Centrify Server Suite 2016

Centrify Identity and Access Management for DataStax

May 2016

Centrify Server Suite 2015.1

DataStax version 4.8

Legal notice

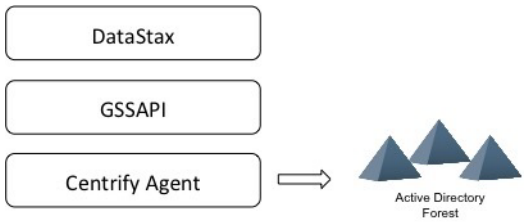
Contents

Benefits of integrating with Centrify

Centrify Server Suite is an enterprise-class solution that supports NoSQL DataStax big data environments. Together, Centrify and DataStax allow you to use your organization's existing Active Directory infrastructure to deliver access control and privilege management.

By installing the Centrify agent on NoSQL nodes you can provide single sign-on identity and access management for the users who will log on to clusters in the big data environment with their Active Directory credentials, by implementing Kerberos using GSSAPI.

The Centrify agent reduces identity-related risks by enforcing access controls and least-privilege security across nodes and clusters, and increases regulatory compliance through control over user access and the ability to trace activity back to an individual user.



Preparing for integration with Centrify

The following sections describe how to install and configure DataStax for integration with the Centrify agent. For detailed instructions on preparing your NoSQL environment, go to http://www.datastax.com/

Basic prerequisites

- Active Directory must be installed and at least one domain controller must be available.
- You should have a Windows workstation joined to the domain where you can run administrative consoles.
- You should have appropriate Centrify Server Suite Enterprise Edition software installed or available to be installed. For example, to install DataStax on a machine running RedHat Enterprise Linux 6, you would download centrify-suite-2015.1-rhel: http://centrify.com/support/customer-support-portal/download-center/#2015.1

  You can request a free trial of Centrify Server Suite by filling out the http://www.centrify.com/free-trial/server-suite-form/ on the Centrify website and specifying `NoSQL` in the Comments field.

- You should have Centrify Server Suite documentation available for reference.

  You can download documentation from http://community.centrify.com/t5/custom/page/page-id/Centrify-Documentation after you register your free trial and set up your Centrify account.

- You should have the latest Enterprise version of DataStax software available, this guide uses version 4.8. You can download the latest version from https://academy.datastax.com/downloads.
- You should install and set the Oracle Java Development Kit as your default JDK. The software can be downloaded from http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html.

Integrating Datastax and Centrify

The following sections describe how to install and integrate your DataStax environment with Centrify on a Linux machine running RedHat Enterprise Linux (RHEL) 6.0.

You must log on as root to complete the installation.

Note You must install the Enterprise edition of the DataStax environment. The standard edition does not support Kerberos authentication.

Install DataStax

1. Install DataStax.

   ```
   ./ DataStaxEnterprise-4.8.2015071416-linux-x64-installer.run
   ```

2. Accept the default configurations.

   You will need to change some of the configuration settings later in the installation process.

3. Start Cassandra.

   In `/usr/share/dse/bin`, run the following commands:

   ```
   service dse start
   ```

4. Ensure that the Cassandra service is running.

   ```
   ps -ef | grep dse
   ```

Install the Centrify agent on each node

When you install the agent on each node, you ensure that users are authorized to access the node through their Active Directory account.

1 Download the appropriate `.tgz` file.

2 Unzip and extract the agent package.

For example:

```
gunzip centrify-suite-2015.1-rhel4-x86_64.tgz
tar -xvf centrify-suite-2015.1-rhel4-x86_64.tgz
```

You should now see two files; an `.rpm` file and an executable file.

