

Cisco Nexus Data Broker Deployment Guide, Release 3.10.x

First Published: 2021-01-08

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Trademarks



Trademarks

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS REFERENCED IN THIS DOCUMENTATION ARE SUBJECT TO CHANGE WITHOUT NOTICE. EXCEPT AS MAY OTHERWISE BE AGREED BY CISCO IN WRITING, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENTATION ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

The Cisco End User License Agreement and any supplemental license terms govern your use of any Cisco software, including this product documentation, and are located at: https://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html. Cisco product warranty information is available at https://www.cisco.com/c/en/us/products/warranty-listing.html. US Federal Communications Commission Notices are found here https://www.cisco.com/c/en/us/products/us-fcc-notice.html.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

The Java logo is a trademark or registered trademark of Sun Microsystems, Inc. in the U.S. or other countries.

© 2021 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Trademarks ii

CHAPTER 1

Overview 1

About Cisco Nexus Data Broker 1

Cisco Nexus Data Broker Hardware and Software Interoperability Matrix 2

Python Activator Scripts for NX-OS Images 2

System Requirements 2

CHAPTER 2

Deploying Cisco Nexus Data Broker in Embedded Mode 5

Obtaining the Cisco Nexus Data Broker Embedded Software for NX-API 5

Installing and Activating the Cisco Nexus Data Broker Embedded Software for NX-API Mode for NX-OS 9.3(1) or Later 6

Upgrading to Release 3.10 for Cisco NXOS Release 9.3(1) or Later 8

CHAPTER 3

Deploying Cisco Nexus Data Broker Software in Centralized Standalone Mode 11

Installing or Upgrading the Cisco Nexus Data Broker Software in Centralized Mode 11

Installing the Cisco Nexus Data Broker Software in Centralized Mode 11

Upgrading the Application Software in Centralized Mode Using CLI 13

Upgrading the Application Software in Centralized Mode Using GUI 15

Starting the Application 16

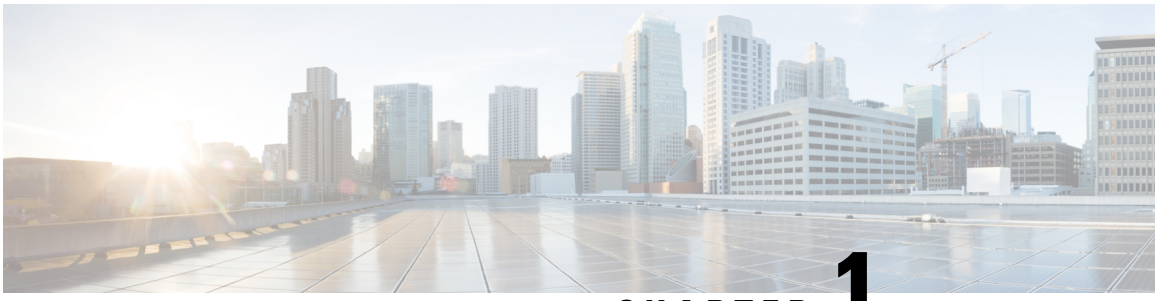
Verifying The Application Status 17

CHAPTER 4

Deploying Cisco Nexus Data Broker Software in Clusters 19

Installing a Cisco Nexus Data Broker Cluster 19

Upgrading a Cisco Nexus Data Broker Cluster 21



CHAPTER 1

Overview

This chapter gives us an overview of the Cisco NDB.

- About Cisco Nexus Data Broker , on page 1
- Cisco Nexus Data Broker Hardware and Software Interoperability Matrix , on page 2
- Python Activator Scripts for NX-OS Images, on page 2
- System Requirements, on page 2

About Cisco Nexus Data Broker

Visibility into application traffic has traditionally been important for infrastructure operations to maintain security, troubleshooting, and compliance and perform resource planning. With the technological advances and growth in cloud-based applications, it has become imperative to gain increased visibility into the network traffic. Traditional approaches to gain visibility into network traffic are expensive and rigid, making it difficult for managers of large-scale deployments.

Cisco Nexus Data Broker (NDB) with Cisco Nexus Switches provides a software-defined, programmable solution to aggregate copies of network traffic using Switched Port Analyzer (SPAN) or network Test Access Point (TAP) for monitoring and visibility. As opposed to traditional network taps and monitoring solutions, this packet-brokering approach offers a simple, scalable and cost-effective solution that is well-suited for customers who need to monitor higher-volume and business-critical traffic for efficient use of security, compliance, and application performance monitoring tools.

With the flexibility to use a variety of Cisco Nexus Switches and the ability to interconnect them to form a scalable topology provides the ability to aggregate traffic from multiple input TAP or SPAN ports, and replicate and forward traffic to multiple monitoring tools which may be connected across different switches. Using the Cisco NX-API agent to communicate to the switches, Cisco Nexus Data Broker provides advance features for traffic management.

