INSTALLING CISCO NSO

TRAINOCATE

Prerequisites

- NSO is supported on **Linux** and **OS X**.
- **ncs-VERSION.OS.ARCH.installer.bin**
  - **Version:** The version to install
  - **Operating System (OS):** The operating system (linux for all Linux distributions and darwin for OS X)
  - **Architecture (ARCH):** The CPU architecture (x86_64 or i686)

```
$ ls
nso-5.7.1.linux.x86_64.signed.bin
```

Prerequisites

- Java JDK-8.0 or higher is installed: java --version indicates “1.8” or higher
- Python 3.5 or higher is installed: python3 --version

```
$ sudo apt install openjdk-17-jre-headless
$ java –version
$ python3 --version
```

NSO Installation Types

2 installation types:

- **Local installation**: for development and testing
- **System installation**: for production and preproduction verification lab

System Installation

- **Step 1**: Unpack the signed.bin file

  sudo sh nso-5.7.1.linux.x86_64.signed.bin --skip-verification

```
osboxes@osboxes:~/module5/nso$ ls
nso-5.7.1.linux.x86_64.signed.bin
osboxes@osboxes:~/module5/nso$ sudo sh nso-5.7.1.linux.x86_64.signed.bin --skip-verification
Unpacking...

osboxes@osboxes:~/module5/nso$ ls -l
total 395480
-rw-r--r-- 1 root    root       15598 Mar 10  2021 cisco_x509_verify_release.py
-rw-r--r-- 1 root    root       15447 Mar 10  2021 cisco_x509_verify_release.py3
-rwxr-xr-x 1 root    root    202456323 Jan 26 07:40 nso-5.7.1.linux.x86_64.installer.bin
-rw-r--r-- 1 root    root         256 Jan 26 07:40 nso-5.7.1.linux.x86_64.installer.bin.signature
-rw-rw-r-- 1 osboxes osboxes 202469128 May 16 01:02 nso-5.7.1.linux.x86_64.signed.bin
-rw-r--r-- 1 root    root        4088 Jan 26 07:40 README.signature
-rw-r--r-- 1 root    root        1383 Mar 10  2021 tailf.cer
```

System Installation

- **Step 2**: Install NSO using -–system-install

  sudo sh nso-5.7.1.linux.x86_64.installer.bin --system-install

```
osboxes@osboxes:~/module5/nso$ sudo sh nso-5.7.1.linux.x86_64.installer.bin --system-install
INFO  Using temporary directory /tmp/ncs_installer.42668 to stage NCS installation bundle
INFO  Using /opt/ncs/ncs-5.7.1 for static files
INFO  Using /etc/ncs for configuration files
INFO  Using /var/opt/ncs for run-time state files
INFO  Using /var/log/ncs for log files
INFO  Unpacked ncs-5.7.1 in /opt/ncs/ncs-5.7.1
INFO  Found and unpacked corresponding DOCUMENTATION_PACKAGE
INFO  Found and unpacked corresponding EXAMPLE_PACKAGE
INFO  Found and unpacked corresponding JAVA_PACKAGE
INFO  Generating default SSH hostkey (this may take some time)
INFO  SSH hostkey generated
INFO  Environment set-up generated in /opt/ncs/ncs-5.7.1/ncsrc
INFO  NSO installation script finished
INFO  Found and unpacked corresponding NETSIM_PACKAGE
INFO  Generating keys for encrypted-strings
INFO  Configuring installation for PAM authentication
INFO  Using PAM service common-auth for authentication
INFO  Generating self-signed certificates for HTTPS
INFO  Installed init script /etc/init.d/ncs
INFO  Installed user profile script ncs.sh in /etc/profile.d
INFO  Installed user profile script ncs.csh in /etc/profile.d
INFO  Installed 'logrotate' configuration file ncs in /etc/logrotate.d

INFO  The installation has been configured for PAM authentication,
INFO  with group assignment based on the OS group database
INFO  (e.g. /etc/group file). Users that need access to NCS must
INFO  belong to either the 'ncsadmin' group (for unlimited access
INFO  rights) or the 'ncsoper' group (for minimal access rights).
INFO  To create the 'ncsadmin' group, use OS shell command:

  groupadd ncsadmin
```

