



Two-stage Configuration Commit

This chapter describes how to enable two-stage configuration commit mode on the Cisco NX-OS device.

This chapter includes the following sections:

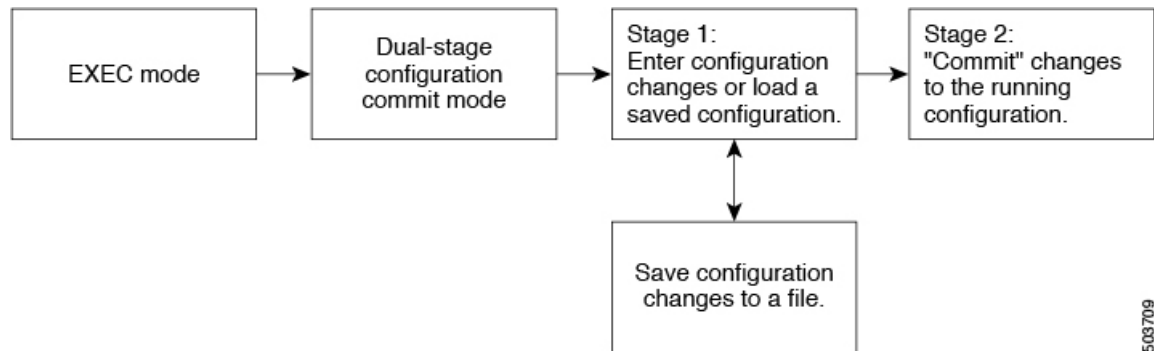
- [About Two-stage Configuration Commit, on page 1](#)
- [Guidelines and Limitations, on page 2](#)
- [Configuring in Two-Stage Configuration Commit Mode, on page 3](#)
- [Aborting the Two-Stage Configuration Commit Mode, on page 7](#)
- [Displaying Commit IDs, on page 7](#)
- [Rollback Capability, on page 8](#)
- [Viewing Current Session Configurations, on page 8](#)

About Two-stage Configuration Commit

In an interactive session, when you run a command, it's executed and it changes the running configuration. This behaviour is known as one-stage configuration commit. In the confirm-commit or the two-stage configuration commit, changes in configurations are stored in a staging database. These changes don't affect the running configuration until you run the **commit** command. This two-stage process creates a target configuration session, where you can make, edit, and verify configuration changes before committing them to the running state of the switch. You can also commit the changes for a time period you specify before you commit them permanently. After the specified time period, the switch reverts to the previous configuration if you don't run the **commit** command. When a commit is successful, you can view the commit information that includes the commit ID, username, and timestamp.

The following figure shows the two-stage configuration commit process.

Figure 1: Two-Stage Configuration Commit Process



503709

Guidelines and Limitations

Two-stage configuration commit has the following configuration guidelines and limitations:

- This feature is supported only for a CLI interface in a user-interactive session.
- Before you run any feature-related configuration commands, enable the feature using the **feature** command and commit it using the **commit** command.
- Two-stage configuration commit mode doesn't support other modes like maintenance mode, scheduler mode, or virtual mode.
- When you're in the two-stage configuration commit mode, avoid editing configurations in one-stage configuration commit mode from different sessions at the same time.
- Review the configurations using the **show configuration** command before committing the changes.
- Show configuration displays the staged configs:
 - It displays the real difference, that is yes and no form of the same command will result in empty config.
 - It is recommended to issue the exact no form of the cli to negate the config.

Example: to negate 'ip address x' config, user has to give 'no ip address x' instead of 'no ip address'.
 - Interface layer change commands (switchport/no switchport) should be issued explicitly.
 - Any invalid config in the session should manually be removed by the user before attempting commit. If could not remove manually clear the session and start a new session.
- If the verification fails, edit and retry the commit.
- If the commit fails, the configuration rolls back to the previous configuration.
- Configurations that you don't commit aren't saved after you reload the switch.
- This feature doesn't support commits with NX-API, EEM, PPM and Netconf.
- You can have only one active two-stage configuration commit session at a given time.

Configuring in Two-Stage Configuration Commit Mode

To enable a feature in the two-stage configuration commit mode, perform the following steps:



Note In this procedure, the BGP feature is enabled as an example.