Cisco NDB provides management support for multiple disjointed Cisco NDB networks. You can manage multiple Cisco NDB topologies that may be disjointed using the same application instance. For example, if you have 5 data centers and want to deploy an independent solution for each data center, you can manage all 5 independent deployments using a single application instance by creating a logical partition (network slice) for each monitoring network.

Cisco Nexus Data Broker Hardware and Software Interoperability Matrix

See *Cisco Nexus Data Broker Release Notes, Release 3.10* for the latest matrix.

Python Activator Scripts for NX-OS Images

The following table lists the Python Activator scripts and corresponding NX-OS Image names:

Note The activator scripts are availble for download at: https://github.com/datacenter/nexus-data-broker.

Note Check the Guestshell version using the **show guestshell** command. If the Guestshell version is 2.2 or earlier, either upgrade the Guestshell or destroy and re-run the script to start NDB embedded.

Table 1: Python Activator Scripts for NX-OS Images

Python activator script file name	NX-OS Image
NDBActivator4.0_9.3_plus.py	Cisco NXOS version 9.3(1) and above.

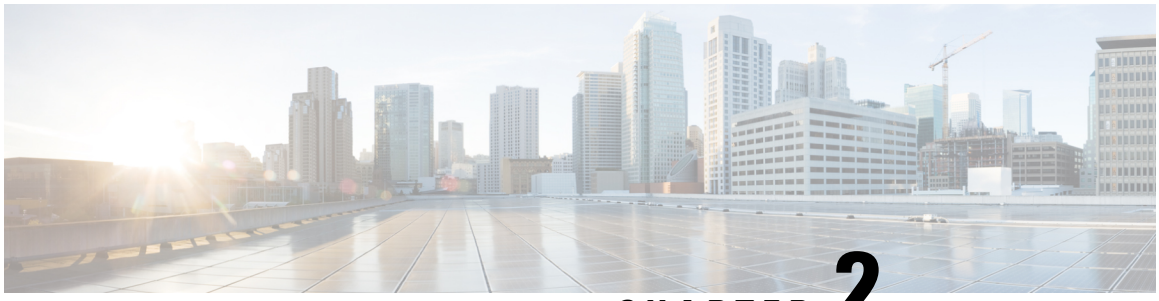
System Requirements

The following table lists the system requirements as per the deployment size:

Table 2: System Requirements per Deployment Size

Description	Small	Medium	Large
CPUs (virtual or physical)	6-core	12-core	18-core
Memory	8 GB RAM	16 GB RAM	24 GB RAM
Number of switches for TAP and SPAN aggregation	Upto 25 switches	Upto 50 switches	75 to 100 switches
Hard disk	Minimum of 40 GB of free space available on the partition on which the Data Broker software is installed.		
Operating System	A recent 64-bit Linux distribution that supports Java, preferably Ubuntu, Fedora, or Red Hat.		

Description	Small	Medium	Large
Other	Java Virtual Machine 1.8.		



CHAPTER 2

Deploying Cisco Nexus Data Broker in Embedded Mode

This chapter contains details of procedures for deploying Cisco Nexus Data Broker on Nexus series switches.

Before your proceed with the upgrade/ install procedures in this chapter, compare the **md5sum** between the NDB CCO image and image file copied to linux. Use the following command to check (linux):

```
cisco@NDB-virtual-machine:~/3.10.0/$ md5sum ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip
Displayed output: 685a0fe58f3280590dd878b7835241aa  ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip
```

Generating TLS certificate between NDB server and NDB switch for Embedded mode of deployment is not supported. For details about TLS, see the *Managing TLS Certificate, KeyStore and TrustStore Files* chapter in the *Cisco Nexus Data Broker Configuration Guide*.

- Obtaining the Cisco Nexus Data Broker Embedded Software for NX-API, on page 5
- Installing and Activating the Cisco Nexus Data Broker Embedded Software for NX-API Mode for NX-OS 9.3(1) or Later, on page 6
- Upgrading to Release 3.10 for Cisco NXOS Release 9.3(1) or Later, on page 8

Obtaining the Cisco Nexus Data Broker Embedded Software for NX-API

Procedure

Step 1 In a web browser, navigate to Cisco.com.

Step 2 In the center pane, click **Cloud and Systems Management**.

Step 3 If prompted, enter your Cisco.com username and password to log in.

Step 4 In the right pane, click **Network Controllers and Applications**, and then click **Cisco Nexus Data Broker.**

Step 5 Download and unzip the **Cisco Nexus Data Broker Release 3.10** application bundle zip file.

Step 6 Download activator script. The Python activator script needed to activate the NDB is available at: https://github.com/datacenter/nexus-data-broker.

JRE is unbundled from NDB 3.10 release package. Activator script will take care of installing the packages if internet connectivity is available on the switch; else, JRE and unzip packages need to be downloaded from an authorized location and provided as an argument for the activator script.

What to do next

Install the software on supported Cisco Nexus series switches. See *Cisco Nexus Data Broker Release 3.10, Release Notes* for the list of supported platforms.