TRAINOCATE

System Installation

- **Step 3**: Create **ncsadmin** (unlimited access rights) and **ncsoper** (minimal access rights) user groups. Assign user “osboxes” to ncsadmin group.

  sudo groupadd ncsadmin

  sudo groupadd ncsoper

  sudo usermod -a -G ncsadmin osboxes

```
osboxes@osboxes:~/module5/nso$ sudo groupadd ncsadmin
osboxes@osboxes:~/module5/nso$ sudo usermod -a -G ncsadmin osboxes
```

System Installation

- **Step 3**: Source the ncsrc file located in /opt/ncs/ncs-<version>/ncsrc. Then start NSO

```
osboxes@osboxes:~/module5/nso$ sudo -s
root@osboxes:/home/osboxes/module5/nso#
root@osboxes:/home/osboxes/module5/nso#
```

```
osboxes@osboxes:~/module5/nso$ ls -l /opt/ncs/
total 16
lrwxrwxrwx  1 root root    9 May 16 01:16 current -> ncs-5.7.1
drwxr-xr-x  2 root root 4096 May 16 01:16 downloads
-rw-r--r--  1 root root   80 May 16 01:16 installdirs
drwxr-xr-x 17 root root 4096 May 16 01:16 ncs-5.7.1
drwxr-xr-x  2 root root 4096 May 16 01:16 packages
```

- Start NSO

/etc/init.d/ncs start

```
$ source /opt/ncs/ncs-5.5.3.1/ncsrc

root@osboxes:/home/osboxes/module5/nso# /etc/init.d/ncs start
Starting ncs:
.
root@osboxes:/home/osboxes/module5/nso#
```

System Installation

- **Step 4**: Reboot the linux server. Then access the NSO CLI.

  $ sudo reboot

```
osboxes@osboxes:~$ ncs_cli -C

User osboxes last logged in 2022-05-16T05:29:05.844242+00:00, to osboxes, from 10.0.2.2
using cli-ssh
osboxes connected from 10.0.2.2 using ssh on osboxes
osboxes@ncs#
```

System Installation

- **Step 5 (optional)**: To access NSO WebUI, go to /etc/ncs and edit the ncs.conf file. Change the value for **webui enabled** to **true.** Then restart NSO.

