

Troubleshoot NSO NED External Errors

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components used](#)

[Types of External Errors](#)

[Reading NED Trace Logs](#)

[Enabling NED Trace Logs](#)

[CLI and Generic NEDs](#)

[Troubleshoot External Errors](#)

[Common Specific Issues:](#)

[Translation Issue](#)

[Ordering Issue](#)

[Invalid Value Issue](#)

[Errors, Warnings and Informational messages from the device:](#)

[Resolutions](#)

[User Error](#)

[Network Device](#)

[Ned-settings](#)

[NED-Service Conflict](#)

[NED Enhancements and TAC Support](#)

Introduction

This document describes how to analyze CLI and Generic type NED traces and identify the cause of external errors in Cisco Crosswork NSO.

Prerequisites

Requirements

- The troubleshooting aspect of this document applies to all currently supported Network Element Driver (NED) versions and Cisco® Crosswork Network Service Orchestrator (NSO) versions.
- In order to troubleshoot NED external errors, you need the ability to enable NED trace logs as well as have access to these logs.
- In order to accurately assess NED external errors, knowledge of the network device managed by NSO and its API is recommended.

Components used

- Crosswork NSO 6.4.3

Types of External Errors

NED external errors are a sign of a communication breakdown between the NED and the device. They occur in three broad categories:

- The NED receives a response from the network device it did not expect.
- The NED receives a response from the network device which violates the yang model in the NED.
- The NED does not receive a response within the allotted time.

The category of unexpected responses is by far the most common category is external errors your NED can encounter. It includes the device returning an error message, an informational message or any other type of information which does not match what NSO was expecting to see returned. NEDs are designed to handle informational messages or warnings which can safely be ignored. Many NEDs have ned-settings available to customize which messages to ignore or which messages to treat as an external error.

You can see an external error raised by the NED when the NED receives information which does not match the yang model during a sync-from or compare-config operation. A typical example is a yang model which accepts a value of 0 to 8 for a given leaf, yet in the most recent version of this OS, the range has been increased to 0 to 16. The NED does not accept a value of 16 as it is outside of the modeled range. Alternatively the error can be raised when a leaf is marked mandatory in the yang model but not provided by the device or when the device provides a string when NSO expects an integer.

For CLI and Generic NEDs, no external error is raised if the NED receives configuration which is not modeled in the NED yang model. Instead, this is logged as a skipped line in the trace file.

Finally, when a NED does not receive the expected response from the device within the allotted time, an external error is raised. This can happen because the device is non responsive and did not send a response, but it can also happen when the response from the device was not recognized by the NED.

