	DELETE_COMPLETED	TERMINATED
Delete Failed	Red "Termination failed"	DELETE_FAILED	
Deleting	Red "Terminating"	DELETE_IN_PROGRESS	
Instance deleted on cloud provider's side	Orange "Unhealthy"	UNHEALTHY	DELETED_ON_PROVIDER_SIDE
All nodes deleted	Red "Deleted on provider side"	DELETED_ON_PROVIDER_SIDE	DELETED_ON_PROVIDER_SIDE
Network connectivity lost to some nodes but not	Orange "Unhealthy"	UNHEALTHY	UNREACHABLE
Network connectivity lost to all nodes	Orange "Unreachable"	UNREACHABLE	UNREACHABLE
Some nodes stopped on cloud provider side but not all	Orange "Unhealthy"	UNHEALTHY	STOPPED
All nodes stopped on cloud provider side	Gray "Stopped"	STOPPED	STOPPED
Some nodes with running with health check failures	Orange "Unhealthy"	UNHEALTHY	Bad nodes are UNHEALTHY
All nodes unhealthy	Orange "Unhealthy"	UNHEALTHY	UNHEALTHY

Vertically scale FreeIPA instances

If necessary, you can select a larger or smaller instance type for FreeIPA after the environment has been created.

About this task

Selecting a larger instance type adds more vCPU and/or RAM to your instances. Instances can be scaled both up and down, but scaling down to a smaller size requires 4 CPU and a minimum of 4 GB memory.

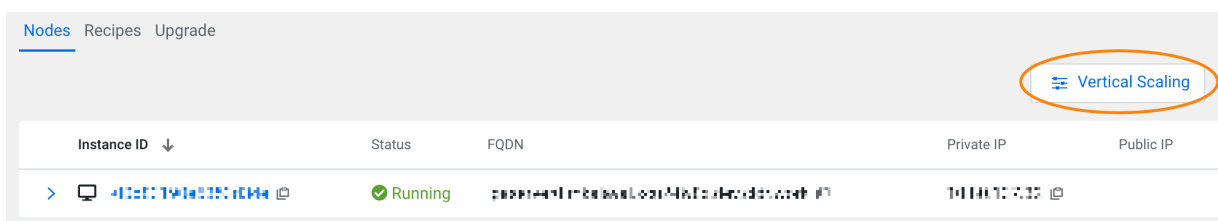
If you are using an instance without ephemeral disks, you can scale up or down to a new instance with ephemeral disks; however, the reverse is not supported. You cannot start with an instance with ephemeral disks and move to an instance without ephemeral disks.

Vertical scaling is supported on AWS. It is also supported on Azure by entitlement. Contact Cloudera customer support for entitlements.

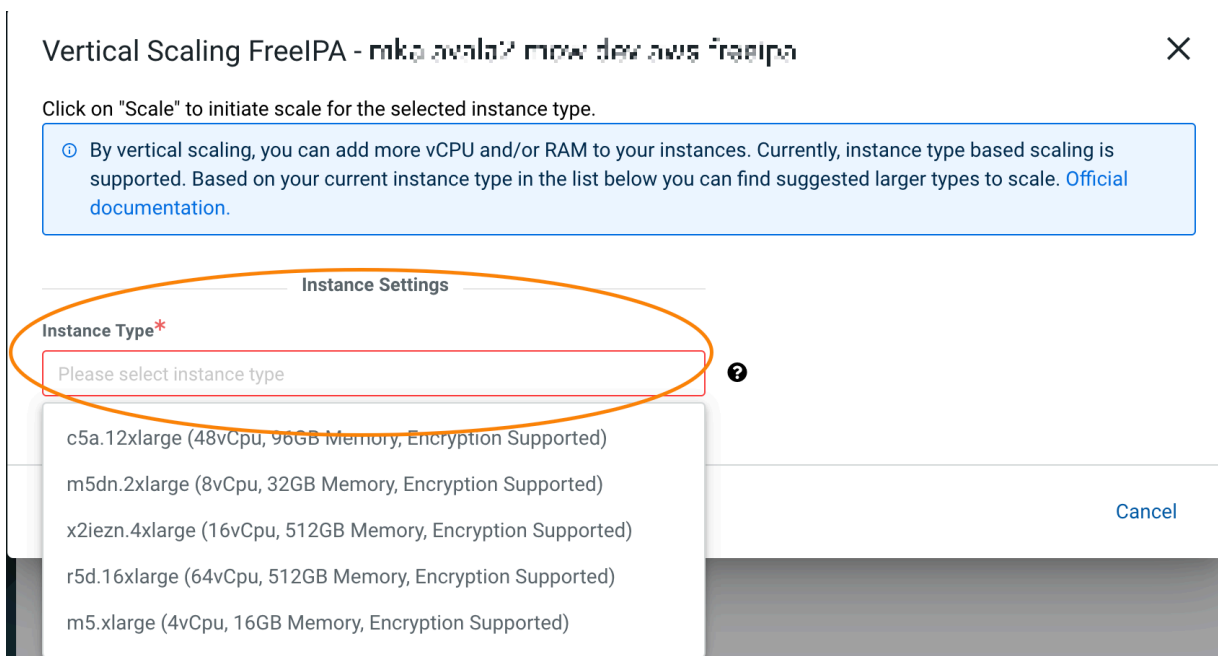
FreeIPA instances must be stopped before scaling. See [Change the instance type](#) in AWS documentation for more information.

Procedure

1. In the CDP main navigation menu, click Environments and select the environment that requires a larger FreeIPA instance type.
2. Scroll to the bottom of the page and click the **FreeIPA** tab.
3. Click Vertical Scaling.