Installing and Activating the Cisco Nexus Data Broker Embedded Software for NX-API Mode for NX-OS 9.3(1) or Later

Cisco NDB is not installed as OVA, it is installed in the Guestshell. To install Cisco Nexus Data Broker Embedded software on NXOS 9.3(1) or later release, use the NDB activator script, NDBActivator4.0_9.3_plus.py.

The activator script performs the following functions:

- Resizes the Guestshell resources.
- Unzips and places the ndb folder into the `/usr/bin` directory.
- Configures the Guestshell to management VRF.



Note Cisco NDB Release 3.10 supports Oracle Java only.

Before you begin



Note By default, you cannot install a new version of the Cisco Nexus Data Broker Embedded if you already have an existing Cisco Nexus Data Broker Embedded application installed and active. You can forcefully run the activator script even if it is already activated using **python bootflash** command with **--force** attribute. For example:

```
Syntax: python <file path>NDBActivator4.0_9.3_plus.py -v guestshell+ <zip file path> --force

Example:  python bootflash:NDBActivator4.0_9.3_plus.py -v guestshell+
/bootflash/ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip --force
```



Note To uninstall NDB application, destroy the Guestshell using the command, **guestshell destroy**. If an instance of NDB exists in the guestshell and you need to install a new NDB instance, you need to destroy the existing Guestshell and re-install the Guestshell and the NDB.

```
N9K-switch# guestshell destroy
```

Note After you disable and enable NXAPI mode, you should reconfigure **nxapi use-vrf management** command on the node.

Important Ensure that you have sufficient space available in the bootflash. The **ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip** file require a total of ~600 MB of space in the bootflash (/volatile folder) for the decompression processes. The script runs only on NXOS platform, version 9.3(1), with memory greater than 8GB.

Procedure

Step 1 switch# **copy** ftp:/10.10.10.1 **NDBActivator4.0_9.3_plus.py bootflash:vrf management**

Copies the NDBActivator4.0_9.3_plus.py from the directory where you downloaded it to the switch. You can download the file from different sources such as HTTP, FTP, or SSH.

Step 2 switch# **copy** ftp://10.10.10.1 **ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip bootflash:vrf management**

Copies the Cisco Nexus Data Broker Embedded package from the directory where you downloaded it to the switch. You can download the file from different sources such as HTTP, FTP, or SSH.

Step 3 switch# **show virtual-service list**

Monitors the status of the copy processes.

Step 4 switch# **guestshell enable**

Enables the guestshell.

Step 5 switch# **python bootflash:NDBActivator4.0_9.3_plus.py -v guestshell+ /bootflash/ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip**

Installs the Cisco Nexus Data Broker Embedded package on the switch.

Note
Internet connectivity is needed for downloading the packages.

When internet connectivity is not available on the switch, follow these steps:

a) Download JRE tar.gz file.

Download Java from an authorized site. Click `Linux x64` to download JRE tar.gz.

b) Download (from an authorized site) and unzip RPM package.

c) Copy JRE tar.gz file from the directory, where you downloaded it to the switch, using **copy scp://10.10.10.1 jre-8u271-linux-x64.tar.gz bootflash: vrf management**.

Replace **jre-8u271-linux-x64.tar.gz** with the downloaded file name.

d) Copy `unzip rpm` package from the directory where you downloaded it, to the switch, using **copy scp://10.10.10.1 unzip-6.0-21.el7.x86_64.rpm bootflash:vrf management**.

Replace `unzip-6.0-21.el7.x86_64.rpm` with the downloaded file name.

e) switch# **python bootflash:NDBActivator4.0_9.3_plus.py -v guestshell+ /bootflash/ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip /bootflash/jre-8u271-linux-x64.tar.gz /bootflash/unzip-6.0-21.el7.x86_64.rpm**

Step 6 switch# **show virtual-service list**

Monitors the status of the installations.

Note
To start the NDB application, use the **guestshell enable** command. If the NDB application is initiated through the Python script, guestshell is enabled automatically.

Note
To stop the NDB application, use the **guestshell disable** command. Use **guestshell enable** command to enable NDB.

Note
Do not continue until installation completes successfully. NDB application starts after it is installed successfully.

Step 7 switch# **show processes cpu sort | grep java**

Example:

switch# show processes cpu sort | grep java

19587 3 6 551 0.00% java

switch#

Verify whether NDB installed and initiated successfully.

What to do next

Log in to the Cisco Nexus Dashboard Data Broker GUI using HTTPS. The default HTTPS web link for the Cisco Nexus Dashboard Data Broker GUI is `https://IP_address:8443/monitor`. Enter the username and password (default values are admin/admin).

Upgrading to Release 3.10 for Cisco NXOS Release 9.3(1) or Later

This process involves using the GUI to download the configuration, perform the upgrade, and then upload the configuration. The process is applicable for Release 3.10 and later.

Note Cisco NDB Release 3.10 supports Oracle Java only.