Reading NED Trace Logs

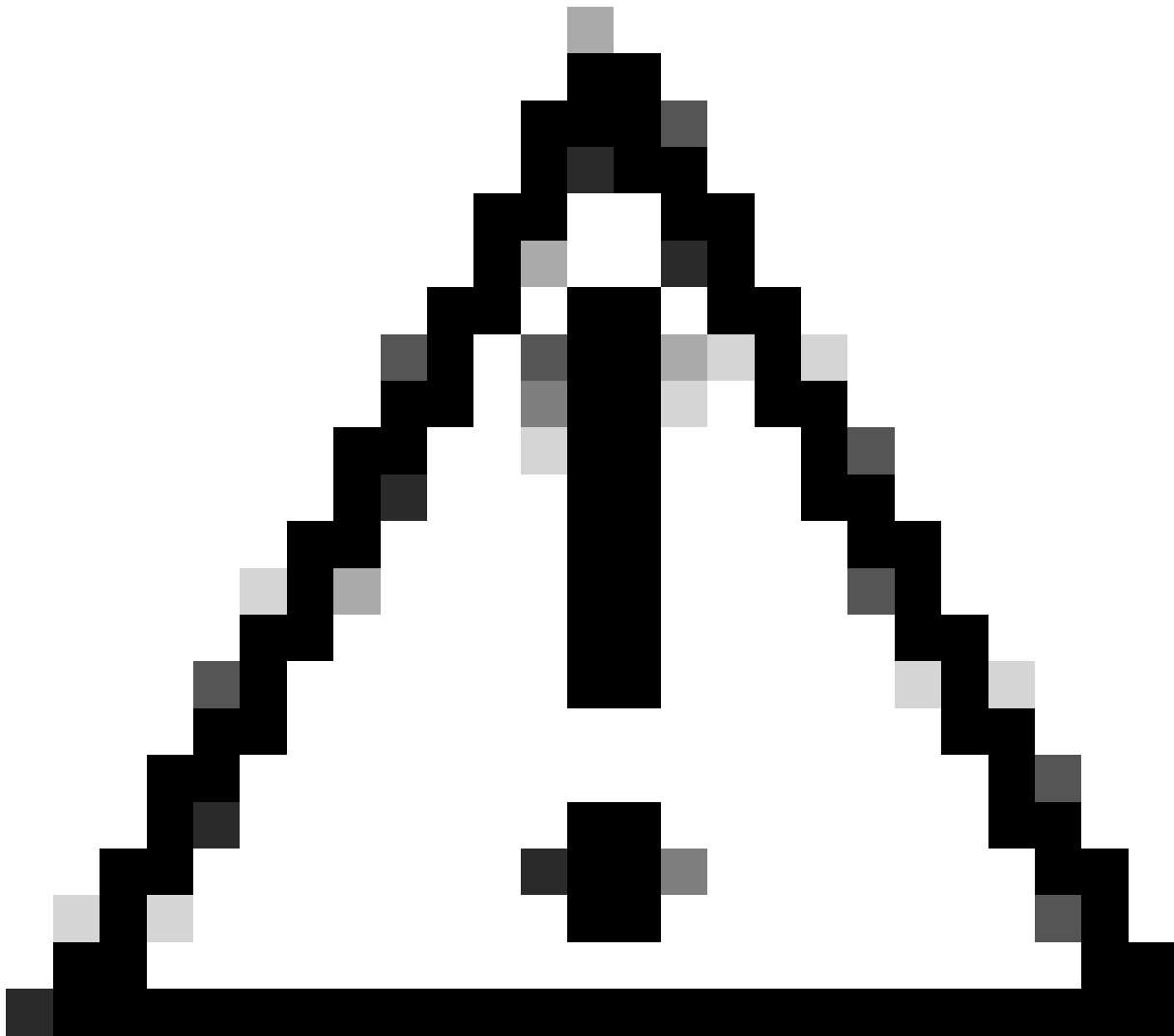
Trace logs are the best available logs to troubleshoot external errors.

Enabling NED Trace Logs

NED trace logs are enabled from the NSO CLI.

```
ncs_cli -C -u admin
admin@ncs# configure
admin@ncs(config)# devices device dev-1 ned-settings [ned-id] logging level debug
admin@ncs(config)# devices device dev-1 trace raw
admin@ncs(config)# commit
admin@ncs(config)# devices device dev-1 disconnect
admin@ncs(config)# devices clear-trace
admin@ncs(config)# devices device dev-1 compare-config
```

For [ned-id] , use the ned-id for the device you are targeting with the command.



Caution: The clear-trace command clears the data for all NED trace logs currently in the log directory. If you have trace logs you want to keep for this device or any other devices you must archive those before running this command. In current NSO versions you can execute clear-trace for a single device.



Note: In case you do not find "ned-settings [ned-id] logging level debug", you can skip this command.

These commands clear any old data from the trace file and prepare it with the current configuration on the device. At this point you reproduce the found issue using `ncs_cli` or your NSO service. If you encountered the error during a commit operation you must capture CLI outputs for `commit dry-run` and `commit dry-run outformat native` for future reference.