4. From the drop-down menu of available instance types, select the instance that you want to scale to.



5. Click Scale. You can monitor the action from the **Event History** tab.
Alternatively, you can use the CDP CLI to select a new instance for FreeIPA:

```
cdp environments start-free-ipa-vertical-scaling
--environment <your-environment-name-or-its-crn>
--instance-template instanceType="<m5.4xlarge>"
```

Upgrade FreeIPA

To ensure that your FreeIPA nodes are running with the latest patches, you should periodically upgrade your FreeIPA cluster.

FreeIPA is the core component of an environment that runs on one or more instances.

To ensure that your FreeIPA nodes are running with the latest patches, you should periodically upgrade your FreeIPA cluster. CDP currently allows you to upgrade all FreeIPA clusters, updating OS-level security patches on the cluster nodes. The upgrade process includes launching entirely new instances with the new OS image.

The upgrade process is a rolling upgrade consisting of the following steps:

1. A new instance is provisioned from a newer image.
2. CDP validates that the new node is reachable, all services are up, and that replication works.
3. The old instance is removed.
4. If there is more than one instance in the cluster, the previous 3 steps are repeated for the second instance only after the first instance has been successfully upgraded.

Your environment remains functional during the upgrade process, with an exception of a short (one minute or less) downtime. During and after the FreeIPA upgrade, the status of the environment changes to **Update Started** as the environment is refreshing all the clusters to the new FreeIPA configurations (IP addresses for DNS resolution). Refreshing the clusters in the environment can take some time. You can minimize the time of the FreeIPA upgrade by not initiating any cluster related activities such as scaling and provisioning.

During FreeIPA provisioning a full backup is performed and there are periodic incremental backups afterwards. These backups can be used in case a recovery is needed.

Use either the CDP web interface or CDP CLI to initiate an upgrade. The upgrade process takes about one hour; The duration varies depending on the cloud provider and the number of nodes.

Performing a FreeIPA upgrade

You can trigger the FreeIPA upgrade via CDP web interface or CDP CLI.

Prerequisites

- If the upgrade involves upgrading from CentOS to RHEL, review the [Prerequisites for upgrading from CentOS to RHEL](#).
- On Azure, before you run the FreeIPA upgrade, make sure that the resource group has neither a DELETE nor a READ-ONLY lock applied.
- Before performing the FreeIPA upgrade for Data Hub clusters where autoscaling is enabled, you must disable autoscaling and start all compute nodes to ensure the healthy state of Data Hub clusters. After upgrading FreeIPA, autoscaling can be enabled again. For more information, see the [Autoscaling must be stopped before performing FreeIPA upgrade](#) description in the Known issue.

Required roles: EnvironmentAdmin or Owner of the environment

Steps

For CDP UI

1. Log in to the CDP web interface.
2. From the navigation pane, select Environments.
3. Click on your environment.
4. Click on the FreeIPA Upgrade tab.
5. From the Target image dropdown, select a target image. If you would like to upgrade from CentOS to RHEL, ensure that you select a RHEL image. Once you have reviewed the information related to your current base

image and the target base image that will be used for provisioning new instances during the upgrade process. Once you've reviewed the information, click on Upgrade:

FreeIPA Details

STATUS	STATUS REASON	RECIPES	NODES
Running	FreeIPA is Available	0	1

FreeIPA CRN
cmr:cdp:freeipa:us-west-1:pepsi:freeipa:24526722-2dc9-45b5-a1c5-ad04069efba0

UPGRADE STATUS
Upgrade Available

Nodes Recipes **Upgrade**

Upgrade FreeIPA

Target Image
FreeIPA (Latest, OS: redhat8)

A newer FreeIPA image is available. Review the details listed below and click "Upgrade" to proceed.

Image	Name	Date	OS Type	Image Catalog URL
Current	ami-0deabc388a546458a	2023-06-08	centos7	https://cloudbreak-imagecatalog.s3.amazonaws.com/v... Show more
Target	ami-0ebbadb085e7f983	2024-02-06	redhat8	https://cloudbreak-imagecatalog.s3.amazonaws.com/v... Show more

⚠ Please make sure that any 3rd party software that you have installed on your nodes are compatible with the newly selected redhat8 OS type.
Please make sure that any recipes that you have attached to your FreeIPA can be run on the redhat8 OS. If any recipe needs a change, please remove this then attach the updated version as a new one.

Upgrade

- The upgrade process starts. During the upgrade, FreeIPA status switches to **Update in progress**, but the environment remains available. A new instance is provisioned, then the old instance is removed. If more than one instance is present, the process is repeated for the additional instances.
- Once the upgrade is completed, the FreeIPA status changes to **Running**.

For CDP CLI

Use the following command to trigger FreeIPA upgrade (including a CentOS to RHEL upgrade):

```
cdp environments upgrade-freeipa --environment-name <ENVIRONMENT_NAME_OR_CRN>
```

For example:

```
cdp environments upgrade-freeipa --environment-name my-env
```

The command will return a JSON including target and original image information and the operation ID.

```
{
  "targetImage": {
    "catalog": "https://gist.githubusercontent.com/lacikaaa/c096c999ade874a60bf15be897220eed/raw/e30ed12b14c77175b82ceb9caf26500ab1340f26/freeipa-test-catalog.json",
    "id": "9c1c8959-86a7-4b7d-af5a-be252f8b395d",
    "os": "centos7",
    "imageName": "ami-0dfafedeed3a4474f",
    "date": "2021-06-10"
  },
  "originalImage": {
    "catalog": "https://gist.githubusercontent.com/lacikaaa/c096c999ade874a60bf15be897220eed/raw/e30ed12b14c77175b82ceb9caf26500ab1340f26/freeipa-test-catalog.json",
    "id": "0b73e149-6e22-4667-acc5-1d0ba82f8245",
    "os": "centos7",
    "imageName": "ami-030e1d907cfa5ca33",
    "date": "2021-06-04"
  },
  "operationId": "28432209-a4ff-4aea-a8ec-4604d67cdb7c"
}
```

Use the following command to track the progress of the upgrade process:

```
cdp environments get-repair-freeipa-status --operation-id <OPERATION-ID-FROM-UPGRADE-FREEIPA-COMMAND-OUTPUT>
```

The operation ID can be obtained from the output of the upgrade-freeipa command.