For detailed information about upgrading Cisco Nexus OS from I4(6) to I6(1) and I7(1), see *Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide, Release 9.3(x)*.

Procedure

Step 1 Navigate to **Administration** > **Backup/Restore** in the running embedded NDB 3.6 or later version. Click **Backup and Backup Locally** to download the configuration in a zip file format.

Step 2 Copy the NDBActivator4.0_9.3_plus.py from the directory where you downloaded it to the switch. You can download the file from different sources such as HTTP, FTP, or SSH.

Example:

```
switch# copy scp://10.10.10.1 NDBActivator4.0_9.3_plus.py bootflash:vrf management
```

Step 3 Copy the Cisco Nexus Data Broker Embedded package from the directory where you downloaded it to the switch. You can download the file from different sources such as HTTP, FTP, or SSH.

Example:

```
switch# copy scp://10.10.10.1 ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip bootflash:vrf management
```

Step 4 Monitor the status of the copy processes.

Example:

```
switch# show virtual-service list
```

Step 5 Enable the guestshell.

Example:

```
switch# guestshell enable
```

Step 6 Install the Cisco Nexus Data Broker Embedded package on the switch.

Example:

```
switch# python bootflash:NDBActivator4.0_9.3_plus.py -v guestshell+
/bootflash/ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip
```

When internet connectivity is not available on the switch, follow these steps:

a) Download JRE tar.gz file.

Download Java from an authorized site. Click `Linux x64` to download JRE tar.gz.

b) Download (from an authorized site) and unzip RPM package.

c) Copy JRE tar.gz file from the directory, where you downloaded it to the switch, using **copy scp://10.10.10.1 jre-8u271-linux-x64.tar.gz bootflash: vrf management**.

Replace **jre-8u271-linux-x64.tar.gz** with the downloaded file name.

d) Copy `unzip rpm` package from the directory where you downloaded it, to the switch, using **copy scp://10.10.10.1 unzip-6.0-21.el7.x86_64.rpm bootflash:vrf management**.

Replace `unzip-6.0-21.el7.x86_64.rpm` with the downloaded file name.

e) switch# **python bootflash:NDBActivator4.0_9.3_plus.py -v guestshell+ /bootflash/ndb1000-sw-app-emb-9.3-plus-k9-3.10.0.zip /bootflash/jre-8u271-linux-x64.tar.gz /bootflash/unzip-6.0-21.el7.x86_64.rpm**

Step 7 Monitor the status of installation process.

Example:

```
switch# show virtual-service list
```

To stop the NDB application, use the guestshell disable command.

Note
Do not continue until installation completes successfully. NDB application starts after it is installed successfully.

Step 8 Verify whether NDB installed and initiated successfully.

Example:

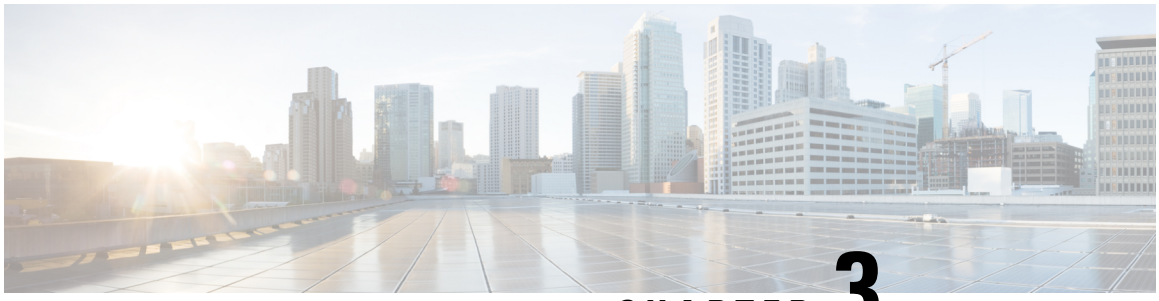
```
switch# show processes cpu sort | grep java
Example:
switch# show processes cpu sort | grep java
19587 3 6 551 0.00% java
```

Step 9 Copy the running configuration to the startup configuration.

Example:

```
switch(config)# copy running-config startup-config
```

Step 10 Upload Cisco NDB 3.10 configuration that you downloaded in Step 1 in the Cisco NDB user interface (UI). Navigate to **Administration** > **Backup Restore** > **Restore Locally**.



CHAPTER 3

Deploying Cisco Nexus Data Broker Software in Centralized Standalone Mode

This chapter contains details of procedures for installing and upgrading NDB in centralized mode.

Before your proceed with the upgrade/ install procedures in this chapter, compare the **md5sum** between the NDB CCO image and image file copied to linux. Use the following command to check (linux):

```
cisco@NDB-virtual-machine:~/3.10/$ md5sum ndb1000-sw-app-k9-3.10.0.zip
Displayed output: 518db25b4a89c996340c0316f72a6287 ndb1000-sw-app-k9-3.10.0.zip
```


- Installing or Upgrading the Cisco Nexus Data Broker Software in Centralized Mode, on page 11
- Starting the Application , on page 16
- Verifying The Application Status, on page 17

Installing or Upgrading the Cisco Nexus Data Broker Software in Centralized Mode

Before proceeding with the installation, check the System Requirements section in the Overview chapter of this guide.