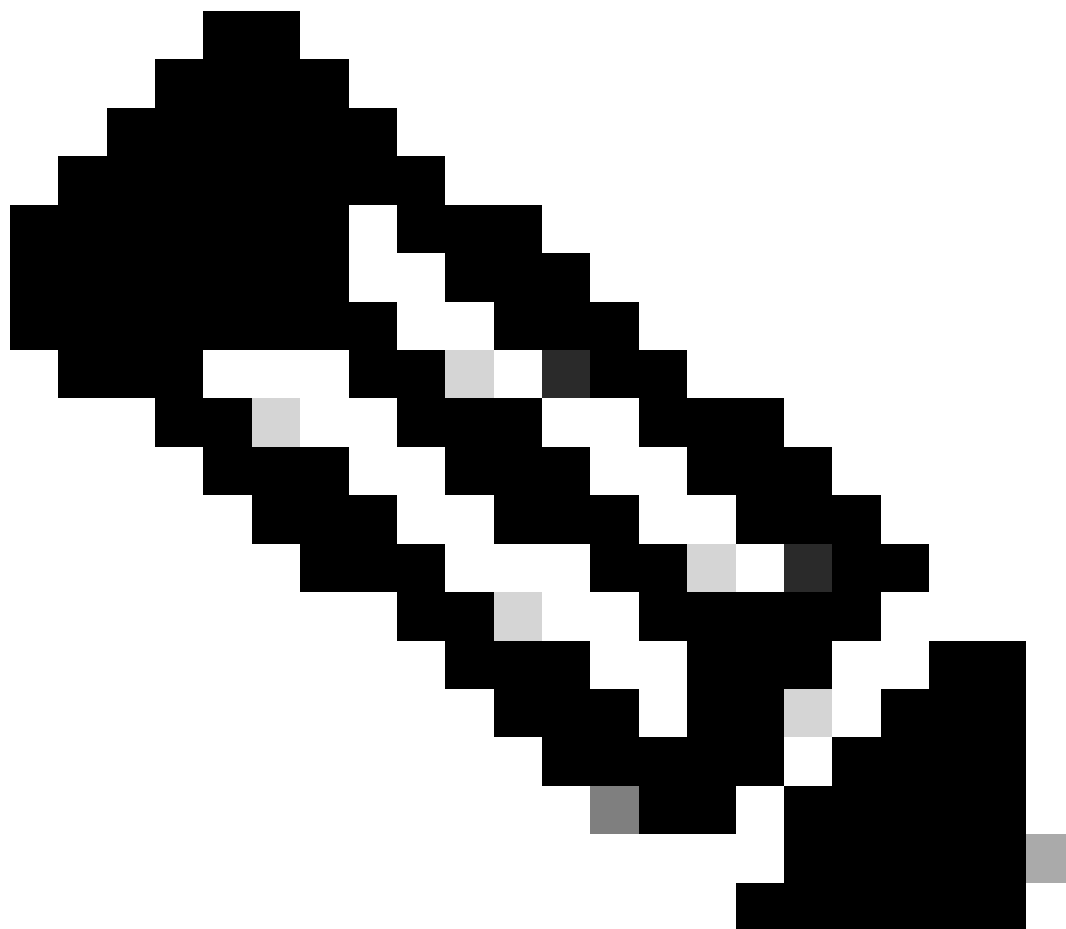
In the NED README file you can find a chapter labeled "How to report NED issues and feature requests" for more detailed instructions.

CLI and Generic NEDs

NED traces for CLI and Generic NEDs are sectioned in distinct phases which are useful for troubleshooting. The most important phases to understand for the purposes of troubleshooting external errors are the **SHOW** and **PREPARE** phases.

The **SHOW** phase is invoked when NSO is reading information from the network device. It is part of `sync-from` and `compare-config` operations. During this step, NSO prompts the device with a command such as `show running-config`

before reading and parsing the response from the device. Outgoing messages, sent from NSO to the device, are headed with `*** output` while incoming messages, sent by the device to NSO, start with `*** input`.