What to do next

After upgrading FreeIPA, autoscaling can be enabled again. For more information, see the [Autoscaling must be stopped before performing FreeIPA upgrade](#) description in the Known issue.

In rare cases, the FreeIPA upgrade process might fail. In such cases, you should trigger a retry of the FreeIPA upgrade.

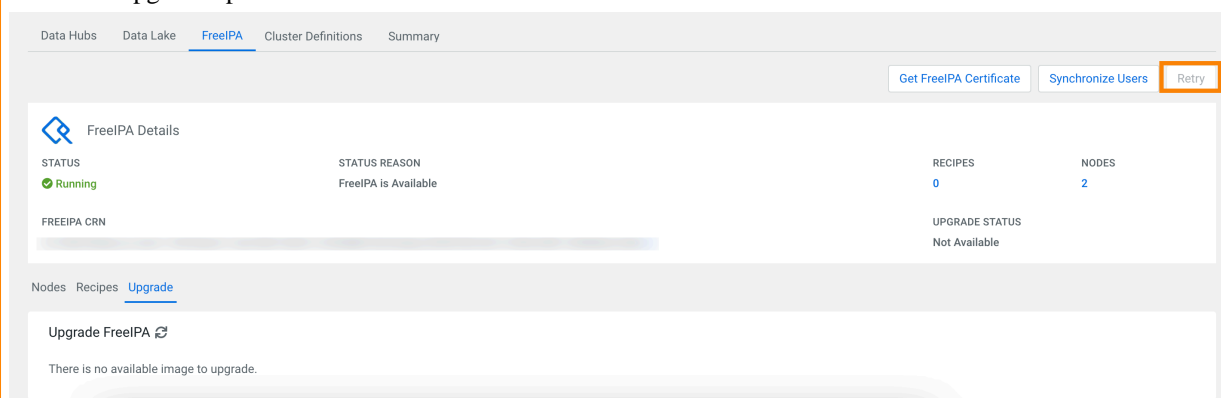
Retry a FreeIPA upgrade

In rare cases, the FreeIPA upgrade process might fail. In such cases, you should trigger a retry of the FreeIPA upgrade.

Steps

For CDP UI

If you are performing the upgrade via CDP web interface, you can find the retry option on the same **FreeIPA** tab where the upgrade option is located:



The Retry option is grayed out if there is no reason to trigger the repair and is only enabled when FreeIPA upgrade encounters problems.

For CDP CLI

If you would like to trigger the repair via CDP CLI, use the following command:

```
cdp environments retry-freeipa -environment <MY-ENVIRONMENT-NAME-OR-CRN>
```

The command returns an operation ID that you can pass in the following command to track the status of the repair:

```
cdp environments get-repair-freeipa-status --operation-id <OPERATION-ID-FROM-UPGRADE-FREEIPA-COMMAND-OUTPUT>
```

Troubleshooting

Issue	Workaround
After performing a FreeIPA upgrade, CDW VMs are broken due to the new FreeIPA server instances' IP addresses not being reflected in CDW.	Manually restart every cdp-coredns-updater pod running in your CDW Kubernetes cluster namespace.
After performing a FreeIPA upgrade, CML workspaces are broken because the IP addresses of the new FreeIPA server instances are not reflected in CML.	Restart the cdp-coredns-updater pod by running <code>kubectl delete -n kube-system \$(kubectl get pods -n kube-system -l app.kubernetes.io/name=cdp-coredns-updater --no-headers -o name)</code> against your CML Kubernetes cluster as soon as the FreeIPA upgrade has completed.

Related Information

[Repair FreeIPA](#)

Repair FreeIPA

When running in high-availability mode, the identity management system runs multiple instances of FreeIPA on separate hosts. In case of failure, you can repair failed hosts using the CDP command-line within one week of a node failing.

The repair process for FreeIPA hosts performs the following steps, stopping when a healthy status returns:

- Start any FreeIPA hosts that are stopped.
- Reboot FreeIPA hosts. (same physical host, same public DNS name, private IP address, and associated storage).
- Restore identity management data from backup.
- Stop and restart FreeIPA hosts (rebuild using new hardware).

This procedure uses the CDP command-line interface. If you haven't already installed the CLI, see “Installing CDP client” for instructions.

On Azure, before you run the repair command, make sure that the resource group has neither a DELETE nor a READ-ONLY lock applied.

Steps

Run the FreeIPA repair command. Run this command from a computer that has network access to the FreeIPA hosts.

```
cdp environments repair-freeipa --environment-name <value>
    [--force | --no-force]
    [--instances <value>]
    [--cli-input-json <value>]
    [--generate-cli-skeleton]
    [--repair-type <string>]
```

where the options are the following:

Option	Description
<code>--environment-name <value></code>	Specifies the FreeIPA's environment name or CRN. The environment CRN is listed in Environment Summary General .
<code>--force --no-force</code>	Choose to force the repair even if the status if the FreeIPA nodes are good. If not specified, defaults to no-force.
<code>--instances <value></code>	Specifies the instance IDs to repair. Use a space to separate multiple instance IDs. If no IDs are provided then all instances are considered for repair. The FreeIPA instance IDs are listed in Environment Summary FreeIPA . You can get the IDs from the output of the <code>cdp environments get-freeipa-status</code> command.