In Cisco NDB Release 3.10, JRE is unbundled from the NDB package. You need to install Java on the NDB server before starting NDB 3.10.0.

Note Only Oracle Java is supported.

1. Install Java version greater then 1.8.0_45
2. Set JAVA_HOME environment variable; command to be used- export JAVA_HOME=(jre_path)

Installing the Cisco Nexus Data Broker Software in Centralized Mode

Complete these steps to install Cisco Nexus Data Broker software in Centralized mode:

Procedure

Step 1 In a web browser, navigate to **www.cisco.com**.

Step 2 Under **Support**, click **All Downloads**.

Step 3 In the center pane, click **Cloud and Systems Management**.

Step 4 If prompted, enter your Cisco.com **username** and **password** to log in.

Step 5 In the right pane, click **Network Controllers and Applications**, and then click **Cisco Nexus Data Broker.**

The file information for Release 3.10 is displayed: Cisco Nexus Data Broker Software Application: ndb1000-sw-app-k9-3.10.0.zip

Step 6 Download the Cisco Nexus Data Broker application bundle.

Step 7 Create a directory in your Linux machine where you plan to install Cisco Nexus Data Broker.

For example, in your Home directory, create `CiscoNDB`.

Step 8 Copy the Cisco Nexus Data Broker zip file into the directory that you created.

Step 9 Unzip the Cisco Nexus Data Broker zip file.

The Cisco Nexus Data Broker software is installed in a directory called `ndb`. The directory contains the following:

- `runndb.sh` file—The file that you use to launch Cisco Nexus Data Broker.
- `version.properties` file—The Cisco Nexus Data Broker build version.
- `configuration` directory—The directory that contains the Cisco Nexus Data Broker initialization files.

  This directory also contains the `startup` subdirectory where configurations are saved.
- `bin` directory—The directory that contains the following script:
  - `ndb` file—This script contains the Cisco Nexus Data Broker common CLI.
- `etc` directory—The directory that contains profile information.
- `lib` directory—The directory that contains the Cisco Nexus Data Broker Java libraries.
- `logs` directory—The directory that contains the Cisco Nexus Data Broker logs.

  Note
  The `logs` directory is created after the Cisco Nexus Data Broker application is started.
- `plugins` directory—The directory that contains the OSGi plugins.
- `work` directory—The webserver working directory.

  Note
  The `work` directory is created after the Cisco Nexus Data Broker application is started.

Step 10 Start the NDB application using `runndb.sh -start .`

Upgrading the Application Software in Centralized Mode Using CLI

Use the **upgrade** command to upgrade to Cisco NDB Release 3.10.0.

Note

- When you upgrade the software to Cisco Nexus Data Broker Release 3.2 or later release, the hostname should not be changed during the upgrade process. If the hostname is changed during the upgrade process, the upgrade might fail. If you are upgrading from release 2.x, 3.0 and 3.1, the domain name configuration in the switch should be removed before upgrading the software.

  In case, the upgrade was not successful because of the mismatch of the host name, use RMA to correct the configuration of the device. See the *RMA* section in the *Cisco Nexus Data Broker Configuration Guide*.

- When you run the **upgrade** command, the installation and the configuration are upgraded. However, any changes you made to the shell scripts or configuration files, for example, `config.ini`, are overwritten. After you complete the upgrade process, you must manually reapply your changes to those files.

- The latest NDB zip file must be extracted in an empty directory.

Hitless Upgrade

For hitless upgrade, configuration backup for releases prior to NDB Release 3.8, will always be the standard upgrade by re-configuring the devices during the upgrade.

Configuration backup taken for releases NDB 3.8 and after, will always be a Hitless upgrade using CLI.

Before you begin

- Backup up the NDB configuration. See the *Backup/ Restore* section in the *Cisco Nexus Data Broker Configuration Guide.*
- Back up your `config.ini` file.

Important You should manually backup your `config.ini` file before upgrading, because the backup process does not back them up for you. If you do not backup your files before upgrading, any changes you made will be lost.

Note When you run `runndb.sh` script, there is a thread in the script that monitors the log and the Cisco Nexus Data Broker JAVA process to monitor the health of the Cisco Nexus Data Broker. The default value for this option is 30 Seconds.

Procedure

Step 1 Backup your Cisco Nexus Data Broker release installation using your standard backup procedures.

Step 2 In a web browser, navigate to Cisco.com.

Step 3 Under **Support**, click **All Downloads**.

Step 4 In the center pane, click **Cloud and Systems Management**.

Step 5 In the right pane, click **Network Controllers and Applications**, and then click **Cisco Nexus Data Broker.**

Step 6 Download the Cisco NDB Release 3.10 applicable bundle: Cisco Nexus Data Broker Software Application—ndb1000-sw-app-k9-3.10.0.zip

Step 7 Create a temporary directory in your Linux machine where you plan to upgrade to Cisco NDB.