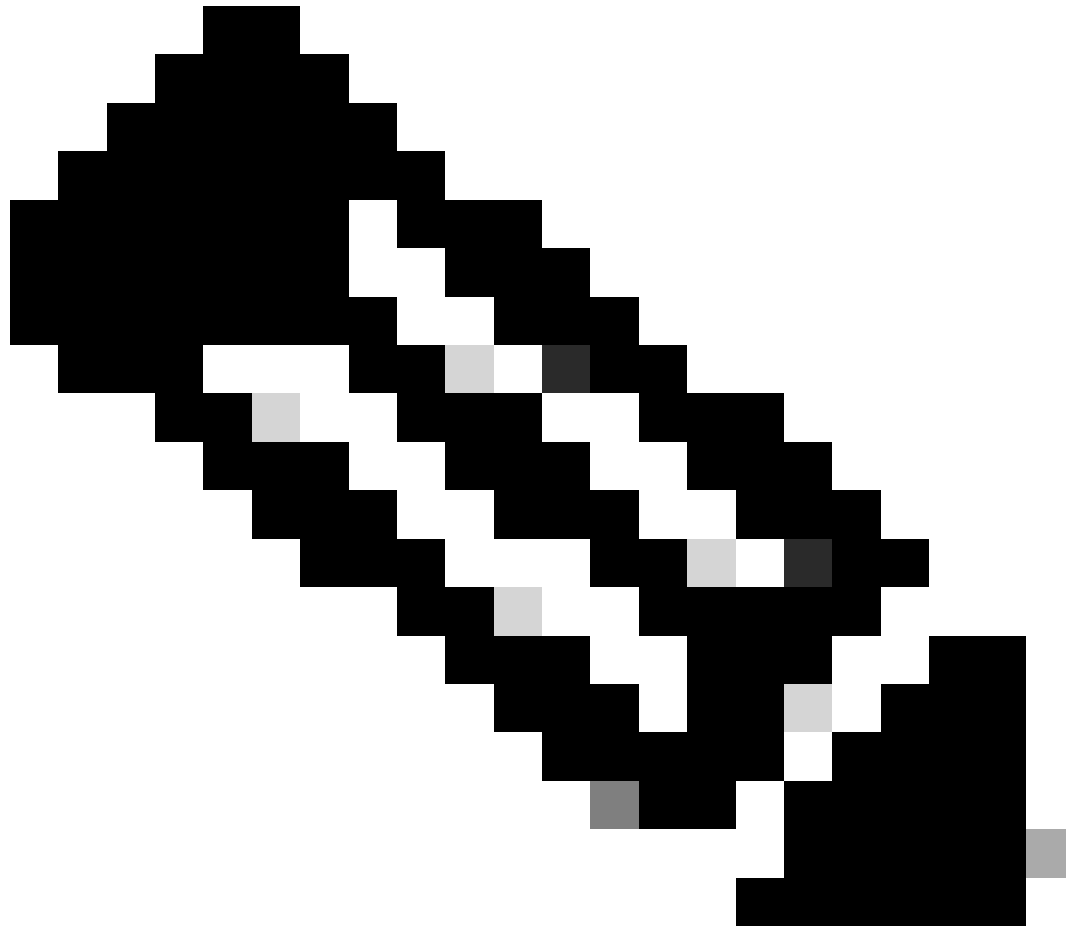
Note: External errors during a `SHOW` operation include values which are not accepted under the current yang model, and timeout issues.

The `PREPARE` phase is invoked as part of `commit` operations. During this phase NSO sends instructions to the device. At the start of a `PREPARE` phase the NED prints a list of the changes NSO intends to make to the trace file. After this initial summary, NSO sends the instructions to the device. For certain devices, NSO sends the commands in bulk, while for other devices these commands are sent one by one. This behavior can be altered using the relevant `ned-settings` for NEDs which support it. For example, the `cisco-iosxr-cli` NED has NED setting `"write number-of-lines-to-send-in-chunk <1-1000> (default 100)"`

For CLI NEDs, it is common to see the commands sent by NSO as output returned as input. This is because the command appears in the text-based UI of the device and NSO regards all text which appears in this UI as input. An example where NSO sends commands one by one can look like:

```
*** output 1-Jan-2024::09:56:00.928 user: admin/425 thandle 7428 hostname NS01 device test-device ***
interface GigabitEthernet 0/0/0/2.34280485 12transport
```

```
*** input 1-Jan-2024::09:56:00.929 user: admin/425 thandle 7428 hostname NS01 device test-device ***
interface GigabitEthernet 0/0/0/2.34280485 12transport
```



Note: External errors during a PREPARE operation include any messages returned by the device which do not fit the expectations of NSO such as errors, warnings or informational messages.

Troubleshoot External Errors

When troubleshooting external errors for CLI and Generic NEDs: enable the trace, reproduce the issue and examine the latest SHOW or PREPARE phase, depending on which operations triggered the error.

For an issue where NSO complains about a specific value provided by the device:

- What part of the configuration did NSO complain about?
- Is the configuration shared by the device correct for this model and version of the device?
- Are you aware of any other device models or versions using this NED which have more restrictive input requirements?

For an issue where NSO raises an external error involving a timeout:

- What is the last command sent by NSO prior to the timeout?
- What is NSO waiting for before proceeding?

It can be difficult to determine what NSO is waiting for. Some NEDs at increased verbosity print the regex expression they are looking for. In some cases, the message NSO was looking for does appear in the trace file but NSO did not recognize it and continues to wait.

For an issue where NSO raises an external error because of an unexpected response:

- What is the specific response the device provided?
- What command sent by NSO triggered the response?
- Is the response expected based on what NSO sent?
- Could the response from the device be safely ignored or must it abort the commit?
- Did NSO send the correct commands in the correct order?