cd /etc/ncs

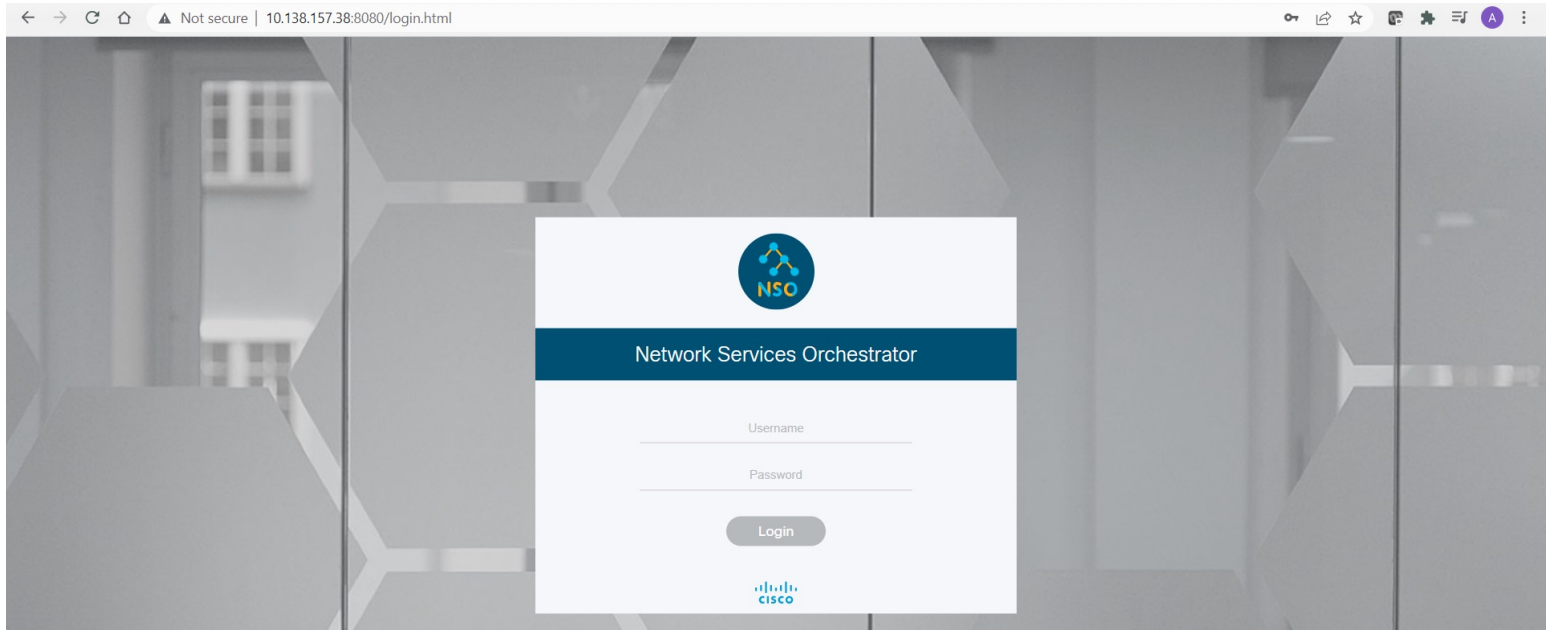
sudo nano ncs.conf

sudo /etc/init.d/ncs restart

```
<webui>
  <enabled>true</enabled>
  <transport>
    <tcp>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>8080</port>
    </tcp>
```

System Installation

- Access NSO WebUI using http://[server-ip-address]:8080/login.html



TRAINOCATE

System Installation

- NSO directories for system-install:
  - ✓ **Installation directory:** /opt/ncs/ncs-5.5.3.1 (linked to by /opt/ncs/current)
  - ✓ **Running directory:** /var/opt/ncs => contains packages (NEDs, service packages)
  - ✓ **Log directory:** /var/log/ncs
  - ✓ **Configuration directory:** /etc/ncs

Install NEDs

- Change permission to /var/opt/ncs folder

```
osboxes@osboxes:~$ sudo chmod 777 /var/opt/ncs/
osboxes@osboxes:~$ sudo chomod 777/var/opt/ncs/packages/
```

Go to folder containing NEDs file

```
osboxes@osboxes:~$ cd /opt/ncs/ncs-5.7.1/packages/neds
osboxes@osboxes:/opt/ncs/ncs-5.7.1/packages/neds$ ls -l
total 40
drwxr-xr-x 8 root root 4096 Jan 26 03:39 a10-acos-cli-3.0
drwxr-xr-x 7 root root 4096 Jan 26 03:39 alu-sr-cli-3.4
drwxr-xr-x 8 root root 4096 Jan 26 03:39 cisco-asa-cli-6.6
drwxr-xr-x 7 root root 4096 Jan 26 03:39 cisco-ios-cli-3.0
drwxr-xr-x 7 root root 4096 Jan 26 03:39 cisco-ios-cli-3.8
drwxr-xr-x 8 root root 4096 Jan 26 03:39 cisco-iosxr-cli-3.0
drwxr-xr-x 8 root root 4096 Jan 26 03:39 cisco-iosxr-cli-3.5
drwxr-xr-x 8 root root 4096 Jan 26 03:39 cisco-nx-cli-3.0
drwxr-xr-x 8 root root 4096 Jan 26 03:39 dell-ftos-cli-3.0
drwxr-xr-x 5 root root 4096 Jan 26 03:39 juniper-junos-nc-3.0
```