Step 8 Unzip the Cisco NDB Release 3.10 zip file into the temporary directory that you created.

Step 9 Navigate to the `ndb` directory that was created when you installed the Cisco Nexus Data Broker release (created in the previous step).

Step 10 Stop the running Cisco Nexus Data Broker instance.

Step 11 Navigate to the `ndb/bin` directory in the temporary directory that you created for Cisco NDB Release 3.10 upgrade software.

Step 12 Upgrade the application by entering the **./ndb upgrade --perform --target-home** {*ndb_directory_to_be_upgraded*} **[--verbose] [--backupfile** {*ndb_backup_location_and_zip_filename*}**]** command.

You can use one of the following options:

Option	Description
--perform --target-home {*ndb_directory_to_be_upgraded*}	Upgrades the Cisco ndb Monitor Manager intallation to Cisco NDB.
--perform --target-home {*ndb_directory_to_be_upgraded*} **--backupfile** {*ndb_backup_location_and_zip_filename*}	Upgrades the Cisco ndb Monitor Manager installation to Cisco NDB and creates a backup.zip file in the directory path that you set. **Note** • You must provide the name of the backup file and the .zip extension. • The backup file should not be saved in the ndb directory with current NDB installation or its subdirectory.
--verbose	Displays detailed information to the console. This option can be used with any other option and is disabled by default.
--validate --target-home {*ndb_directory_to_be_upgraded*}	Validates the installation.
./ndb help upgrade	Displays the options for the **upgrade** command.

Step 13 Navigate to the older folder (**xnc** in releases prior to Release 3.10) where you originally installed Cisco NDB. Rename the folder from **xnc** to **ndb**.

Step 14 Start the application processes using **runndb.sh -start**.

Upgrading the Application Software in Centralized Mode Using GUI



Note The latest NDB zip file must be extracted in an empty directory.

Hitless Upgrade

For hitless upgrade, configuration backup for releases prior to NDB Release 3.8, will always be the standard upgrade by re-configuring the devices during the upgrade.

Configuration backup taken for releases NDB 3.8 and after, will always be a Hitless upgrade using CLI.

Complete the following steps to upgrade the application software in the Centralized mode using GUI:

Procedure

Step 1 Login to the NDB GUI.

Step 2 Navigate to **Administration** > **Backup/Restore** to download the configuration in zip file format.

The default name of the zip file is **configuration_startup.zip**.

Step 3 Stop the current NDB instance using the **runxnc.sh -stop** command.

Example:

```
./runxnc.sh -stop
```

Step 4 If TLS certification is enabled between NDB server and NXOS switch, copy the tlsTrustStore and tlsKeyStore files to /ndb/configuration from the old ndb backup.

Step 5 In a web browser, navigate to Cisco.com.

Step 6 Under **Support**, click **All Downloads**.

Step 7 In the center pane, click **Cloud and Systems Management**.

Step 8 In the right pane, click **Network Controllers and Applications**, and then click **Cisco Nexus Data Broker.**

Step 9 Download the Cisco NDB Release 3.10 applicable bundle: Cisco Nexus Data Broker Software Application—ndb1000-sw-app-k9-3.10.0.zip

Step 10 Create a directory in your Linux machine where you plan to upgrade to Cisco NDB.

Step 11 Unzip the Cisco NDB Release 3.10 zip file into the directory that you created.

Step 12 Navigate to the `ndb` directory that was created when you installed the Cisco Nexus Data Broker release (created in the previous step).

Step 13 Start the new NDB installation using the **runndb.sh -start** command.

Example:

```
./runndb.sh -start
```

Step 14 Navigate to **Administration** > **Backup/Restore**.

Step 15 To reconfigure the device during the upgrade, select the **Restore** option (by checking the check-box) during configuration upload.

Step 16 Restart the new NDB instance using the **runndb.sh -restart** command.

Example:

```
./runndb.sh -restart
```

Starting the Application

Procedure

Note
When you are running NDB for the first time, the URL that you need to connect to and the port that it is listening on are displayed on the screen. For example, when you run the ./runndb.sh script, the following message is displayed on the screen: Web GUI can be accessed using below URL: *[https://<IP_address>: 8443]*.

Java 8 is required for NDB. Setup JAVA_HOME before NDB is started.

You can use one of the following options:

Option	Description
no option	
-jmxport *port_number*	Enables JMX remote access on the specified JVM port.
-debugport *port_number*	Enables debugging on the specified JVM port.
-start	
-start *port_number*	
-stop	
-restart	
-status	
-console	
-help	Displays the options for the **./runndb.sh** command.
-tls	To enable TLS, start the controller by entering the **./runndb.sh -tls -tlskeystore** *keystore_file_location* **-tlstruststore** *truststore_file_location* command.

Verifying The Application Status

Procedure

Step 1 Navigate to the `ndb` directory that was created when you installed the software.