Option	Description
<code>--cli-input-json <value></code>	Performs the operations indicated in the command provided in JSON format. Call <code>generate-cli-skeleton</code> to see a template of the required JSON content. If other arguments are provided on the command line, the command line values override the JSON-provided values.
<code>--generated-cli-skeleton</code>	Displays an example of the format of the input JSON. The repair command does not run.
<code>--repair-type <string></code>	The type of FreeIPA repair to perform. Possible values: <ul style="list-style-type: none"> AUTO - Currently, this is the same as reboot but this may change in the future. REBOOT - Repair the failed instances by rebooting them. REBUILD - Repair the failed instances by deleting them and creating new instances, then replicate data from an existing instance to the new instances. Needs <code>powerUser</code> role.

For example, the following command repairs two instances without forcing the repair:

```
$ cdp environment repair-freeipa --environment-name crn:cdp:environments:us-west-1:12a0079b-1591-dd33-b721-a446bda74e67:environment:36853fcc-2fef-4094-834c-557b4aea34ee --instances i-078ba50f9feb6638f i-09e8b54a343b33d2
```

```
cdp environments repair-freeipa --environment-name john-doe-env2-25793 --instances i-0288d991ed998ec03 --force
{
  "operationId": "edda2c68-5a29-4f60-a150-aca963b36ead",
  "status": "RUNNING",
  "successfulOperationDetails": [],
  "failureOperationDetails": [],
  "startDate": "2020-10-01T19:48:36.009000+00:00"
}
```

If you see the following error, consider rerunning the repair command with the `--force` option:

```
An error occurred: {"message":"No unhealthy instances to reboot. Maybe use the force option."} (Status Code: 404; Error Code: NOT_FOUND; Service: environments; Operation: repairFreeipa; Request ID: eca3a7fb-aa6b-48f7-ae29-b881161869e5;)
```

Related Information

[Installing CDP client](#)

Check FreeIPA repair status

You can check the status of an in-progress repair operation with `get-repair-freeipa-status`.

This procedure uses the CDP command-line interface. If you haven't already installed the CLI, see “Installing CDP client” for instructions.

Steps

Run the FreeIPA status command. Run this command from a computer that has network access to the FreeIPA hosts.

```
cdp environments get-repair-freeipa-status --operation-id <value>
[ --cli-input-json <value> ]
[ --generate-cli-skeleton ]
```

where the options are the following:

Option	Description
<code>--operation-id <value></code>	Operation-id for the previously requested repair operation.

Option	Description
--cli-input-json <value>	Performs service operation based on the JSON string provided. The JSON string follows the format provided by --generate-cli-skeleton. If other arguments are provided on the command line, the CLI values will override the JSON-provided values.
--generate-cli-skeleton	Prints a sample input JSON to standard output. Note the specified operation is not run if this argument is specified. The sample input can be used as an argument for --cli-input-json.

This command can take 15 to 45 seconds to run as it gathers information in real-time. The output of the status command provides the status for the repair operation (in JSON format).

```
cdp environments get-repair-freeipa-status --operation-id edda2c68-5a29-4f60
-a150-aca963b36ead
{
  "status": "COMPLETED",
  "successfulOperationDetails": [
    {
      "environmentCrn": "crn:cdp:environments:us-west-1:9d74eee4-1ca
d-45d7-b645-7ccf9edbb73d:environment:08c55413-6e2b-4664-8367-ef3fc0787773"
    }
  ],
  "failureOperationDetails": [],
  "startDate": "2020-10-01T19:48:36.009000+00:00",
  "endDate": "2020-10-01T19:49:08.392000+00:00"
}
```

Element	Data Type	Description
status	string	Status of a repair operation. Possible values: REQUESTED, RUNNING, COMPLETED, FAILED, REJECTED, TIMEDOUT.
successfulOperationDetails	array	List of operation details for all successes. If the repair is only partially successful both successful and failure operation details will be populated.
failureOperationDetails	array	List of operation details for failures. If the repair is only partially successful both successful and failure operation details will be populated.
error	string	If there is any error associated. The error will be populated on any error and it may be populated when the operation failure details are empty. The error will typically contain the high level information such as the associated repair failure phase.
startDate	datetime	Date when the operation started.
endDate	datetime	Date when the operation ended. Omitted if operation has not ended.

Resize FreeIPA

After resizing your Data Lake from Light Duty to Medium Duty you should also resize your FreeIPA cluster. You can do this via CDP CLI.

When you register a CDP environment, the FreeIPA node count (also known as “availability type”) depends on the Data Lake cluster shape:

- When an environment with a Medium Duty Data Lake is provisioned, an HA FreeIPA cluster consisting of three instances is created along with it.
- When an environment with a Light Duty Data Lake is provisioned, a two-node FreeIPA cluster is created along with it.

When you resize (or “migrate”) your Data Lake from Light Duty to Medium Duty, you should resize your FreeIPA cluster accordingly from two to three nodes so that a Medium Duty Data Lake can be backed by a FreeIPA server that offers higher availability.