Install NEDs

Copy NED files to NCS running directory

```
osboxes@osboxes:/opt/ncs/ncs-5.7.1/packages/neds$ cp -r cisco-iosxr-cli-3.5 cisco-ios-cli-3.0 juniper-junos-nc-3.0
/var/opt/ncs/packages/

osboxes@osboxes:/opt/ncs/ncs-5.7.1/packages/neds$ cd /var/opt/ncs/packages/
osboxes@osboxes:/var/opt/ncs/packages$ ls -l
total 12
drwxr-xr-x 7 osboxes osboxes 4096 May 16 01:43 cisco-ios-cli-3.0
drwxr-xr-x 8 osboxes osboxes 4096 May 16 01:43 cisco-iosxr-cli-3.5
drwxr-xr-x 5 osboxes osboxes 4096 May 16 01:43 juniper-junos-nc-3.0
```

Install NEDs

- Go to NSO CLI and reload packages

  ncs_cli –C

  packages reload

```
osboxes@osboxes:/var/opt/ncs/packages$ ncs_cli -C

User osboxes last logged in 2022-05-16T05:31:44.284868+00:00, to osboxes, from 10.0.2.2 using cli-ssh
osboxes connected from 10.0.2.2 using ssh on osboxes
osboxes@ncs# packages reload

>>> System upgrade is starting.
>>> Sessions in configure mode must exit to operational mode.
>>> No configuration changes can be performed until upgrade has complete
...
>>> System upgrade has completed successfully.
reload-result {
   package cisco-ios-cli-3.0
   result true
}
reload-result {
   package cisco-iosxr-cli-3.5
   result true
}
reload-result {
   package juniper-junos-nc-3.0
   result true
}
```

Install NEDs

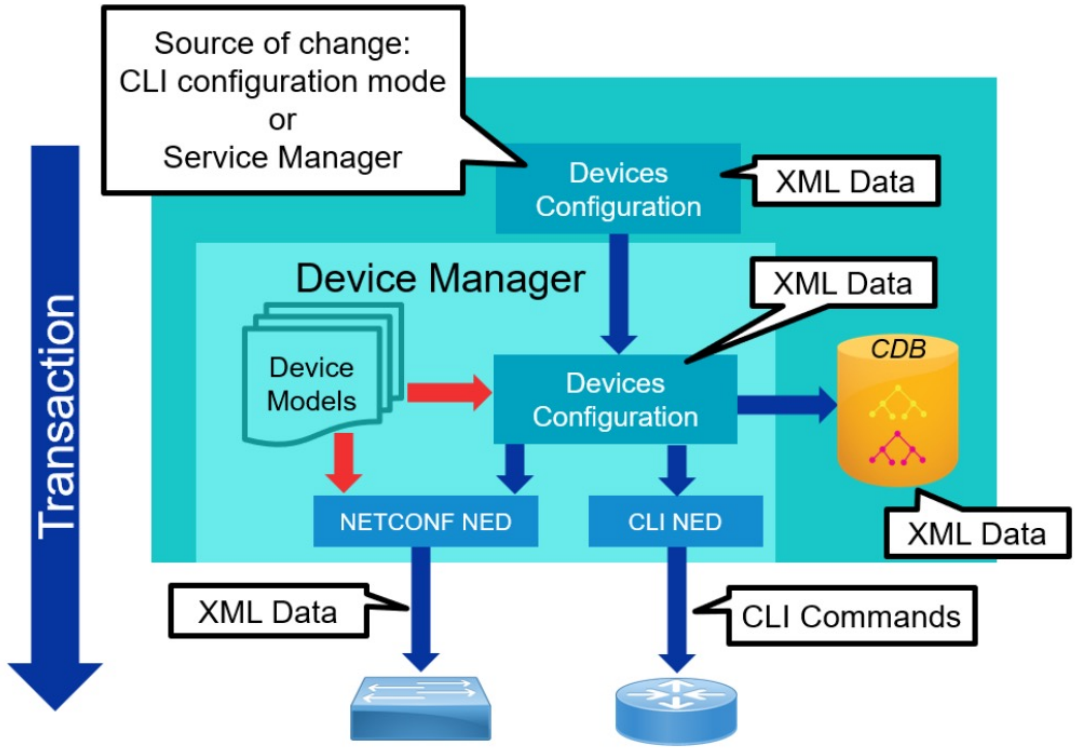
- Check status of NED packages

```
osboxes@ncs# show packages package oper-status
packages package cisco-ios-cli-3.0
 oper-status up
packages package cisco-iosxr-cli-3.5
 oper-status up
packages package juniper-junos-nc-3.0
 oper-status up
```