Procedure

	Command or Action	Purpose
Step 1	configure dual-stage Example: <pre>switch# configure dual-stage switch(config-dual-stage)#</pre>	Creates a new target configuration session. Note The target configuration isn't a copy of the running configuration. It has only the configuration commands entered during the target configuration session.
Step 2	feature <i>feature_name</i> Example: <pre>switch(config-dual-stage)# feature bgp switch(config-dual-stage)#</pre>	Enables the feature. Note <ul style="list-style-type: none"> You can enable the feature even before entering the two-stage configuration commit mode. You can't combine feature-related commands in a commit if the feature isn't already enabled.
Step 3	commit [confirmed <i>seconds</i>] Example: <pre>switch(config-dual-stage-router)# commit confirmed 30 Verification Succeeded. Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time. Configuration committed by user 'admin' using Commit ID : 100000001 switch(config-dual-stage)# switch(config-dual-stage)# commit Confirming commit for trial session. switch(config-dual-stage)#</pre> Example:	Commits changes to the running configuration. <ul style="list-style-type: none"> confirmed: Commits the changes to the running configuration. seconds: Commits the configuration in global configuration mode on a trial basis for a minimum of 30 seconds and a maximum of 65535 seconds. Note If you enter a trial period, run the commit command to confirm the configuration. If you don't run the commit command, the switch reverts to the previous configuration after the trial period.

	Command or Action	Purpose
	<pre>switch(config-dual-stage)# hostname example-switch switch(config-dual-stage)# commit Verification Succeeded.</pre> <p>Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time.</p> <p>Configuration committed by user 'admin' using Commit ID : 1000000002</p> <pre>example-switch(config-dual-stage)#</pre>	
Step 4	<p>Example:</p> <pre>switch(config-dual-stage)# router bgp 64515.46 switch(config-dual-stage-router)# switch(config-dual-stage-router)# router-id 141.8.139.131 switch(config-dual-stage-router)#</pre>	Run any feature-related commands that are supported in this configuration mode.
Step 5	<p>show configuration</p> <p>Example:</p> <pre>switch(config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131</pre>	<p>Displays the target configuration.</p> <p>Note You can run this command only in the dual-stage configuration mode.</p>
Step 6	<p>commit [confirmed <i>seconds</i>]</p> <p>Example:</p> <pre>switch(config-dual-stage-router)# commit</pre> <p>Verification Succeeded.</p> <p>Proceeding to apply configuration. This might take a while depending on amount of configuration in buffer. Please avoid other configuration changes during this time.</p> <p>Configuration committed by user 'admin' using Commit ID : 1000000003</p>	Commits changes to the running configuration.
Step 7	<p>(Optional) show configuration commit [changes] <i>commit-id</i></p> <p>Example:</p> <pre>switch(config-dual-stage-router)# show configuration commit changes 1000000003 *** /bootflash/.dual-stage/1000000003.tmp Fri Mar 19 10:59:00 2021 --- /bootflash/.dual-stage/1000000003 Fri Mar 19 10:59:05 2021 ***** *** 378,383 **** --- 378,385 ----</pre>	<p>Displays commit-related information.</p> <p>Only the last 50 commits or the commit files stored in the reserved disk space are saved. The reserved disk space is 20 MB. All the commit sessions will be removed when you reload the switch. However, the commit IDs are not removed. Also, these commit IDs will not be removed in case of write, erase, and reload.</p> <p>Use the show configuration commit changes <i>commit-id</i> command to view only the changes</p>

	Command or Action	Purpose
	<pre> line console line vty boot nxos bootflash:/nxos64.10.1.1.44.bin + router bgp 64515.46 + router-id 141.8.139.131 xml server timeout 1200 no priority-flow-control override-interface mode off Example: switch(config-dual-stage)# show configuration commit 1000000003 feature bgp router bgp 64515.46 router-id 141.8.139.131 . . . </pre>	<p>in the current session of the commit you specify.</p> <p>Use the show configuration commit <i>commit-id</i> command to view the complete configurations in the commit you specify.</p>
Step 8	<p>(Optional) save configuration <i>filename</i></p> <p>Example:</p> <pre> switch(config-dual-stage)# save configuration bootflash:test.cfg </pre>	<p>Saves the target configurations to a separate file without committing them to the running configuration.</p> <p>Note</p> <ul style="list-style-type: none"> • You can load the target configuration files later, modify, or commit. The file will be saved in bootflash. • You can view the configuration file you saved by running the show configuration file <i>filename</i> command. • Some of the user-specific information will be masked based on the user role. • Configs saved in dual stage mode is an encrypted file and can be viewed only using #show configuration file <> and not using #show file <>.
Step 9	<p>(Optional) load <i>filename</i></p> <p>Example:</p> <pre> switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)# load </pre>	<p>Loads a target configuration that you saved. After loading a file, you can modify it or commit it to the running configuration. To save the changes, use the save configuration <i>filename</i> command.</p>