In general, FreeIPA resizing scenarios are supported:

- Upscaling from one to three nodes
- Upscaling from two to three nodes
- Downscaling from three to two nodes

The following validations are performed before the scaling operation:

- Scale is possible given the current node count.
- The instances are not deleted and all the instances are available.
- FreeIPA stack is available.
- The scaling path is supported by Cloudera Management Console

Scale up FreeIPA

To upscale your FreeIPA cluster, use the following command:

```
cdp environments upscale-freeipa \  
  --environment-name <ENVIRONMENT-NAME> \  
  --target-availability-type HA
```

For example:

```
cdp environments upscale-freeipa \  
  --environment-name my-env \  
  --target-availability-type HA
```

This operation requires the `--target-availability-type` parameter, with the following possible values:

- HA - 3 node HA cluster
- TWO_NODE_BASED - 2 node cluster
- NON_HA - 1 node-cluster

Scale down FreeIPA

To downscale your FreeIPA cluster, use the following command:

```
cdp environments downscale-freeipa \  
  --environment-name <ENVIRONMENT-NAME> \  
  --target-availability-type <AVAILABILITY-TYPE>
```

For example:

```
cdp environments downscale-freeipa \  
  --environment-name my-env \  
  --target-availability-type HA
```

This operation requires the `--target-availability-type` parameter, with the following possible values:

- HA - 3 node HA cluster
- TWO_NODE_BASED - 2 node cluster
- NON_HA - 1 node-cluster

Configuring workload password policies

In order to bring your workload password complexity requirements in line with company policy, you can set your FreeIPA password policies via CDP web interface and CDP CLI. Password policies can be configured for length, complexity, expiration, and scope.



Note: Configuring password policies takes effect in all environments within a tenant, but applies to newly set passwords only. As such, admins should advise users to reset their passwords to achieve compliance with their new password policy.



Note: Prior to introducing this feature, CDP web UI had stricter password complexity requirements than the CLI. With the release of the password policy feature, both the UI and CLI enforce the same stronger password policy by default.



Warning: There is currently no stable notification system in place that would inform users that their password expired. When users SSH to a node and their password has expired, they may be prompted to reset their password in the SSH session. As resetting the password in the SSH session may only work for a short period, the users should instead set a new workload password using the Management Console in CDP. For instructions, see [Setting the workload password](#).

Workload password policy types

There are two types of password policies:

- Global policies - Apply to all users including machine users
- Machine user policies - Apply to machine users only

You can set either or both policies. By default, global policies are applied to all users, including machine users. An optional override for configuring a different policy for machine users is available. For example, setting strict password expiration policies for machine users may not be desired, as password expiration in those accounts may cause upstream failures in the applications that use them.

Default workload password policy

If a password policy has not been set, the following default password policy is used:

- A minimum password length of 8 characters
- Must include at least 1 upper case character, lowercase character, number and special character. Supported special characters are: "#", "&", "*", "\$", "%", "@", "^", ".", "_", and "!".
- All previous passwords can be reused
- The password can be changed at any time
- The password never expires

For detailed information on how to manage workload password policies, refer to the following documentation:

Check the current workload password policy

You can check your current password policy from the Workload Password Policies page or using the `cdp iam get-account` CLI command.

Required role: PowerUser

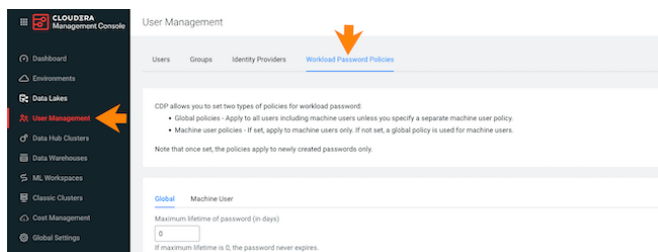
Steps

For CDP UI

You can check the current password policy from the Workload Password Policies page. To access this page:

1. Log in to the CDP web interface.

2. Navigate to the Management Console > User Management > Workload Password Policies:



For CDP CLI

Use the `cdp iam get-account` to obtain your current password policy.

Set a password policy

You can set password policies from the Workload Password Policies page or using the via CDP CLI using the `cdp iam set-workload-password-policy` command.



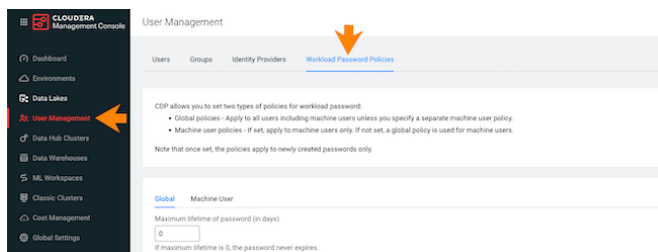
Note: Prior to introducing this feature, CDP web UI had stricter password complexity requirements than the CLI. With the release of the password policy feature, both the UI and CLI enforce the same stronger password policy by default.

Required role: PowerUser

Steps

For CDP UI

1. Log in to the CDP web interface.
2. Navigate to the Management Console > User Management > Workload Password Policies:



3. In the Global tab, specify a policy that applies to all CDP users and machine users. The following options are available:

UI option	Description
Minimum lifetime of password (in days)	Once set, the password must remain the same for this period of time. Note: If a user forgets their password, they will be unable to reset it until the minimum period has passed or until the PowerUser unsets the minimum lifetime of the password for them.
Maximum lifetime of password (in days)	Allows you to specify password expiration period in days.
Minimum password length	Allows you to specify minimum password length, must be between 6 and 256 characters.

Number of previous passwords that can't be reused	If set to 0, all previous passwords can be reused. Any number larger than 0 indicates the number of most recent passwords that can't be reused. The maximum allowed value for this parameter is 20, so you can prevent users from reusing up to 20 recent passwords. Note: Password history information is only recorded when password history size is set to a value other than zero. This means that when the password history size is initially set from zero to non-zero, the previous passwords that were set (while the password history size was as at 0) are not considered when the password history check is done.
Must include uppercase characters	When checked, at least one uppercase character is required
Must include lowercase characters	When checked, at least one lowercase character is required
Must include numbers	When checked, at least one number is required
Must include symbols	When checked, at least one special character is required. Supported special characters are: "#", "&", "*", "\$", "%", "@", "^", ".", "_", and "!".