3 Install the Centrify Agent.

For example: `rpm -Uvh centrifydc-5.2.3-rhel4-x86_64.rpm`

4 Install the Centrify Enterprise Agent.

```
./install.sh -ent-suite
```

5 Run the executable file to ensure that you can join to your Active Directory domain.

In the following example, our domain controller is `dd-demo.test`:

```
./adcheck-rhel4-x86_64 dd-demo.test
```

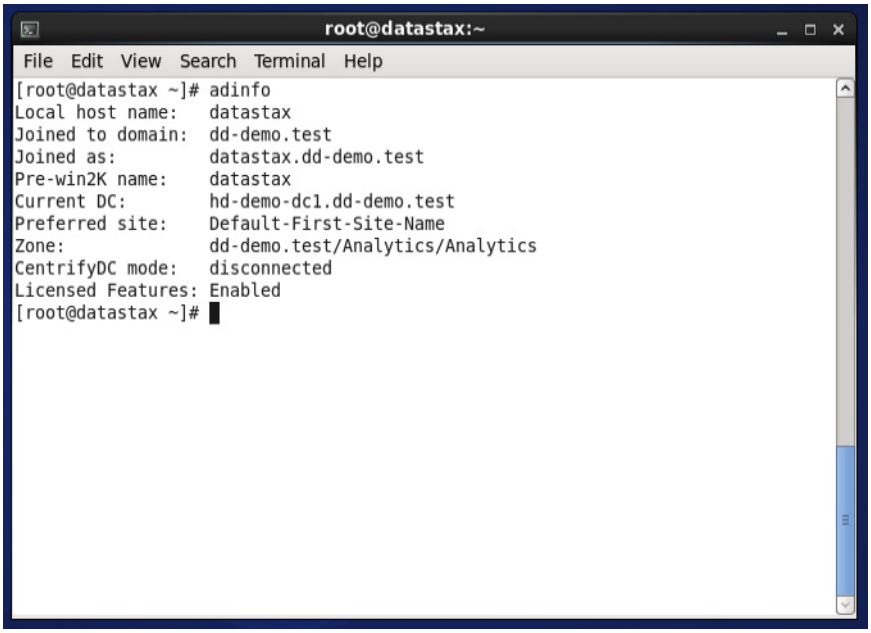
Note You should not see an error prompt if the machine is able to connect to an available domain controller.

6 Join to the Active Directory domain.

In the following example, the zone is `Analytics`, the user is `datastax`, -V displays debugging information, and the domain is `dd-demo.test`.

```
adjoin -z Analytics -u datastax -V dd-demo.test
```

7 You can use the `adinfo` command to view the status of your connection.



Configuring DataStax and Kerberos

By integrating the Centrify agent with DataStax, you can centrally create, secure, and distribute the service accounts and Kerberos key table (`keytab`) files that you require for distributed computing. The service accounts are stored securely in Active Directory with the domain controller acting as the Kerberos key distribution center (KDC).

Create the Kerberos keytab file

Create the Kerberos keytab file and service account.

For example:

```
sudo adkeytab -V -n -u unix.admin -K /etc/dse2.keytab -U dse2/mrhel6-7.dd-
demo.test@DD-DEMO.TEST -P dse2/mrhel6-7.dd-demo.test -c ou=NoSQL -S dse2
dse2
```

In this example, the variables are defined in the following ways:

`-n`: creates a new account.

`-u`: the user name.

`-K`: the path for the keytab file.

`-U`: the user principal name.

`-P`: the service principle name.

`-c`: the organizational unit or container for NoSQL.

`-S`: the account surname.

In this example the service account is dse2.

Get the Kerberos ticket (TGT/TGS) and authorize Kerberos

1. Get the TGT.

   ```
   /usr/share/centrifydc/kerberos/bin/kinit -kt/etc/ dse2.keytab dse2/mrhel6-7.dd-demo.test@DD-DEMO.TEST
   ```

   To view the TGT information, enter the following:

   ```
   /usr/share/centrifydc/kerberos/bin/klist
   ```

2. Get the TGS.

   ```
   /usr/share/centrifydc/kerberos/bin/kinit -kt/etc/ dse2.keytab -S dse2/mrhel6-7.dd-demo.test@DD-DEMO.TEST des2
   ```

   To view the TGS information, enter the following:

   ```
   /usr/share/centrifydc/kerberos/bin/klist
   ```

3. Move the generated keytab file to `/var/lib/dse2`.

   ```
   mv/etc/dse2.keytab /var/lib/dse2/dse2.keytab
   ```

4. Change the owner of the keytab to `cassandra`.

   ```
   chown cassandra:cassandra /var/lib/dse2/dse2.keytab
   ```

Configure Cassandra

Before you install the Java Cryptography Extension Unlimited Jurisdiction Policy Files, ensure that Cassandra is running with the Oracle JDK.