USING DEVICE MANAGER

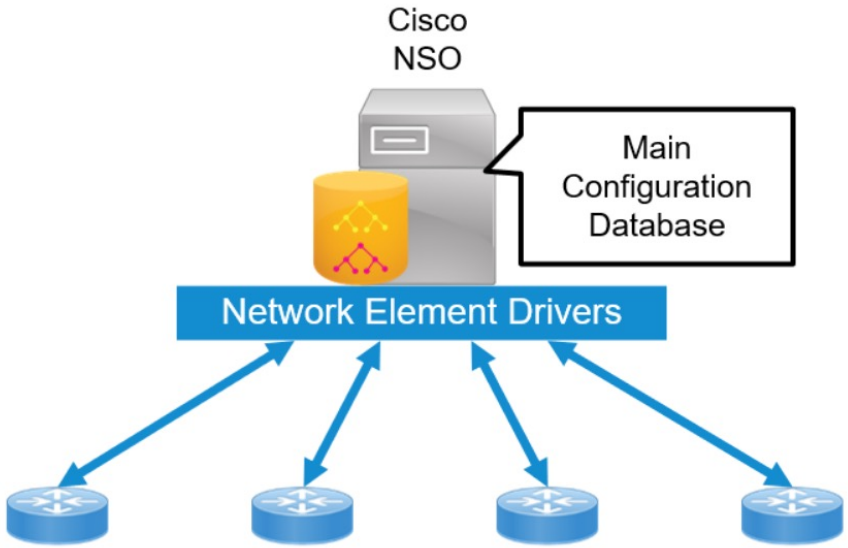
Device Manager Overview

- Responsible for communication with the managed devices using NEDs.



Device Manager Overview

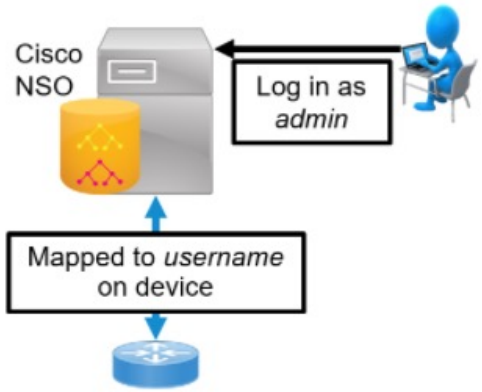
- Maintain a list of managed devices.
- Keep the main copy of each device's config in CDB.
- Validate configuration changes.
- Deploy configuration changes to devices.
- Rollback.
- Sync device configuration and NSO database.



