

Logging in and Managing the Cisco Nexus Dashboard Data Broker

This chapter has details about logging in and managing the Cisco Nexus Dashboard Data Broker, and overview of the GUI.

Beginning with Release 3.10.1, Cisco Nexus Data Broker (NDB) has been renamed to Cisco Nexus Dashboard Data Broker. However, some instances of NDB are present in this document, to correspond with the GUI, and installation folder structure. References of NDB/ Nexus Data Broker/ Nexus Dashboard Data Broker can be used interchangeably.

- Configuring High Availability Clusters , on page 1
- Changing Controller Access, on page 5
- Cisco Nexus Dashboard Data Broker GUI Overview, on page 5
- Syslog, on page 8

Configuring High Availability Clusters

Cisco Nexus Dashboard Data Broker supports high availability clustering in active/active mode with up to five controllers. To use high availability clustering with Cisco Nexus Dashboard Data Broker, you must edit the `config.ini` file for each instance of Cisco Nexus Dashboard Data Broker.

NDDB release 3.10.4 supports 3-node clusters only.

In case of a split-brain scenario, 3-node clusters are handled as follows:

Cluster health is indicated as Yellow. A minumum of two nodes should be *up* and connected in the cluster for the cluster to be in the operational state. If not, the cluster nodes will move to a non-operational state. Override option is not available. Fix the VM and/or network link, as required.

Note IPv6 is supported in centralized Nexus Dashboard Data Broker mode only, it is not supported in Embedded mode.

Table 1: Cluster Operation Status

Cluster Indicator	Cluster Status	Recommendation
Green	Operational	*No recommendation as the status is operational.*
Yellow	Some of the cluster nodes are not available	Do not make any changes or add to the existing Nexus Dashboard Data Broker configuration.
Red	The node is isolated from the cluster.	Do not make any changes or add to the existing Nexus Dashboard Data Broker configuration. Note: For two node cluster, you need to override in any one of the cluster node only, to ensure regular operation.

Before you begin

- All IP addresses must be reachable and capable of communicating with each other.
- All switches in the cluster must connect to all of the controllers.
- All controllers must have the same HA clustering configuration information in the `config.ini` files.
- All controllers must have the same information in the `ndb/configuration/startup` directory.
- If using cluster passwords, all controllers must have the same password configured in the `ndbjgroups.xml` file.
- To mark a node as a preferred primary, add the required node IP address as the first node in the list of supernodes in the `config.ini` file. If you would prefer to always have the first node as the preferred primary, then set `enablePreferredPrimary=true` in the `config.ini` file.

  To change the preferred primary node, *stop* the ndb controller and make the required modifications to the `config.ini` file.

  If you donot wish to mark the first node as the preferred primary from the list of super nodes, then change the parameter to *false* in the `config.ini` file:

  ```
  enablePreferredPrimary=false
  ```

  Now, in case of a failure, and the primary node is *down*, another member node takes over as the primary node. The member node will continue to be the primary node even when the former primary node is *up*.



Note The data broker controller checks for the number of configured supernodes (in the `config.ini` file), and if the number is less than three, an error is displayed indicating that 2-node cluster is not supported.

Procedure

Step 1 Open a command window on one of the instances in the cluster.

Step 2 Navigate to the `ndb/configuration` directory that was created when you installed the software.

Step 3 Use any text editor to open the `config.ini` file.

Step 4 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 5 **Example:**

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;10.4.1.1;10.5.1.1
```

Example:

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 6 Save the file and exit the editor.

Step 7 Repeat Step 3 through Step 7 for each instance of Cisco Nexus Dashboard Data Broker in the cluster.

Step 8 Restart Cisco Nexus Dashboard Data Broker.

For Nexus Dashboard Data Broker cluster deployment, the expected latency between the node is three seconds, with three retries. You can configure the latency time and the number of maximum retries. See the procedure, below.

What to do next

(Optional) Use this procedure to configure the delay time for a node and the number of retries.

1. Open a command window on one of the instances in the cluster.
2. Navigate to the ndb configuration directory.
3. Use any text editor to open the ndbjgroups.xml file.
4. Locate the following text:

   ```
   FD timeout="3000" max_tries="3"/
   ```

5. Modify the *Latency Time* value and *maximum_tries* value.
6. Save the file and exit the editor.
7. Repeat the above steps for all the instances of the cluster.

Password Protecting High Availability Clusters

Procedure

Step 1 Open a command window on one of the instances in the cluster.

Step 2 Navigate to the `ndb/configuration` directory.

Step 3 Use any text editor to open the `ndbjgroups.xml` file.

Step 4 Locate the following text:

```
<!-- <AUTH auth_class="org.jgroups.auth.MD5Token" auth_value="ciscoXNC" token_hash="MD5"></AUTH> -->
```

Step 5 Remove the comments from the AUTH line.

Example:

```
<AUTH auth_class="org.jgroups.auth.MD5Token" auth_value="ciscoXNC" token_hash="MD5"></AUTH>
```

Step 6 (Optional) Change the password in the auth_value attribute.

By default, the cluster is protected with the password "ciscoXNC". You can change this password to whatever value you want, you need make the similar changes on all machines in the cluster.

Step 7 Save the file and exit the editor.

Adding a Standby Node

Beginning with release 3.10.4, you can add a standby node to support a cluster. You need to append `-standby` to the IP address of the standby node to indicate the standby node while configuring the supernodes in the `config.ini` file. An example is shown here:

```
supernodes=<ip1>;<ip2>;<ip3>;<ip4>-standby
```

Release 3.10.4 supports 3-node clusters only. The cluster is said to be *fully healthy* if all the three nodes are working fine, *partially healthy* if two (out of the three nodes) are working fine. If only one node is working fine, that is, two nodes (out of the three nodes of a cluster) are *down*, then, the cluster is *unhealthy*. You need to manually `start` the standby node, which forms a cluster with the running node.

The standby node terminates automatically:

- If the running node gets terminated abruptly.
- After recovery, if the two nodes that were earlier *down* are now *up*.

Guidelines and limitations for standby nodes

- You cannot `start` the standby node, if all the nodes in a cluster are *down*. One node needs to be running to form a cluster with the standby node.
- You cannot `start` the standby node, if the cluster is healthy (that is, if two nodes in a three node cluster are working fine).
- Ensure that the cluster is unhealthy before starting the standby node. Ensure that the nodes of the cluster are *down*, and not a disruption in connection between the nodes.

Consider a scenario wherein, node 1 and node 2 are located together and node 3 and the standby node are located together elsewhere. If nodes 1 and 2 are temporarily disconnected from node 3 and the standby node, it gives a false impression that nodes 1 and 2 are *down*, and based on this information, if the standby node is started, then the standby node and node 3 form a cluster (even when nodes 1,2 are *up*). This will lead to a configuration mismatch/ loss when the connection between node 1- node 2 and node 3-standby node is restored.

Changing Controller Access

An unencrypted (HTTP) access to the GUI and the API to the controller access is disabled by default. You cannot access the controller with the URL http://<host>:8080.

To change the controller access to HTTP, complete the following steps:

Before you begin

Cisco Nexus Dashboard Data Broker is shipped with a certificate for the HTTPS connection between the Cisco Nexus Dashboard Data Broker and a browser. You can change to a different certificate.

The script **generateWebUIcertificate.sh** is available in the **ndb/configuration** folder. If you execute this script, it moves the shipped certificate to **old_keystore** and the new certificate is generated in **keystore**. On the next Cisco Nexus Dashboard Data Broker restart, this new certificate is used.

Procedure

Step 1 Remove the comment character from the connector for port 8080 in the `tomcat-server.xml` files in the configuration directory as displayed in the following example:

```
<Service name="Catalina">
<!--
<Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" server="Cisco NDB" enableLookups="false" />
-->
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
scheme="https" secure="true"
clientAuth="false" sslProtocol="TLS"
keystoreFile="configuration/keystore"
keystorePass="ciscondb" server="Cisco NDB"
connectionTimeout="60000" enableLookups="false" />
```

Step 2 Restart the controller.

Cisco Nexus Dashboard Data Broker GUI Overview

The Cisco Nexus Dashboard Data Broker GUI contains the following tabs and each of these tabs are discussed in detail (as separate chapters) in the subsequent pages of this guide.

- Dashboard

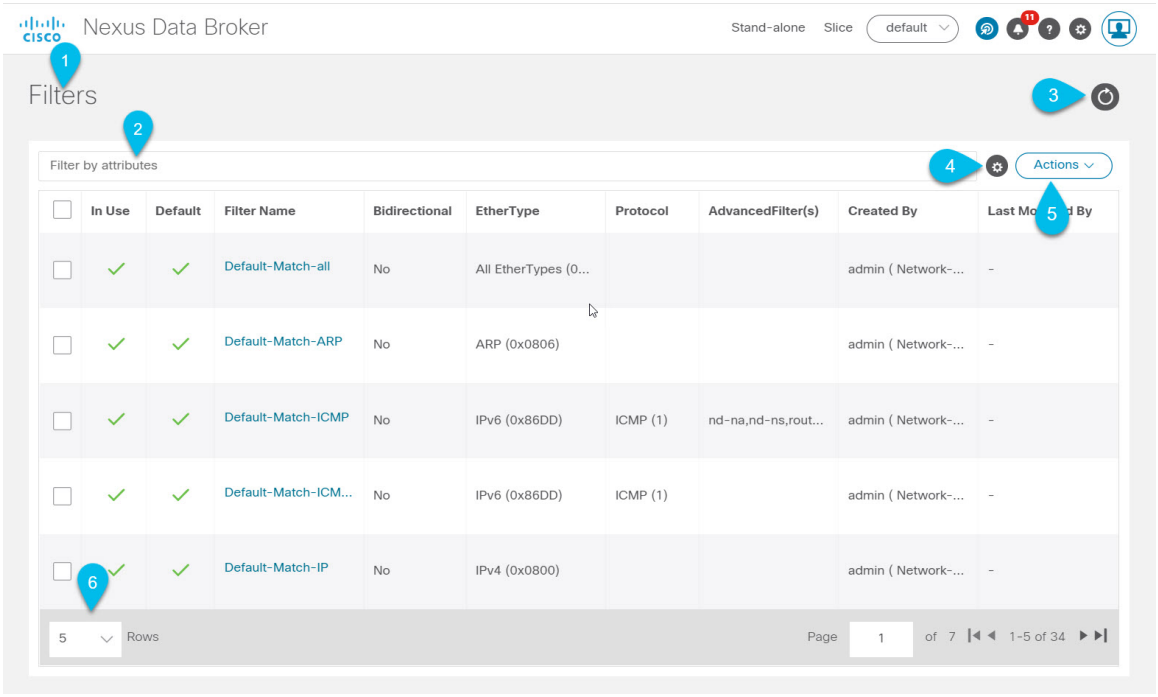
- Topology
- Devices
- Connections
- Components
- Sessions
- Statistics
- Troubleshooting
- Administration

For details about the header icons, see Header.

Components of a screen of the Cisco Nexus Dashboard Data Broker

When you click a tab/ sub-tab, the current information of the tab is displayed in a table.

A typical screen representing one of the tabs of the Release 3.10.1 Cisco Nexus Dashboard Data Broker GUI is displayed here:



- 1—Name of the tab/sub-tab.
- 2—Use the *Filter by attributes* bar to filter the displayed table which has the details of the selected tab. Choose the attribute, operator and filter-value.

  You can also filter the displayed table based on the *filter* icon that appears when you hover over an element of the table.

- 3—Use the *Refresh* icon to refresh the displayed details and get the latest information about the tab/ sub-tab.
- 4—Use the *Column Customization* icon to select the columns you want to see in the displayed table.
- 5—Click the **Actions** button to see the available actions for the screen.
- 6—From the **Rows** drop-down list, select the number of rows to be displayed in the table.

Header

This section provides an overview of the Cisco Nexus Dashboard Data Broker GUI Header (upper right corner) icons:

Table 2: Cisco Nexus Dashboard Data Broker Header Icons

Icon	Description
Slice	Displays the default slice.
Figure 1: Create 	Provides quick navigation to often-used configuration and administration procedures.
Figure 2: Alarm 	Displays the number of inconsistent NDB devices. Click the Alarm icon; you are directed to the Flow Management tab for details.
Figure 3: Help Menu-Bar 	Displays the following options: • **What's New**—displays new features for the latest release. • **Help**—displays online help content.
Figure 4: System Tools Menu-Bar 	Provides the following options: • **Application Properties**— set the TLS-related passwords. • **Northbound API**—takes you to the **Swagger** UI for details about Nexus Dashboard Data Broker REST APIs. • **About Nexus Dashboard Data Broker**—displays Nexus Dashboard Data Broker details such as, build and version.

Icon	Description
Figure 5: User Profile Menu-Bar 	Provides the following options: • **Welcome** *User*—displays the current user of the GUI.

Syslog

In the Nexus Dashboard Data Broker server backend, you can configure the `logback.xml` file to send logs to the Syslog server. You can customize the log format as per your requirement. The logback file is located at: `/ndb/configuration/logback.xml`

Note If the Nexus Dashboard Data Broker server(s) are running, restart the servers after the changes are made in the `logback.xml` file.

```
Sample Syslog configuration:

Add below config with respective Syslog server IP address and port number in logback.xml
file.
  <appender name="SYSLOG" class="ch.qos.logback.classic.net.SyslogAppender">
    <syslogHost>10.16.206.171</syslogHost>
    <facility>LOCAL7</facility>
    <port>514</port>
    <suffixPattern>[%thread] %logger %msg</suffixPattern>
  </appender>

Append "<appender-ref ref="SYSLOG" />" in root as shown below,
  <root level="error">
    <appender-ref ref="STDOUT" />
    <appender-ref ref="SYSLOG" />
    <appender-ref ref="ndb.log" />
  </root>
```

After an upgrade, these configuration changes in the `logback.xml` file are lost. Ensure to check and restore the configuration manually, after upgrading the controller to a newer Nexus Dashbaord Data Broker version.