**Chapter: Cisco Nexus Data Broker-Migration Tool [Open Flow to NXAPI]**

**Chapter Contents:**

**NDB Migration Tool Overview**

**Limitations**

**Deployment – Environment Setup**

**NDB Tool Usage**

**Troubleshooting**

**NDB Migration Tool Overview**

NDB Tool does Migration from current NDB version to NDB 3.6 along with the NDB switch conversion from OF to NXAPI

**Tool Feature**

* Single Touch Migration for OF to NXAPI devices
* Supports NDB version from NDB 3.5.
* Supports all NDB Platform devices.
* Supports Atomic & Non-Atomic operations.
* Multiple Device Upgrade in a single Migration job.

**OF to NXAPI Migration**

* NDB upgrade to NDB 3.6 and above
* Export the device configuration in NDB 3.6
* NDB configuration cleanup
* Device conversion from OF to NXAPI, device upgrade(if needed)
* Import the NXAPI device configuration in NDB 3.6

**Limitations**

* Portgroups are not supported. In case NDB has port groups configured, user has to manually configure portgroups and related configuration after NDB migration is done.
* Special characters in ports description is not supported. As a prerequisite, before running script user has to remove the special characters in ports description.(For any NDB version below 3.4)

**Deployment – Environment Setup**

**Prerequisites**

User with Administrative access

**Packages :**

* Python(2.7)
* Open SSL
* Pip(10.0.1)
* pexpect
* yaml
* Requests
* ExScript
* configobj
* Paramiko
* Git

**Steps for package installation for different Linux flavours**

* **Ubuntu**

Pip (version 10.0.1)

**Pip**

Sudo apt-get install pip

To install specific version:

pip install pip==10.0.1

Copy below packages to requirements.txt file

pexpect==4.6.0

pyyaml==3.12

Requests==2.18.4

ExScript==2.5.7

configobj==5.0.6

To install all packages present in requirement.txt use below command.

# pip install -r requirements.txt

**Git**

sudo apt-get install git

**Python**

Python version 2.7

sudo apt-get install python2.7

* **Redhat**

**Pip**

Pip (version 10.0.1)

Sudo yum install python-pip

**pip install -r requirements.txt**

**Note:**

In Redhat if you have packages with older version, you will get error in updating the packages.

So use the below command to update to mentioned version in requirements.txt file

**install pur package using**

pip install pur

**Then run the below command to update**

pur -r requirements.txt

**requirements.txt**

pexpect==4.6.0

pyyaml==3.12

Requests==2.18.4

ExScript==2.5.7

configobj==5.0.6

**Git**

yum install git

**Python**

Python version 2.7

sudo yum install python27

**Migration Script download**

* Clone the code from the below link in server.

git clone <http://ndb-build.cisco.com/gerrit/NDBMigration>

**NDB Tool Usage**

**Migration Script Input**

Navigate to script path

**Ex: /home/cisco/NDBMigration**

* Fill up the the device details in Input.yaml file . Input Yaml file will be under Utilities folder Ex: /home/cisco/NDBMigration/Utilities .Refer example\_migration.yaml under the same folder for more information.

**Ex:**

NDBserver detail

old\_path\_ndb\_build

new\_path\_ndb\_build

device detail(ip address, Nxos image name,ofa\_ova\_name,tcam configuration)

**Note: New ndb build must present in server (new\_path\_ndb\_build)**

* For Script run - python NDBMigration.py

**What happens when script fails?**

* If revertFlag is set to 1, upon failure in any steps, NDB and device configurations are reverted back to old NDB version along with OF device configurations.
* If revertFlag is set to 0, below is behavior.
* NDB upgrade failure – reverted back to initial NDB
* NDB export failure – reverted back to initial NDB
* NDB clean up failure – reverted back to initial NDB
* Device conversion failure – script run will continue and state for failed device will be “FAIL”
* NDB import failure – script run will continue and state for failed device will be “FAIL”.
* User can do a rerun when script run fails as below.

**python NDBMigration.py –rerun <failedjobid>**

**Ex: *python NDBMigration.py –rerun job.2018Aug07\_04:49:39***

* Rerun will create a new jobid and the script will start from the place where it failed.

**Troubleshooting:**

**Job Id:**

* Unique jobid folder created in script directory for every run.
* jobid folder contains three subfolder – “Backup”, “Log” and “Report”
* Backup contains old ndb zip file, export json, import json and state file
* Log folder contains script log file.
* Report folder contains report of script run

**State file**

* State file created for every run.
* Pass
* Fail
* Skip
* The above status maintained for every operation by NDB Migration script

**Steps to revert back to old Configuration for any failure with revertFlag set to 0.**

If revertFlag is set to 0 and user wants to revert back all configurations.

**Procedure**

**Step1**

Remove all interface configuration from all device.

**Step2**

Disable openflow on all device.

(config)# no feature openflow

**Step3**

Load previous nxos image which was loaded before running migration script on all device.

Ex:-

switch# install all system <system-img> kickstart <kickstart-img>

#install all nxos <img>

**Step4**

Install and activate openflow ova on all device.

Ex:-

#virtual-service install name ndb package bootflash:<openflow-ova>

**Step5**

Configure open flow configuration on all device

We can find old configuration from “Migration\_backup” file which will be present on each device.

Use following command to check “Migration\_backup” file.

#show file <file\_name>

Demo-3172# show file Migration\_backup\_2018Aug09\_07:15:29

Ex:-

#openflow

#switch 1

#pipeline 201

#probe-interval 5

#controller ipv4 10.16.206.136 port 6653 vrf management security none

#of-port interface Ethernet1/1-54

**Step6**

Configure tcam region which was present before running migration script.

**Step7**

Stop 3.6 NDB and Start older NDB which was running before starting migration.

**FAQ’s**

* **If user hit any proxy issue while adding device user has to unset proxy which is configured.**

Steps-

unset http\_proxy

unset https\_proxy

JAVA\_HOME variable needs to be set in NDB server in /etc/environment file.

export JAVA\_HOME=/opt/jdk1.8.0\_141

Ex:

ERROR - HTTPSConnectionPool(host='10.16.206.197', port=8443): Max retries exceeded with url: /monitor (Caused by ProxyError('Cannot connect to proxy.', error('Tunnel connection failed: 504 Gateway Timeout',)))

* **Where to run migration?**

User can run the migration script from a VM in which NDB is running or from a new VM(Ubuntu/Redhat). The recommended way is to go for a new VM**.**

* **What happens if user already has python packages with older version in Redhat?**

Use the below commands –

pip install pur

pur –r requirements.txt

* **How to check python version?**

Use the below command –

python –V

* **How to check pip version?**

Use the below command –

pip –V

* **How to check pip packages?**

Use the below command to check for the specific installed package with its version –

pip list

* **NDB Import Failure on N3K device**
* Check the switching mode of N3K :

N3K-123# show system switch-mode

system switch-mode n3k

* N3K-3132Q-40GX , N3K-3172PQ-10GE - Supports both N3K and N9K mode ( by default it’s N3K), NDB supports 3K mode