4. Click Update.

5. By default, global policies are applied to machine users. If you would like to set a different policy for machine users:

- a. Navigate to the Machine User tab.
- b. Uncheck Inherit from global policy.
- c. Set a desired policy (the available options are the same as for global policy).
- d. Click Update.

For CDP CLI

The following example creates a global policy:

```
cdp iam set-workload-password-policy --global-password-policy minPasswordLength=8,mustIncludeUpperCaseCharacters=true,mustIncludeLowerCaseCharacters=true,mustIncludeNumbers=true,mustIncludeSymbols=false
```

The following example creates a machine user policy. This overrides the global policy for machine users:

```
cdp iam set-workload-password-policy --machine-users-password-policy minPasswordLength=8,mustIncludeUpperCaseCharacters=true,mustIncludeLowerCaseCharacters=true,mustIncludeNumbers=true,mustIncludeSymbols=true
```

The following password complexity requirements can be set as part of your policy:

CLI option	Type	Description	Default value
minPasswordLifetimeDays	integer	Minimum period in days during which password cannot be changed once it is set Note: If a user forgets their password, they will be unable to reset it until the minimum period has passed or until the PowerUser unsets the minimum lifetime of the password for them.	0
maxPasswordLifetimeDays	integer	Expiration period in days	0
minPasswordLength	integer	Minimum password length; must be between 6 and 256 characters.	8

passwordHistorySize	integer	Number of previous passwords that can't be reused. 0 indicates that all previous passwords can be reused. Any number above 0 indicates the number of most recent passwords that can't be reused. The maximum allowed value for this parameter is 20, so you can prevent users from reusing up to 20 recent passwords. Note: Password history information is only recorded when password history size is set to a value other than zero. This means that when the password history size is initially set from zero to non-zero, the previous passwords that were set (while the password history size was as at 0) are not considered when the password history check is done.	0
mustIncludeUpperCaseCharacters	true/false	At least one uppercase character is required	true
mustIncludeLowerCaseCharacters	true/false	At least one lowercase character is required	true
mustIncludeNumbers	true/false	At least one number is required	true
mustIncludeSymbols	true/false	At least one special character is required. Supported special characters are: "#", "&", "*", "\$", "%", "@", "^", ".", "_", and "!".	true

Reset a password policy

You can reset password policies via CDP CLI using the `cdp iam unset-workload-password-policy` command. As a result, default password policies will be reinstated.

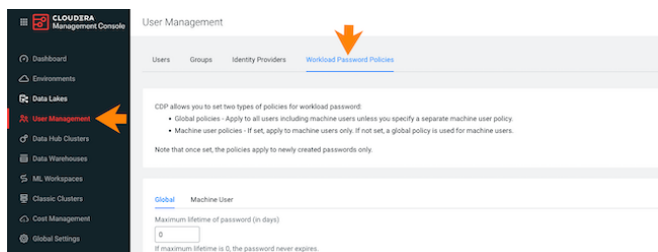
A global password policy is always present for an account. When the global password policy is unset, the policy will revert to the documented defaults. A machine user password policy may or may not be present in the account. When the machine user policy is not set for the account, the global password policy will be enforced for machine users.

Required role: PowerUser

Steps

For CDP UI

1. Log in to the CDP web interface.
2. Navigate to the Management Console > User Management > Workload Password Policies:



3. Navigate to the Machine User tab and make sure that Inherit from global policy is checked.



Note: If this option remains unchecked, machine user policies will not be reset.

4. Click on Reset to default values.
5. Click OK to confirm.

For CDP CLI

Use the following commands to reset password policies:

```
cdp iam unset-workload-password-policy --unset-global-password-policy
```

```
cdp iam unset-workload-password-policy --unset-machine-users-password-policy
```

Unset user's minimum password lifetime

With the minimum workload password lifetime enabled, it may happen in very rare cases that a user becomes locked out of their account and their minimum password lifetime needs to be temporarily unset.

If your organization's workload password policy has the minimum password lifetime enabled, a user is unable to set a new workload password until the minimum password lifetime duration has passed. Consequently, a user who forgets their password will be unable to set a new password until the minimum password lifetime duration has passed. In such a case, a PowerUser can unset the minimum lifetime date for the user who has been locked out of their account. This is a one-time override that will allow the user to set a new workload password.

Required role: PowerUser

Steps

Use the following command to unset the minimum password lifetime for a specific user:

```
cdp iam unset-workload-password-min-lifetime --user <USER-CRN-OR-ID>
```

This is a one-time override. Once the user sets their password, the minimum password lifetime will be reset to the original value that you set for your organization.

Accessing a FreeIPA cluster via SSH

If you plan to access a FreeIPA cluster (for example for troubleshooting purposes) via a command line client, SSH into the master node.

SSH to a FreeIPA node as cloudbreak user

Required role: No CDP role is required

CDP administrators can access FreeIPA cluster nodes as cloudbreak user with the SSH key provided during cluster creation.

On Mac OS, you can use the following syntax to SSH to the VM::

```
ssh -i "privatekey.pem" cloudbreak@publicIP
```

For example:

```
ssh -i "testkey-kp.pem" cloudbreak@90.101.0.132
```

On Windows, you can access your cluster via SSH by using an SSH client such as PuTTY. For more information, refer to [How to use PuTTY on Windows](#).