Common Specific Issues:

Translation Issue

A translation issue happens when NSO has the correct intention but the commands it sends to the device are not quite correct. This can happen when a different device version or model using the same NED has slightly different syntax. If you are using an older version of the NED, please verify whether the same behavior still exists in the latest release of the NED. Additionally, please check if there are any ned-settings available in the README-ned-settings.md file included in the NED to see if any settings allow you to customize this behavior. If the latest NED still has the issue and the ned-settings have no method to resolve it, please open a case with TAC. Provide:

- NSO version
- Using the latest NED, A NED trace capturing a `compare-config` operation followed by a `commit` operation sending the incorrect command.
- The command the NED is currently sending
- The correct command, preferably shown as an example on the real device CLI
- The version and model of the device which has the issue
- Any other version and model of devices in your network using this NED where the current NED behavior is correct.

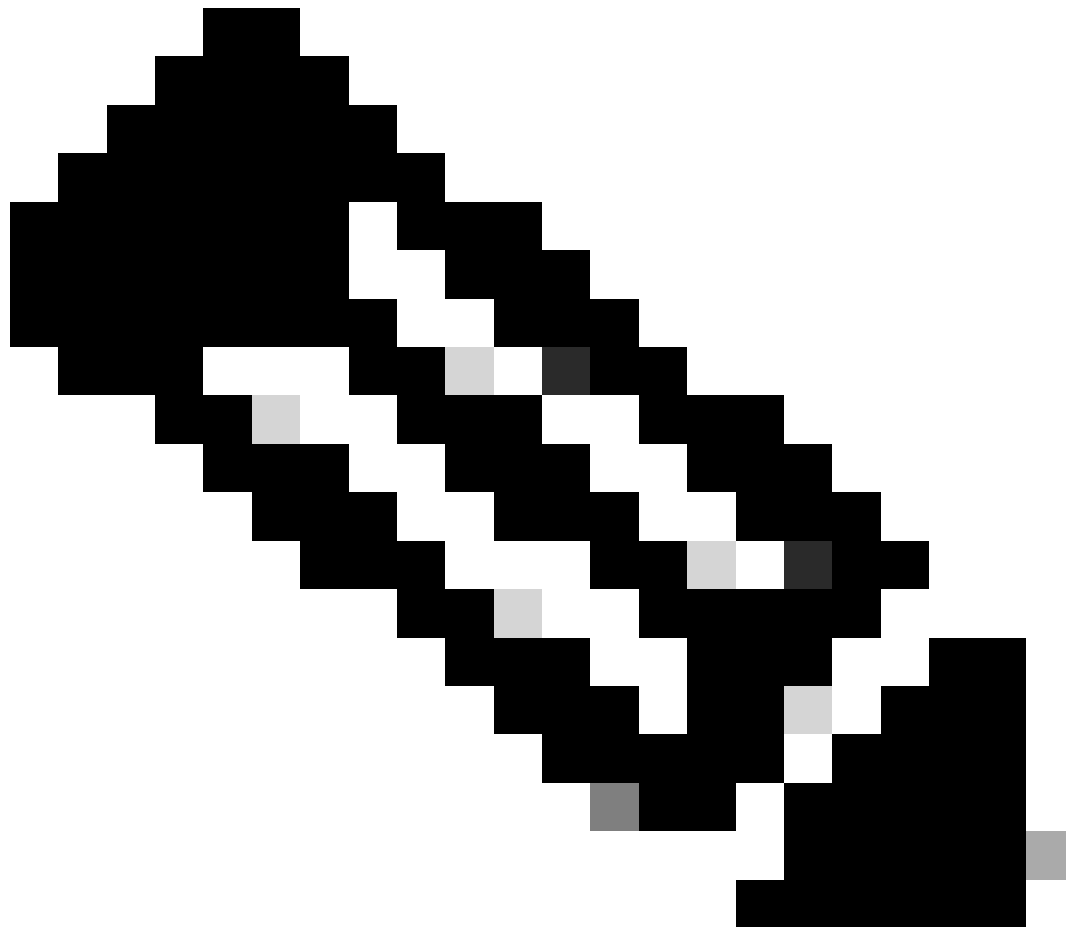
Ordering Issue

An ordering issue happens when the NED is sending the correct commands in the wrong order. For some devices and specific configuration payloads, the order is important. If you are using an older version of the NED, please verify whether the same behavior still exists in the latest release of the NED. Additionally, please check if there are any ned-settings available in the README-ned-settings.md file included in the NED to see if any settings allow you to customize this behavior. If the latest NED still has the issue and the ned-settings have no method to resolve it, please open a case with TAC. Provide:

- NSO version
- Using the latest NED and without using a service: A NED trace capturing a `compare-config` operation followed by a `commit` operation sending the incorrect order.
- The output of `commit dry-run outformat native` for the faulty commit. This shows you the order in which the

NED is currently sending the commands.