Create device authgroup

- Authentication info for devices.
- Map a local NSO user to a remote username/password for the managed device.

```
admin@ncs# show running-config devices authgroups
devices authgroups group default
 umap admin
  remote-name     admin
  remote-password $4$wIo7Yd068FRwhYYI0d4IDw==
 !
 umap oper
  remote-name     oper
  remote-password $4$zp4zerM68FRwhYYI0d4IDw==
 !
!
```



Create device authgroup

- NSO looks up the local Cisco NSO username in the "umap" list of the authgroup.
  - If an entry is found, the credentials configured are used when authenticating to the managed device.
  - If no entry is found in the "umap" list, the credentials configured in "default-map" are used.

```
devices authgroups group iosxr-sandbox
default-map remote-name admin
default-map remote-password C1sco12345
```

Add a device

- There are several options for adding a device:
  - **Manually:** useful for a few devices (for example, development and testing).
  - **Templates:** replicate a device from a device template.
  - **Bulk upload:** useful for initial definition of many devices.

Add a device - Manually

```
devices device iosxr
address   sandbox-iosxr-1.cisco.com
port      22
authgroup iosxr-sandbox
device-type cli ned-id cisco-iosxr-cli-7.33
device-type cli protocol ssh
state admin-state unlocked


devices device iosxr-netconf
address   sandbox-iosxr-1.cisco.com
port      830
authgroup iosxr-sandbox
device-type netconf ned-id cisco-iosxr-nc-7.3
state admin-state unlocked
```

Add a device – Device Templates

Using device template

```
admin@ncs(config)# devices device www5 apply-template template-name std-web-server
admin@ncs(config)# devices device www5 address 127.0.0.1 port 23456 authgroup default
admin@ncs(config)# commit
```

TRAINOCATE

Add a device – Bulk Upload

- Use an XML file

  ncs_load -l -m <file_name.xml>

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <devices xmlns="http://tail-f.com/ns/ncs">
    <device>
      <name>SRT.asr901</name>
      <address>10.138.157.81</address>
      <port>23</port>
      <authgroup>cisco-authgroup</authgroup>
      <device-type>
        <cli>
          <ned-id xmlns:cisco-ios-cli-6.77="http://tail-f.com/ns/ned-id/cisco-ios-cli-6.77">
          cisco-ios-cli-6.77:cisco-ios-cli-6.77</ned-id>
          <protocol>telnet</protocol>
        </cli>
      </device-type>
      <state>
        <admin-state>unlocked</admin-state>
      </state>
    </device>
```

Check managed device list

```
# show devices list
NAME         ADDRESS                    DESCRIPTION  NED ID              ADMIN STATE
-----------------------------------------------------------------------------------------
iosxe        sandbox-iosxe-latest-1.cisco.com  -          cisco-ios-cli-6.74    unlocked
iosxr        sandbox-iosxr-1.cisco.com         -          cisco-iosxr-cli-7.33  unlocked
```

Sync configuration to/from devices

In normal operation, the configuration on the device and the configuration copy inside Cisco NSO should be identical.

```
admin@ncs# devices check-sync
sync-result {
        device PE11
        result in-sync
}
```

```
admin@ncs# devices sync-from

  sync-result {
     device lb0
     result true
  }
```

Sync-from: from device to NSO

```
admin@ncs# devices device PE11 sync-to

result true
```

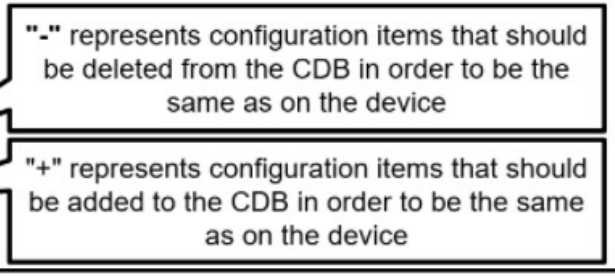
Sync-to: from NSO to device

Compare config

- Compare config stored in CDB and config on device