SSH to a FreeIPA node as your own CDP user

Required role: EnvironmentUser, Data Steward, or EnvironmentAdmin

A user who has the required CDP role assigned to them can SSH to FreeIPA cluster nodes as themselves.

On Mac OS, you can use the following syntax to SSH to the VM::

```
ssh -i "privatekey.pem" cdpusername@publicIP
```

For example:

```
ssh -i "testkey-kp.pem" jsmith@90.101.0.132
```

On Windows, you can access your cluster via SSH by using an SSH client such as PuTTY. For more information, refer to [How to use PuTTY on Windows](#).

Recipes

A recipe is a script that runs on all nodes of a selected host group at a specific time. You can use recipes to create and run scripts that perform specific tasks on your Data Hub, Data Lake, or FreeIPA cluster nodes.

You can use recipes for tasks such as installing additional software or performing advanced cluster configuration. For example, you can use a recipe to put a JAR file on the Hadoop classpath.

Recipes can be uploaded and managed via the CDP web interface or CLI and then selected, when needed, for a specific cluster and for a specific host group. If selected, they will be executed on a specific host group at a specified time.

Depending on the need, a recipe can be executed at various times. Available recipe execution times are:

- Before Cloudera Manager server start
- After Cloudera Manager server start
- After cluster installation
- Before cluster termination

Recipes are stored on the Cloudera Manager server for the lifetime of the master node, and are executed at specific times of your choosing:

- pre-service-deployment (formerly pre-cluster-manager-start): During a Data Hub, Data Lake, or environment deployment, the script will be executed on every node before the CM server starts, and after node repair or OS upgrade of a cluster.
- post-cluster-manager-start: During a Data Hub or Data Lake deployment, the script will be executed on every node after the CM server starts, but before cluster installation. post-cluster-manager start recipes are also executed after node repair or OS upgrade of a cluster. This option is not available for FreeIPA recipes.
- post-service-deployment (formerly post-cluster-install): The script will be executed on every node after cluster installation on the CM server is finished, and after node repair or OS upgrade of a cluster.
- pre-termination: The script will be executed on every node before cluster termination.



Note: On the master node, recipes are triggered when the CM server starts; on other nodes, recipes are triggered when the CM agent starts.

Writing recipes

Refer to these guidelines when creating your recipes.

When using recipes, consider the following guidelines:

- Running bash and python scripts as recipes is supported. We recommend using scripts with [Shebang](#) character sequence, for example:

```
#!/bin/sh
#!/bin/bash
#!/usr/bin/sh
#!/usr/bin/bash
#!/usr/bin/env sh
#!/usr/bin/env bash
#!/bin/sh -x
#!/usr/bin/python
#!/usr/bin/env python
```

- The scripts are executed as root. The recipe output is written to `/var/log/recipes` on each node on which it was executed.
- Supported parameters can be specified as variables by using mustache kind of templating with "`{{{ }}`" syntax. Once specified in a recipe, these variables are dynamically replaced when the recipe is executed, generating the actual values that you provided as part of cluster creation process. For the list of parameters, refer to [Recipe and cluster template parameters](#). For an example, see [Example: Recipe using parameters](#).



Note: Using variable parameters is not supported for FreeIPA recipes.

For example, if your cluster includes an external LDAP and your recipe includes `{{{ldap.connectionURL}}}`, as demonstrated in the following example

```
#!/bin/bash -e

main() {
  ping {{{ ldap.connectionURL }}}
}
[[ "$0" == "$BASH_SOURCE" ]] && main "$@"
```

then, when this recipe runs, the `{{{ldap.connectionURL}}}` is replaced with the actual connection URL specified as part of cluster creation process, as demonstrated in the following example:

```
#!/bin/bash -e

main() {
  ping 192.168.59.103
}
[[ "$0" == "$BASH_SOURCE" ]] && main "$@"
```

- Recipe logs can be found at `/var/log/recipes/${RECIPE_TYPE}/${RECIPE_NAME}.log`
- The scripts are executed on all nodes of the host groups that you select (such as “master”, “worker”, “compute”).
- In order to be executed, your script must be in a network location which is accessible from the Management Console and the virtual network in which your cluster is located.
- Make sure to follow Linux best practices when creating your scripts. For example, don’t forget to script “Yes” auto-answers where needed.
- Do not execute `yum update -y` as it may update other components on the instances (such as salt) – which can create unintended or unstable behavior.

Example Python script

```
#!/usr/bin/python
print("An example of a python script")
import sys
print(sys.version_info)
```

Example bash script for yum proxy settings

```
#!/bin/bash
cat >> /etc/yum.conf
<<ENDOF
proxy=http://10.0.0.133:3128
ENDOF
```

Example recipe including variables

Original recipe:

```
#!/bin/bash -e

function setupAtlasServer() {
    curl -iv -u {{{ general.userName }}}:{{{ general.password }}} -H "X-Requeste
d-By: ambari" -X POST -d '{"RequestInfo":{"command":"RESTART","context
":"Restart all components required ATLAS","operation_level":{"level":"SERVIC
E","cluster_name":"{{{ general.clusterName }}}"},"service_name":"ATLAS"}}',"Re
quests/resource_filters":[{"hosts_predicate":"HostRoles/stale_configs=false&
HostRoles/cluster_name={{{ general.clusterName }}}"}]}' http://$(hostname -f
):8080/api/v1/clusters/{{{ general.clusterName }}}/requests
}

main() {
    setupAtlasServer
}

[[ "$0" == "$BASH_SOURCE" ]] && main "$@"
```

Generated recipe (to illustrate how the variables from the original recipe were replaced during cluster creation):

```
#!/bin/bash -e

function setupAtlasServer() {
    curl -iv -u admin:admin123 -H "X-Requested-By: ambari" -X POST -d '{"R
equestInfo":{"command":"RESTART","context":"Restart all components required
ATLAS","operation_level":{"level":"SERVICE","cluster_name":"super-cluster","
service_name":"ATLAS"}}',"Requests/resource_filters":[{"hosts_predicate":"Hos
tRoles/stale_configs=false&HostRoles/cluster_name=super-cluster"}]}' http://
$(hostname -f):8080/api/v1/clusters/super-cluster/requests
}

main() {
    setupAtlasServer
}

[[ "$0" == "$BASH_SOURCE" ]] && main "$@"
```

Register a recipe

In order to use your recipe for clusters, you must first register it with the Management Console.

