HDP Security Secure Credentials Management 3

Securing Credentials

Date of Publish: 2018-07-15

http://docs.hortonworks.com

Contents

Secure Credentials Management Overview

The Secure Credentials Management section covers encrypting database and LDAP passwords, changing the master key, and configuring Ambari for non-root.

Related Information

Configuring Ambari for Non-Root

Encrypt Database and LDAP Passwords in Ambari

Remove Encryption Entirely

Change the Current Master Key

Encrypt Database and LDAP Passwords in Ambari

By default the passwords to access the Ambari database and the LDAP server are stored in a plain text configuration file. To have those passwords encrypted, you need to run a special setup command.

About this task

Ambari Server should not be running when you do this: either make the edits before you start Ambari Server the first time or bring the server down to make the edits.

Procedure

1. On the Ambari Server, run the special setup command and answer the prompts: ambari-server setup-security.
2. Select Option 2:

```
Choose one of the following options:
[1] Enable HTTPS for Ambari server.
[2] Encrypt passwords stored in ambari.properties file.
[3] Setup Ambari kerberos JAAS configuration.
```

3. Provide a master key for encrypting the passwords. You are prompted to enter the key twice for accuracy.

   If your passwords are encrypted, you need access to the master key to start Ambari Server.
4. You have three options for maintaining the master key:
   - Persist it to a file on the server by pressing y at the prompt.
   - Create an environment variable AMBARI_SECURITY_MASTER_KEY and set it to the key.
   - Provide the key manually at the prompt on server start up.
5. Start or restart the Server: ambari-server restart.

Remove Encryption Entirely

How to reset Ambari database and LDAP passwords to a completely unencrypted state.

Procedure

1. On the Ambari host, open /etc/ambari-server/conf/ambari.properties with a text editor.
2. Set this property: security.passwords.encryption.enabled=false.
3. Delete /var/lib/ambari-server/keys/credentials.jceks.
4. Delete /var/lib/ambari-server/keys/master.

5. You must now reset the database password and, if necessary, the LDAP password. Run ambari-server setup and ambari-server setup-ldap again.

Related Information

Encrypt Database and LDAP Passwords in Ambari

Preparing for LDAPS integration

Change the Current Master Key

How to change the master key.

Procedure

- If you know the current master key or if the current master key has been persisted:
  a) Re-run the encryption setup command and follow the prompts: ambari-server setup-security.
  b) Select Option 2:

```
Choose one of the following options:
[1] Enable HTTPS for Ambari server.
[2] Encrypt passwords stored in ambari.properties file.
[3] Setup Ambari kerberos JAAS configuration.
```

  c) Enter the current master key when prompted if necessary (if it is not persisted or set as an environment variable).
  d) At the Do you want to reset Master Key prompt, enter yes.
  e) At the prompt, enter the new master key and confirm.
- If you do not know the current master key:
  - Remove encryption entirely, as described in “Remove Encryption Entirely” (link below).
  - Re-run ambari-server setup-security as described above.
  - Start or restart the Ambari Server: ambari-server restart.

Related Information

Remove Encryption Entirely

Configuring Ambari for Non-Root

In most secure environments, restricting access to and limiting services that run as root is a hard requirement. For these environments, Ambari can be configured to operate without direct root access. Both Ambari Server and Ambari Agent components allow for non-root operation, and the following sections will walk you through the process.

Configure Ambari Server for Non-Root

You can configure the Ambari Server to run as a non-root user.

Procedure

1. During the ambari-server setup process, when prompted to Customize user account for ambari-server daemon?, choose y.
2. When prompted, enter the appropriate, non-root user to run the Ambari Server as; for example: ambari.

3. The non-root functionality relies on sudo to run specific commands that require elevated privileges as defined in the Sudoer configuration. Each of the substeps include the specific sudo entries that you should place in /etc/sudoers by running the visudo command:
   a) Enter the the specific commands that must be issued for standard server operations:

```
# Ambari Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /bin/mkdir -p /etc/security/keytabs, /
bin/ls /etc/security/keytabs, /bin/chmod * /etc/security/keytabs/
*.keytab, /bin/chown * /etc/security/keytabs/*.keytab, /bin/chgrp
 * /etc/security/keytabs/*.keytab, /bin/rm -f /etc/security/keytabs/
*.keytab, /bin/cp -p -f /var/lib/ambari-server/data/tmp/* /etc/security/
keytabs/*.keytab
ambari ALL=(ALL) NOPASSWD:SETENV: /bin/mkdir -p /var/lib/ambari-server/
data/tmp, /bin/chmod * /var/lib/ambari-server/data/tmp, /bin/chown * /
var/lib/ambari-server/data/tmp, /bin/chgrp * /var/lib/ambari-server/
data/tmp, /bin/rm -rf /var/lib/ambari-server/data/tmp/*, /bin/cp -f /
tmp/* /var/lib/ambari-server/data/tmp/*, /usr/bin/test * *, /bin/stat -c
 %u %g %a /var/lib/ambari-server/data/tmp/*
```

   b) Some versions of sudo have a default configuration that prevents sudo from being invoked from a non-interactive shell. In order for the agent to run it's commands non-interactively, some defaults need to be overridden:

```
Defaults exempt_group = ambari
Defaults !env_reset,env_delete-=PATH
Defaults: ambari !requiretty
```

What to do next

To ensure that the configuration has been done properly, you can su to the ambari user and run sudo -l. There, you can double check that there are no warnings, and that the configuration output matches what was just applied.