- The correct order, preferably shown as an example on the real device CLI
 - Any other prerequisites which are relevant for this ordering requirement
-



Note: In rare cases, Cisco is unable to address an ordering requirement through the NED in which case you can implement a multi-commit workflow or raise a bug report with the relevant vendor.

Invalid Value Issue

An invalid value issue happens when NSO is allowing a different range of values to be set than what the device accepts or for NSO to not allow the full range the device allows. For example, NSO allows you to define a value between 0-15, but the device only accepts values 0-8. This can happen when the NED is modeled with a specific device model and version in mind but other devices carry different expectations. If you are using an older version of the NED, please verify whether the same behavior still exists in the latest release of the NED. Additionally, please check if there are any ned-settings available in the README-ned-settings.md file included in the NED to see if any settings allow you to customize this behavior. If the latest NED still has the issue and the ned-settings have no method to resolve it, please open a case with TAC. Provide:

- NSO version
- For NSO allowing data which is not accepted by the device, using the latest NED: A NED trace capturing a `compare-config` operation followed by a `commit` sending a value which is rejected by the device.
- For NSO having a stricter interpretation of the data than is allowed by the device, using the latest NED: A NED trace capturing a `sync-from` operation after configuring data on the device which is currently not accepted by NSO.
- The correct value range
- The version and model of the device which has the issue
- Any other version and model of devices in your network using this NED where the current NED behavior is correct.

Errors, Warnings and Informational messages from the device:

When a device responds to NSO commands with an Error or other message, this can trigger an External Error in NSO. NSO NEDs have an internal list of regex expressions it can safely ignore, as well as expressions which trigger an Error. Several NEDs have `ned-settings` which allow you to customize these lists without the need for a NED Enhancement. For example: The `cisco-iosxr-cli` NED `ned-setting write config-warning`.

If the latest NED does not have such an option, please open a case with TAC. Provide:

- NSO version
- A NED trace capturing a `compare-config` operation followed by the operation resulting in the error
- A request to alter the internal lists and/or a requests to add support for a `ned-setting` option to customize the list yourself
- The specific message the device is sending causing the problem

Resolutions

User Error

If you determined that the commands sent by NSO were incorrect, make sure your input to NSO and your service packages generated the correct changes. Verify whether the output of `commit dry-run` matches the changes you wanted to make and whether the output of `commit dry-run outformat native` shows the correct commands and ordering to evoke those changes. If the dry-run predicts unexpected changes, you must verify your inputs to NSO or your service-code. If the dry-run is correct but the commands being sent to NSO are incorrect, check the translation and ordering issue resolutions.

Network Device

In some cases, an external error is the result of configuration, settings or even a bug on the network device itself such as a user not having the correct authorization or a device restricting certain operations. Evaluate whether the configuration or device settings can be altered to allow for NSO to better work with the device.

Ned-settings

Each NED has a range of `ned-settings` to help you customize the way NSO interacts with the device. Ned-settings are documented in the `README-ned-settings.md` file inside the NED and tend to differ from NED to NED. The `cisco-iosxr-cli` NED has options to change the way NSO calculates a checksum for the device, how many commands are sent in bulk, customize additional commands to inject based on specific triggers or whether the NED must collect configuration data using `"show running-config"` or by writing the config to a file

on the device and transferring the file using sftp which can be useful for large configurations.

NED-Service Conflict

A NED-Service conflict happens when a problematic behavior is present when configuration is changed or deleted using a service-package but does not appear when the same configuration changes are made without the use of a service-package. This type of behavior can appear as unexpected configuration being added or removed, resulting in External Errors from the device. This is typically the result of service-ownership over parts of the configuration. Changes to the NSO CDB configuration resulting from a service-package can override NED logic which would normally guard against incorrect changes. If you suspect you encountered this behavior, please verify by attempting the same configuration changes without the use of a service-package.