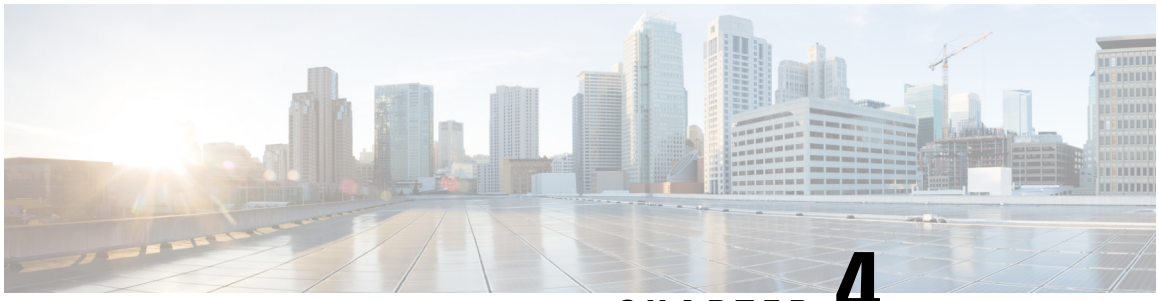
Step 2 Verify that the application is running by entering the **./runndb.sh -status** command.

The controller outputs the following, which indicates that the controller is running the Java process with PID 21680:

```
Controller with PID:21680 -- Running!
```

What to do next

Connect the switches to the controller. For more information, see the configuration guide for your switches.



CHAPTER 4

Deploying Cisco Nexus Data Broker Software in Clusters

This chapter contains the following details:

- Installing a Cisco Nexus Data Broker Cluster , on page 19
- Upgrading a Cisco Nexus Data Broker Cluster, on page 21

Installing a Cisco Nexus Data Broker Cluster

Use this procedure to install a Cisco NDB cluster.

Before you begin

Prerequisites:

- Cisco NDB supports 2-node and 3-node clusters.
- All IP addresses must be reachable and capable of communicating with each other.
- All switches in the cluster must connect to all the controllers.
- All controllers must have the same HA clustering configuration information in the `config.ini` files.
- All the NDB instances should be of the same NDB version to form the cluster.
- If using cluster passwords, all controllers must have the same password configured in the **ndbjgroups.xml** file. See *Password Protecting for HA Clusters* section in the *Cisco Nexus Data Broker Configuration Guide.*

Note All the NDB instances to form the cluster should be of the same NDB version.

Procedure

Step 1 In a web browser, navigate to www.cisco.com.

Step 2 Navigate to **Support** > **Products** > **Downloads**.

Step 3 In the **Find Products and Downloads** search box, enter *Nexus Data Broker* and click **Downloads** from the displayed list.

The file information for Release 3.10 is displayed: Cisco Nexus Data Broker Software Application: `ndb1000-sw-app-k9-3.10.0.zip`

Step 4 Download the Cisco Nexus Data Broker application bundle. If prompted, enter your Cisco.com username and password to login.

Step 5 Create a directory in your Linux machine where you plan to install Cisco Nexus Data Broker.

For example, in your Home directory, create `CiscoNDB`.

Step 6 Copy the Cisco Nexus Data Broker zip file to the created NDB directory.

Step 7 Unzip the Cisco Nexus Data Broker zip file.

The Cisco Nexus Data Broker software is installed in a directory called `ndb`. The directory contains the following:

- runndb.sh file—file to launch NDB.
- version.properties file—NDB build version.
- configuration directory—contains the NDB initialization files. This directory also contains the startup subdirectory where configurations are saved.
- bin directory—contains the NDB file that has the NDB common CLI.
- etc directory—contains profile information.
- lib directory—contains NDB Java libraries.
- logs directory—contains NDB logs.

  Note
  The logs directory is created after the NDB application is started.

- plugins directory—The directory that contains the NDB plugins.
- work directory—webserver working directory.

Step 8 Navigate to the ndb/configuration directory that was created when you installed the software.

Step 9 Use any text editor to open the `config.ini` file and locate the following text:

```
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
# supernodes=<ip1>;<ip2>;<ip3>;<ipn>
```

Step 10 Uncomment the line which consists of supernodes and replace <ip*> with NDB Server IPs.

```
IPv4 example:
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
supernodes=10.1.1.1;10.2.1.1;10.3.1.1

IPv6 example:
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
supernodes=2001:22:11::1;2001:33::44::1;2001:55:66::1
```

Step 11 Save the file and exit the editor.

Step 12 Repeat steps 5 to 11 in all the Linux machines where the NDB is installed.

Step 13 Start the Primary NDB server using the **./runndb.sh -start** command.

Step 14 After the GUI of the primary NDB server is up, start the other NDB servers using the **./runndb.sh -start** command.