Download and install the JCE Unlimited Strength Jurisdiction Policy Files

1. Download the JCE Unlimited Strength Jurisdiction Policy Files from the Oracle Java SE download page at http://www.oracle.com/technetwork/java/javase/downloads/index.html.

2. Unzip the file.

3 Copy `local_policy.jar` and `US_export_policy.jar` to the `$JAVA_HOME/jre/lib/security` directory. This will overwrite the existing jar files.

Create the Kerberos user in the Cassandra database

1 In `/etc/dse/cassandra/cassandra.yaml`, set the authenticator to `PasswordAuthenticator.`

```
authenticator: org.apache.cassandra.auth.PasswordAuthenticator
```

2 Restart the node.

3 In `/usr/share/dse/bin`, log on to Cassandra. The default user name and password are both "cassandra."

```
./cqlsh -u cassandra -p cassandra
```

4 Create a Cassandra super user with the same name as the Kerberos users.

For example:

```
cqlsh>CREATE USER 'dse2@DD-DEMO.TEST' WITH PASSWORD 'dse2' SUPERUSER;
```

5 Secure the DSE nodes.

In `/etc/dse/cassandra/cassandra.yaml`, set the authenticator to Kerberos.

```
authenticator: com.datastax.bdp.cassandra.auth.KerberosAuthenticator
```

6 In `/etc/dse/dse.yaml` change the following Kerberos configurations:

keytab: `/var/lib/dse2/dse2/keytab`

service_principal: `dse2/_HOST@DD-DEMO.TEST`

http_principal: `dse2/_HOST@DD-DEMO.TEST`

qop: `auth`

Note Leave `_HOST` in the configuration parameters above. DataStax Enterprise automatically substitutes the Fully Qualified Domain Name (FQDN) of the host that runs it. This principal must have credentials in the keytab file that must be readable by the user running Cassandra.

7 Restart Cassandra to update the `yaml` files.

```
service dse restart
```

Configure cqlsh with Kerberos

The following section requires that you download and install Python pip, PyKerberos, and python-pure-sasl. For more information on using cqlsh with Kerberos, see

http://docs.datastax.com/en/datastax_enterprise/4.8/datastax_enterprise/sec/secUseCqlshKerb.html.

1. Install the latest Python pip version from https://pypi.python.org/pypi/pip#downloads

   For installation instructions, go to https://pip.pypa.io/en/stable/installing/

2. Install pure-sasl.

   ```
   sudo pip install pure-sasl
   ```

3. Install python-kerberos

   ```
   sudo yum install python-kerberos
   ```

4. Edit `~/.cassandra/cqlshrc` using your host IP address.

   ```
   [connection]
   hostname = 10.0.2.15
   port = 9042


   [kerberos]
   hostname = 10.0.2.15
   service = dse2
   ```

5. Start Cassandra.

   ```
   /usr/share/dse/bin/cqlsh
   ```

   Make sure your Kerberos ticket (TGT/TGS)is still valid.

   If your ticket has expired, repeat the steps to Get the Kerberos ticket (TGT/TGS) and authorize Kerberos.

6. Restart the Cassandra.

Make sure you can run the `cqlsh` command. There might be a slight delay after restarting the service before you can issue the `cqlsh` command.

For additional information on securing Datastax nodes, go to http://docs.datastax.com/en/datastax_enterprise/4.8/datastax_enterprise/sec/secTOC.html