Please refer to the [Article Understand NSO Service Ownership](#) to learn more about Service Ownership and the possible solutions.

NED Enhancements and TAC Support

If no other options are available you can open a ticket with [Cisco TAC](#) and request to have the NED updated to fit your needs.

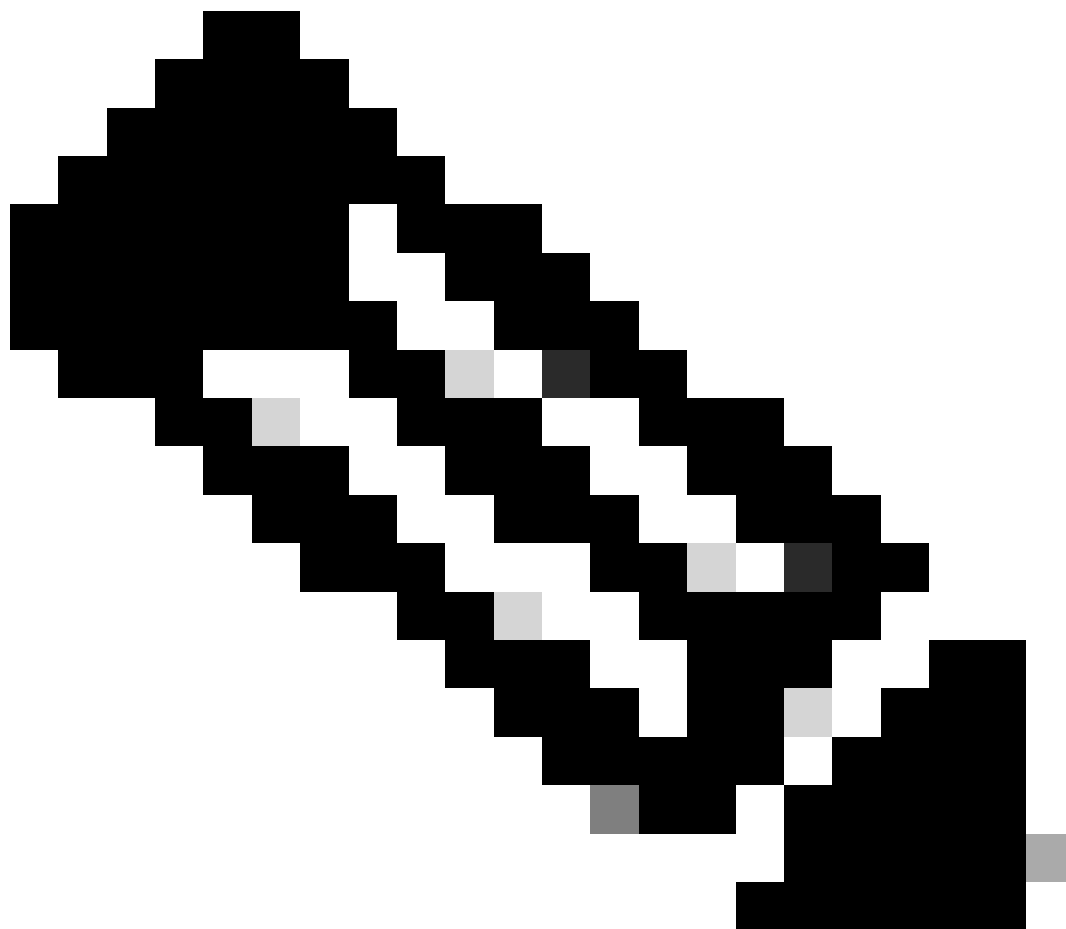
NSO NEDs provided by Cisco are built on updated based on your use cases. Cisco does not proactively attempt to cover all possible device models and versions but the NED is constantly updated to meet the needs of an evolving network and new usecases. You can find a summary of the scope of NED support provided [by the Crosswork NSO Developers here](#)