```
admin@ncs(config)# devices device PE11 check-sync
result out-of-sync
info got: 334bb33aae40155831edfa0b6a978f39 expected: a1424cd35da4499f6a71b3d38ae648a8

admin@ncs(config)# devices device PE11 compare-config
diff
 devices {
     device PE11 {
         config {
             ios:interface {
                 Loopback 10 {
                     ip {
                         address {
                             primary {
-                                address 10.1.1.1;
+                                address 2.2.2.2;
                             }
                         }
```



TRAINOCATE

Configure devices from NSO CLI

```
admin@ncs# config
admin@ncs(config)# devices device PE11 config ios:interface Loopback 20
admin@ncs(config-if)# ip address 10.2.2.2 255.255.255.255
admin@ncs(config)# devices device PE11 config ios:interface Loopback 30
admin@ncs(config-if)# ip address 10.3.3.3 255.255.255.255
admin@ncs(config-if)# show configuration
devices device PE11
 config
  ios:interface Loopback30
   ip address 10.3.3.3 255.255.255.255
   no shutdown
  exit
 !
!
admin@ncs(config)#
```

Displays current configuration items in the current configuration mode

TRAINOCATE

Configure devices from NSO CLI

Commit

```
admin@ncs# config
admin@ncs(config)# devices device PE11 config ios:interface Loopback 20
admin@ncs(config-if)# ip address 10.2.2.2 255.255.255.255
admin@ncs(config)# devices device PE11 config ios:interface Loopback 30
admin@ncs(config-if)# ip address 10.3.3.3 255.255.255.255
admin@ncs(config-if)# commit dry-run
cli {
    local-node {
        data  devices {
                  device PE11 {
                      config {
                          ios:interface {
+                             Loopback 20 {
+                                 ip {
+                                     address {
+                                         primary {
+                                             address 10.2.2.2;
+                                             mask 255.255.255.255;
  ...
}
admin@ncs(config-if)# commit
```

Displays configuration changes , which are accumulated in the candidate data store.

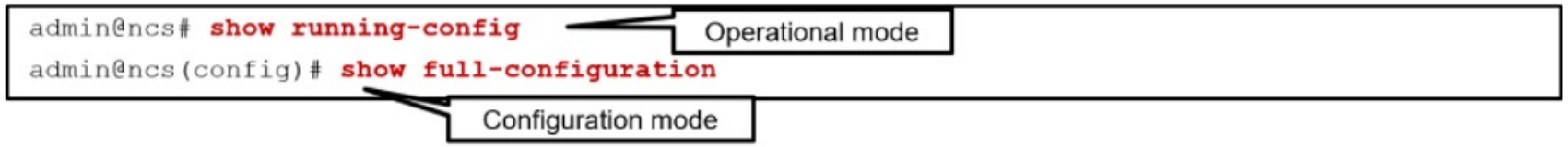
Commit will start the transaction.

Configure devices from NSO CLI

Commit options

```
admin@ncs(config)# commit ?
Possible completions:
  abort                - Abort a pending commit
  and-quit             - Commit current set of changes and exit configuration mode
  bypass-commit-queue  - Commit directly even if commit queue exists
  check                - Validate current configuration
  comment              - Add a commit comment
  confirmed            - Commit current set of changes with a timeout
  dry-run              - Show the diff but do not perform commit
  label                - Add a commit label
  no-confirm           - Commit current set of changes, do not query user
  no-networking        - Send nothing to the devices
  no-out-of-sync-check - Commit even if out of sync
  no-revision-drop     - Fail if device has too old data model
  persist-id           - Specify a persist-id
  through-commit-queue - Commit through the commit queue
```

Display configuration




```
cisco@ncs# show running-config devices device SRT.asr901
devices device SRT.asr901
 address   10.138.157.81
 port      23
 authgroup cisco-authgroup
 device-type cli ned-id cisco-ios-cli-6.77
 device-type cli protocol telnet
 state admin-state unlocked
 config
  hostname  hni05-lab-901-1
  tailfned police cirmode
  version   15.4
```


TRAINOCATE