Configure an Ambari Agent for Non-Root

You can configure the Ambari Agent to run as a non-privileged user

About this task

The non-privileged user requires specific sudo access in order to su to Hadoop service accounts and perform specific privileged commands. Configuring Ambari Agents to run as non-root requires that you manually install agents on all nodes in the cluster. For these details, see “Installing Ambari Agents Manually” (link below). After installing each agent, you must configure the agent to run as the desired, non-root user. In this example we will use the ambari user.

Procedure

1. Change the run_as_user property in the /etc/ambari-agent/conf/ambari-agent.ini file.
   run_as_user=ambari
2. Restart ambari-agent to begin running as the non-root user: ambari-agent restart.
3. The non-root functionality relies on sudo to run specific commands that require elevated privileges as defined in the Sudoer configuration. Each of the substeps include the specific sudo entries that should be placed in /etc/sudoers by running the visudo command:
   a) Enter the su commands and corresponding Hadoop service accounts that are configurable on install:

```
# Ambari Customizable Users
ambari ALL=(ALL) NOPASSWD:SETENV: /bin/su hdfs *,/bin/su ambari-qa *,/
bin/su ranger *,/bin/su zookeeper *,/bin/su knox *,/bin/su falcon *,/
bin/su ams *, /bin/su flume *,/bin/su hbase *,/bin/su spark *,/bin/su
 accumulo *,/bin/su hive *,/bin/su hcat *,/bin/su kafka *,/bin/su mapred
```

```
 *,/bin/su oozie *,/bin/su sqoop *,/bin/su storm *,/bin/su tez *,/bin/su
 atlas *,/bin/su yarn *,/bin/su kms *,/bin/su activity_analyzer *,/bin/
su livy *,/bin/su zeppelin *,/bin/su infra-solr *,/bin/su logsearch *
```

These user accounts must match the service user accounts referenced in the Customize Services > Misc tab during the Install Wizard configuration step. For example, if you customize YARN to run as xyz_yarn, modify the su command above to be /bin/su xyz_yarn.

b) Enter the specific commands that must be issued for standard agent operations:

```
   # Ambari: Core System Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/bin/yum,/usr/bin/zypper,/usr/
bin/apt-get, /bin/mkdir, /usr/bin/test, /bin/ln, /bin/ls, /bin/chown, /
bin/chmod, /bin/chgrp, /bin/cp, /usr/sbin/setenforce, /usr/bin/test, /
usr/bin/stat, /bin/mv, /bin/sed, /bin/rm, /bin/kill, /bin/readlink, /
usr/bin/pgrep, /bin/cat, /usr/bin/unzip, /bin/tar, /usr/bin/tee, /bin/
touch, /usr/bin/mysql, /sbin/service mysqld *, /usr/bin/dpkg *, /bin/rpm
 *, /usr/sbin/hst *
```

```
   # Ambari: Hadoop and Configuration Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/bin/hdp-select, /usr/bin/conf-
select, /usr/hdp/current/hadoop-client/sbin/hadoop-daemon.sh, /usr/lib/
hadoop/bin/hadoop-daemon.sh, /usr/lib/hadoop/sbin/hadoop-daemon.sh, /
usr/bin/ambari-python-wrap *
```

```
   # Ambari: System User and Group Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/sbin/groupadd, /usr/sbin/
groupmod, /usr/sbin/useradd, /usr/sbin/usermod
```

```
# Ambari: Kerberos Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/bin/klist -k /etc/security/
keytabs/*
```

```
   # Ambari: Knox Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/bin/python2.6 /var/lib/ambari-
agent/data/tmp/validateKnoxStatus.py *, /usr/hdp/current/knox-server/
bin/knoxcli.sh
```

```
   # Ambari: Ranger Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/hdp/*/ranger-usersync/setup.sh, /
usr/bin/ranger-usersync-stop, /usr/bin/ranger-usersync-start, /
usr/hdp/*/ranger-admin/setup.sh *, /usr/hdp/*/ranger-knox-plugin/
disable-knox-plugin.sh *, /usr/hdp/*/ranger-storm-plugin/disable-
storm-plugin.sh *, /usr/hdp/*/ranger-hbase-plugin/disable-hbase-
plugin.sh *, /usr/hdp/*/ranger-hdfs-plugin/disable-hdfs-plugin.sh *, /
usr/hdp/current/ranger-admin/ranger_credential_helper.py, /usr/hdp/
current/ranger-kms/ranger_credential_helper.py, /usr/hdp/*/ranger-*/
ranger_credential_helper.py
```

```
   # Ambari Infra and LogSearch Commands
ambari ALL=(ALL) NOPASSWD:SETENV: /usr/lib/ambari-infra-solr/bin/solr
 *, /usr/lib/ambari-logsearch-logfeeder/run.sh *, /usr/sbin/ambari-
metrics-grafana *, /usr/lib/ambari-infra-solr-client/solrCloudCli.sh *
```

Note:

Do not modify the command lists, only the usernames in step 3a may be modified.

This sudo configuration must be done on every node in the cluster.

c) Some versions of sudo have a default configuration that prevents sudo from being invoked from a non-interactive shell. In order for the agent to run it's commands non-interactively, some defaults need to be overridden:

```
Defaults exempt_group = ambari
Defaults !env_reset,env_delete-=PATH
Defaults: ambari !requiretty
```

This sudo configuration must be done on every node in the cluster.

What to do next

To ensure that the configuration has been done properly, you can su to the ambari user and run sudo -l. There, you can double-check that there are no warnings, and that the configuration output matches what was just applied.