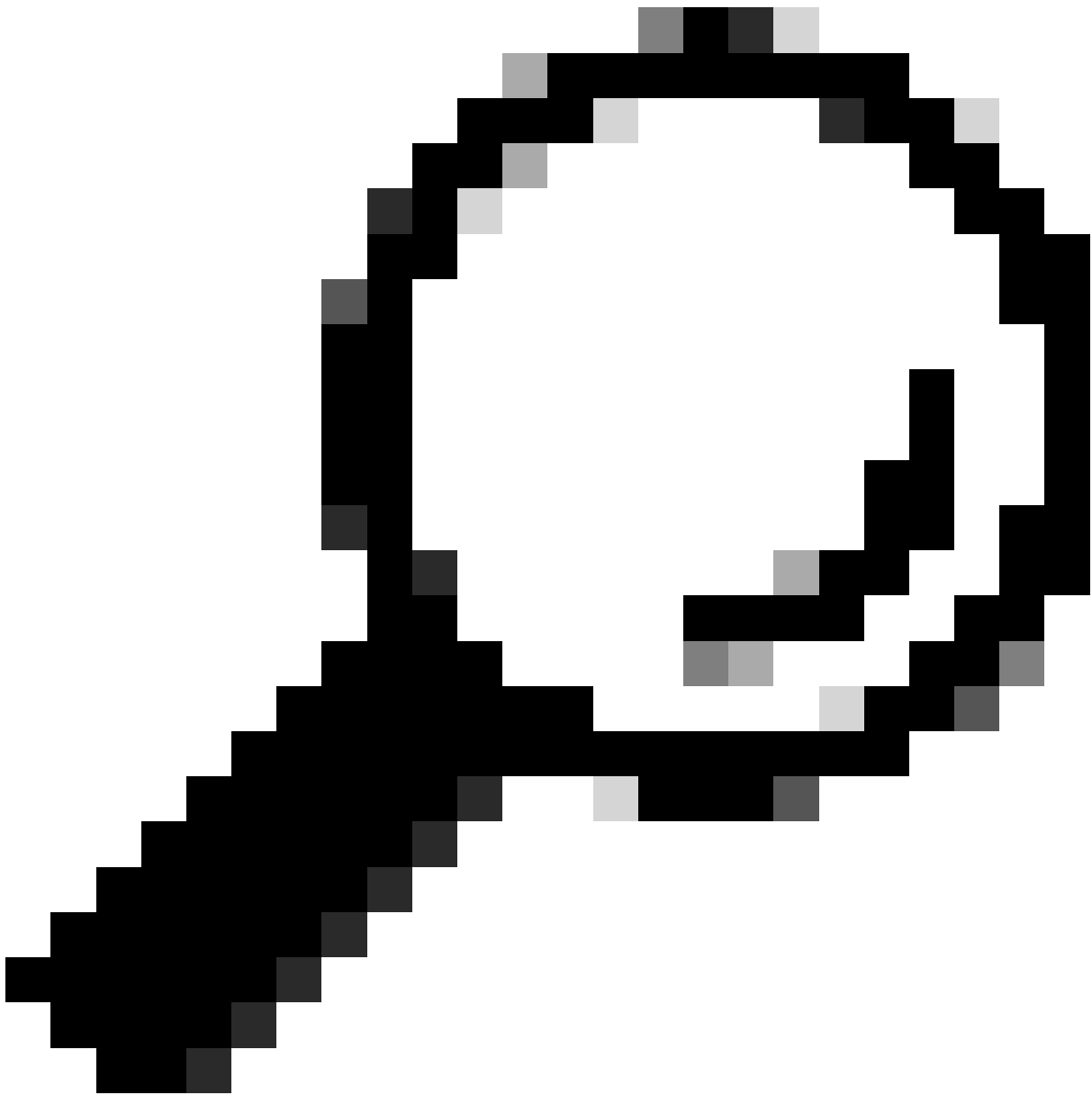
Note: While Cisco does its best to maintain a broad internal testing environment, we are unable to maintain an environment covering every model and version for a broad range of vendors. As such we can require your help to provide access to a device depicting the behavior at issue.

When opening a case with Cisco TAC for a Cisco provided NED, provide:

- NSO Version
- NED trace set to logging level debug
- The NED trace must capture either a `compare-config` or `sync-from operation`
- The NED trace must capture the particular issue you want assistance with
- Any information about other devices or NEDs which also show an issue
- Any information about similar devices on the same NED which do not show an issue



Note: Netconf NEDs built using the NSO NED Builder tool are not supported by Cisco outside of any issues with the tool itself.



Tip: For NEDs provided by the Crosswork NSO Developers, use Technology: NMS (Network Management Services and Sub-Technology: Network Service Orchestrator (NSO) - NED