Upgrading a Cisco Nexus Data Broker Cluster

Before you begin

Prerequisites:

- Cisco NDB supports 2-node and 3-node clusters. If you configure even number of nodes, the last node is not included in the cluster formation, ensuring odd number of nodes in a setup.
- All IP addresses must be reachable and capable of communicating with each other.
- All switches in the cluster must connect to all the controllers.
- All controllers must have the same HA clustering configuration information in the `config.ini` files.
- All the NDB instances should be of the same NDB version to form the cluster.
- If using cluster passwords, all controllers must have the same password configured in the **ndbjgroups.xml** file. See *Password Protecting for HA Clusters* section in the *Cisco Nexus Data Broker Configuration Guide.*

Note All the NDB instances to form the cluster should be of the same NDB version.

Procedure

Step 1 Login to the NDB primary server.

Step 2 Navigate to **Administration** > **Backup/ Restore**.

Step 3 Click **Backup Locally** to download the configuration file.

Step 4 Stop all NDB instances using the **runxnc.xh -stop** command.

Step 5 If TLS certification is enabled between NDB server and NDB Devices, take backup of the tlsTrustStore and tlsKeyStore files from `/xnc/configuration`.

Step 6 Perform the previous step on all the NDB cluster instances.

Step 7 In a web browser, navigate to www.cisco.com.

Step 8 Navigate to **Support** > **Products** > **Downloads**.

Step 9 In the Find Products and Downloads search box, enter “Nexus Data Broker” and click on ‘Downloads’ from search response list.

The file information for Release 3.10 is displayed: Cisco Nexus Data Broker Software Application: `ndb1000-sw-app-k9-3.10.0.zip`

Step 10 Download the Cisco Nexus Data Broker application bundle. When prompted, enter your Cisco.com username and password to login.

Step 11 Create a directory in your Linux machine where you plan to install Cisco NDB.

For example, in your Home directory, create CiscoNDB.

Step 12 Copy the Cisco Nexus Data Broker zip file to the directory that you have created.

Step 13 Unzip the Cisco Nexus Data Broker zip file.

The Cisco Nexus Data Broker software is installed in a directory called `ndb`. The directory contains the following:

- runndb.sh file—file to launch NDB.
- version.properties file—NDB build version.
- configuration directory—contains the NDB initialization files. This directory also contains the startup subdirectory where configurations are saved.
- bin directory—contains the NDB file that has the NDB common CLI.
- etc directory—contains profile information.
- lib directory—contains NDB Java libraries.
- logs directory—contains NDB logs.

  Note
  The logs directory is created after the NDB application is started.

- plugins directory—The directory that contains the NDB plugins.
- work directory—webserver working directory.

Step 14 Navigate to the ndb/configuration directory that was created when you installed the software.

Step 15 Use any text editor to open the `config.ini` file and locate the following text:

Step 16 Locate th following text:

```
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
# supernodes=<ip1>;<ip2>;<ip3>;<ipn>
```

Step 17 Uncomment the line which consists of supernodes and replace <ip*> with NDB Server IPs.

```
IPv4 example:
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
supernodes=10.1.1.1;10.2.1.1;10.3.1.1

IPv6 example:
# HA Clustering configuration (semi-colon-separated IP addresses of all controllers that are part
of the cluster.)
supernodes=2001:22:11::1;2001:33::44::1;2001:55:66::1
```

Step 18 Save the file and exit the editor.

Step 19 Repeat the steps 7 to 18 in all the Linux machines where the NDB is installed.

Step 20 Start the Primary NDB server using the **./runndb.sh -start** command.

Step 21 After the GUI of the Primary NDB Server is up, start the other NDB servers using the **./runndb.sh -start** command.

Step 22 Login to the Primary Server NDB GUI.

Step 23 Navigate to **Administration** > **Backup/Restore** > **Actions** > **Restore Locally** and upload the configuration you had earlier downloaded.

Step 24 Stop all instances of NDB in the cluster using the **./runndb.sh -stop** command.

Step 25 If TLS certification is enabled between NDB server and NDB switches, copy the tls TrustStore and tlSKeyStore files to `ndb/configuration` taken from the step 5 for all NDB instances.

Step 26 Start the primary NDB server using the **./runndb.sh -start** command.

If TLS certification is enabled, use below commands to start NDB servers.

```
./runndb.sh -tls -tlskeystore ./configuration/tlsKeyStore -tlstruststore ./configuration/tlsTrustStore
bin/ndb config-keystore-passwords --user <NDB_username> --password <NDB_password> --url
https://<Cluster_NDB_IP>:8443 --verbose --prompt --keystore-password <keystore-password>
--truststore-password <truststore-password>
```

Step 27 Start the member NDB server(s) using the **./runndb.sh -start** command.

If TLS certification is enabled, use below commands to start NDB servers.

```
./runndb.sh -tls -tlskeystore ./configuration/tlsKeyStore -tlstruststore ./configuration/tlsTrustStore
bin/ndb config-keystore-passwords --user <NDB_username> --password <NDB_password> --url
https://<Cluster_NDB_IP>:8443 --verbose --prompt --keystore-password <keystore-password>
--truststore-password <truststore-password>
```