	Command or Action	Purpose
	<pre>test.cfg switch (config-dual-stage-router)# show configuration ! Cached configuration ! router bgp 1 switch (config-dual-stage-router)#</pre>	You can load a target configuration that you saved using only the save configuration filename command.
Step 10	<p>(Optional) clear configuration</p> <p>Example:</p> <pre>switch (config-dual-stage)# show configuration ! Cached configuration ! router bgp 64515.46 router-id 141.8.139.131 switch (config-dual-stage)# clear configuration switch (config-dual-stage)# show configuration ! Cached configuration switch (config-dual-stage)#</pre>	Clears changes made to the target configuration without terminating the configuration session. It deletes any configuration changes that aren't committed.
Step 11	<p>end</p> <p>Example:</p> <pre>switch (config-dual-stage-if)# end Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]</pre>	<p>Exits the global dual stage configuration mode.</p> <p>If you end a configuration session without committing the configuration changes, you'll be prompted to save changes, discard changes, or cancel the action:</p> <ul style="list-style-type: none"> • Yes: Commits the configuration changes and exit configuration mode • No: Exits the configuration mode without committing the configuration changes • Cancel: Remains in configuration mode without committing the configuration changes <p>Note</p> <ul style="list-style-type: none"> • If you choose to exit when a confirm commit timer is running, the same options are displayed. If you still chose to exit, the trial configuration rolls back instantly. • If the default session times out before the timer expires, the trial configuration rolls back before exiting the session. In this case, no warning message appears.

Aborting the Two-Stage Configuration Commit Mode

When you abort a configuration session, uncommitted changes are discarded and the configuration session ends. No warning appears before the configuration changes are deleted.

```
switch(config-dual-stage)# router bgp 1
switch(config-dual-stage-router)# neighbor 1.2.3.4
switch(config-dual-stage-router-neighbor)# remote-as 1
switch(config-dual-stage-router-neighbor)# show configuration
! Cached configuration
!
router bgp 1
neighbor 1.2.3.4
remote-as 1
switch(config-dual-stage-router-neighbor)# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:17:40 2021
!Time: Wed Mar 17 16:17:55 2021

version 10.1(2) Bios:version
feature bgp

switch(config-dual-stage-router-neighbor)# abort
switch# show run bgp

!Command: show running-config bgp
!Running configuration last done at: Wed Mar 17 16:18:00 2021
!Time: Wed Mar 17 16:18:04 2021

version 10.1(2) Bios:version
feature bgp

switch#
```

Displaying Commit IDs

At each successful commit, the commit ID is displayed in the syslog. The total number of commit IDs saved in the system depends on the configuration size and the disk space available. However, the maximum number of commit IDs stored at any given time is 50.

Use the **show configuration commit list** command to view information about the last 50 commit IDs. Each entry shows the user who committed configuration changes, the connection used to execute the commit, and commit ID timestamp.

```
switch# show configuration commit list
SNO. Label/ID      User      Line      Client      Time Stamp
~~~~~
1    1000000001    admin    /dev/ttyS0  CLI         Wed Jul 15 15:21:37 2020
2    1000000002    admin    /dev/ttyS0  Rollback    Wed Jul 15 15:22:15 2020
3    1000000003    admin    /dev/pts/0  CLI         Wed Jul 15 15:23:08 2020
4    1000000004    admin    /dev/pts/0  Rollback    Wed Jul 15 15:23:46 2020
```

Rollback Capability

You can rollback the configuration to any of the previous successful commits. Use the **rollback configuration** command to rollback to any of the last 50 commits.

```
switch# rollback configuration to ?
1000000015
1000000016
1000000017

:
:

switch#
```

Each commit ID acts as a (checkpoint or) rollback point. You can rollback to any given commit ID. When you roll back the configuration to a specific rollback point, you undo all configuration changes made during the session identified by the commitID for that rollback point, and you undo all configuration changes made after that point. The rollback process rolls back the configuration and commits the rolled-back configuration. The rollback process also creates a new rollback point (commit ID) so that you can roll back the configuration to the previous configuration.

```
switch(config-dual-stage)# rollback configuration to 1000000002
Rolling back to commitID :1000000002
ADVISORY: Rollback operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Rollback failure.

Configuration committed by rollback using Commit ID : 1000000004
switch(config-dual-stage)#
```

Viewing Current Session Configurations

You can view the current session configuration using the **show configuration** command. This command is supported only in the dual-stage mode. The session configuration is cleared if a commit fails.

```
switch(config-dual-stage-cmap)# show configuration
! Cached configuration
!
class-map type control-plane match-any copp-s-ipmcmis
class-map type control-plane match-any copp-s-l2switched
class-map type control-plane match-any copp-s-l3destmiss
switch(config-dual-stage-cmap)#
```

If there is no configuration, the following message appears:

```
switch(config-dual-stage)# show configuration
! Cached configuration
switch(config-dual-stage)# commit
No configuration changes to commit.
switch(config-dual-stage)#
```