About this task

Required role: EnvironmentCreator can create a shared resource and then assign users to it.

SharedResourceUser or Owner of the shared resource can use the resource.

Before you begin

If you are using CDP with a proxy, note that the CDP proxy settings do not apply to cluster recipes. If you planning to use the recipes, then you can set the proxy settings manually. You can find the proxy settings in the `/etc/cdp/proxy.env` file.

Procedure

1. Place your script in a network location accessible from Management Console and from the virtual network in which your clusters are located.
2. Log in to the CDP web interface.
3. Navigate to Shared ResourcesRecipes and click Register Recipe.
4. Provide the following:

Parameter	Value
Name	Enter a name for your recipe.
Description	(Optional) Enter a description for your recipe.
Execution Type	Select one of the following options: <ul style="list-style-type: none"> • pre-service-deployment (formerly pre-cluster-manager-start): During a Data Hub, Data Lake, or environment deployment, the script will be executed on every node (in the host group where you assigned the recipe) before the CM server starts. • post-cluster-manager-start: During a Data Hub or Data Lake deployment, the script will be executed on every node (in the host group where you assigned the recipe) after the CM server starts, but before cluster installation. This option is not available for FreeIPA recipes. • post-service-deployment (formerly post-cluster-install):: The script will be executed on every node (in the host group where you assigned the recipe) after cluster installation on the CM server is finished. • pre-termination: The script will be executed on every node (in the host group where you assigned the recipe) before cluster termination.
Script	Select one of: <ul style="list-style-type: none"> • File: Point to a file on your machine that contains the recipe. • Text: Paste the script.

5. Click Register.

What to do next

- When you create a Data Hub cluster, you can select a previously added recipe on the advanced Cluster Extensions page of the create cluster wizard.
- When you create an environment, you can select a previously added recipe on the Data Access and Data Lake Scaling page of the environment creation wizard, under Advanced Options > Cluster Extensions > Recipes.
- When you create an environment, you can select a previously added FreeIPA recipe on the **Region, Networking, and Security** page of the environment creation wizard, under Advanced OptionsCluster ExtensionsRecipes.
- You can also attach recipes to Data Hub or Data Lake clusters when you create an environment/Data Lake or Data Hub through the CDP CLI.

Update a FreeIPA recipe

You can attach or detach recipes to/from existing FreeIPA clusters in an available state. Using this capability, you can update a recipe by removing it from the cluster, replacing the old recipe with a modified recipe of the same type, and attaching the new modified recipe to the cluster.

Attaching or detaching a recipe will not execute the recipe. The next execution of the recipe will take place based on the type of the recipe. After an upscale, a newly attached recipe runs only on the new hosts.

Required role (one of the following):

- PowerUser on CDP tenant
- Owner of the environment
- EnvironmentAdmin

Steps

For CDP UI

1. Create a new recipe (with updated/modified content) of the same type as the old recipe that you want to replace.
2. In the Management Console, select Environments from the left pane and click on your environment. Next, click on FreeIPA tab and then on the Recipes tab.
3. Find the recipe that you want to remove in the list of recipes for the FreeIPA.
4. Click Remove Recipe next to the name of the recipe that you want to remove, then click Yes in the confirmation window.
5. Once you have removed the old recipe, click on the Add Recipe button for the FreeIPA. Then select the name of the new recipe that contains the modified content and click Add.

For CDP CLI

1. Create a new recipe (with updated/modified content) of the same type as the old recipe that you want to replace.
2. You can use the CDP CLI to detach and attach recipes from a FreeIPA cluster:

```
cdp environments detach-free-ipa-recipes
--environment <value>
--recipes <value>

cdp environments attach-free-ipa-recipes
--environment <value>
--recipes <value>
```

Result: You should see the new recipe appear for the same host group. After this change, the next recipe execution will execute the new script.

Managing recipes from CLI

You can manage recipes from CLI using `cdp datahub` commands.

Required role: EnvironmentCreator can create a shared resource and then assign users to it.

SharedResourceUser or Owner of the shared resource can use the resource. The Owner of the shared resource can delete it.



Note: Currently, recipes use `cdp datahub` commands regardless of whether the recipe is intended to run on Data Hub, Data Lake, or FreeIPA cluster nodes.

- Register a new recipe: `cdp datahub create-recipe --recipe-name <value> --recipe-content <value> --type <value>`

Supported types:

- `PRE_SERVICE_DEPLOYMENT` (formerly `PRE_CLOUDERA_MANAGER_START`)
- `POST_CLOUDERA_MANAGER_START` (this option is not available for FreeIPA recipes)
- `POST_SERVICE_DEPLOYMENT` (formerly `POST_CLUSTER_INSTALL`)
- `PRE_TERMINATION`
- List all available recipes: `cdp datahub list-recipes`
- Describe a specific recipe: `cdp datahub describe-recipe --recipe-name <value>`
- Delete one or more existing recipes: `cdp datahub delete-recipes --recipe-name <value>`