Enable NSO Logs and Verbosities

Contents

Introduction

This document describes the various logs available in NSO, what they are used for, and how to enable them.

Prerequisites

Requirements

To view, enable and set logs you require a user with access to the host environment running the NSO service, as well as access to the NSO CLI and NSO IPC port.

Components used

Cisco Crosswork Network Service Orchestrator (NSO) version 6.4.1

This document was written for the logging options available as of NSO 6.4. While most of the information in this document apply across versions, some logs can have been deprecated or added compared to the version you are using. This document does not cover configuration to export logs outside of the NSO system.

Commands provided in this document assume a system-install NSO using the default directory setup. In your environment the locations of certain files can differ.

- ncs.conf can be found in $NCS_CONFIG_DIR, by default /etc/ncs/ncs.conf
- Logs can be found in $NCS_LOG_DIR, by default /var/log/ncs/
- NSO is installed in $NCSDIR, by default /opt/ncs/
- NSO's running directory is $NCS_RUN_DIR, by default /var/opt/ncs/

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

General Log Guidelines

Logging Impact

Enabling logs at higher verbosity can cause increased load and disk-space requirements for the NSO server. This is especially a consideration for highly active logs such as devel.log. Enabling the verbosity for short periods of time during troubleshooting is generally not a concern but when enabling them for longer periods of time, make sure to take resources and disk-space into account.

Generating a Tech Report

To generate a tech report for NSO, run the script at **/opt/ncs/current/bin/ncs-collect-tech-report**.

Options:
--install-dir <InstallDir> : Specifies the directory for installation of NCS static files, like the --install-dir option to the installer.

--full : Collects an ncs-backup of the system, making it easier for Cisco support to reproduce any errors.

--num-debug-dumps : Default 1, Generates a debug-dump snapshot. For cases tracking resource leaks, such as memory/file descriptor leaks, set this to 3.

Recommended options:

```
/opt/ncs/current/bin/ncs-collect-tech-report --num-debug-dumps 3
```

A backup can be collected and provided separately to limit the file-size of the bundle for easier uploads.

The tech report is generated in the current directory from which the script is run.

Note: A tech report collects the content of the NSO log directory. Verify this directory does not contain any previous tech reports or backups prior to generating your new tech report.

Generating a Backup

```
/opt/ncs/current/bin/ncs-backup
```

Backups are generated in `/var/opt/ncs/backups/`.

Logfiles not Being Generated

When a logfile gets archived or deleted, NSO needs to create a new file. Usually this happens automatically but in case it did not, use command:

```
/opt/ncs/current/bin/ncs_cmd -c reopen_logs.
```

Note: When restricting access to the IPC port, for example, using the ipc-access setting in ncs.conf, make sure you define the necessary variables as part of cron or anacron so the weekly log rotation can properly re-open logs.

Overview of Logs

- NSO internal logs
  - ncs.log: The ncs log logs the main process of NSO. It has limited in depth information but can be used for issues involving shutdown, startup, the loading of packages, and upgrades.
  - audit.log: The audit log logs any users authenticating on NSO through any API. It also logs any activity on the NSO CLI and low-verbosity northbound interfaces.
  - audit-log-commit : Enabling this setting enhances the audit.log. It does not create its own log. It logs all non-default changes to NSO CDB during commit and sync-from operations.
  - audit-log-commit-defaults : Enabling this setting enhances the audit.log. It does not create its own log. It logs all default changes to NSO CDB during commit and sync-from operations.
  - devel.log: The devel log logs the general operations and workflows of NSO.
  - ncs-java-vm.log: The java log logs all java-vm related operations. Most notably any Network Element Driver (NED) and service packages written in Java. All CLI NEDs are written in java.
  - ncs-python-vm.log: The python logs log the activity relating to service packages written in

Python. A separate python log is generated for each service-package written in python. No NEDs are written in python.

  - upgrade.log: The upgrade log logs the changes in the NSO models during NSO upgrades, including NSO version upgrades and NSO package upgrades during packages reload.
  - raft.log: A log specifically for NSO clusters leveraging the HA-Raft capabilities.
  - xpath.trace: The xpath trace logs all xpath-evaluations NSO performs. This can be useful to figure out why a delete operation is taking a long time.
  - ncserr.log: The ncserr.log are binary logs recording errors for internal processes from the NCS daemon. Mandatory for almost any 'internal error' error messages and crash scenarios.
  - transerr.log: The transaction-error log is a log for collecting information on failed transactions that lead to either CDB boot error or runtime transaction failure.
  - progress.trace: The progress trace is used for tracing progress events emitted by transactions and actions in the system. What data to be emitted is configured in /progress/trace.
  - ncs-smart-licensing.log: Logs for the license smart-agent inside NSO.
- Northbound: Arriving to NSO from northbound elements
  - audit.log: The audit log logs commands executed on the NSO CLI.
  - localhost:8080.access/localhost:8888.access : This is an access log for the embedded Web server and collects HTTP activity. This file adheres to the Common Log Format, as defined by Apache
  - traffic.trace: This log collects very-high verbosity HTTP traffic. Use it to debug Restconf and json-rpc API.
  - netconf.log: Log for netconf API
  - netconf-trace.log: Log for high verbosity netconf API
  - json-rpc.log: Log for json-rpc.log API
- Southbound: Logging communication going from NSO to the network.
  - Device NED traces: Each device generates its own trace. Device traces are named as either ned-<ned-id>-<devicename>.trace or netconf-<devicename>.trace
  - audit-network.log: Records configuration commands sent by NSO to the southbound devices.
- System Logs
  - Linux logs: Typically found at /var/log/ and include logs such as messages or syslog. These vary depending on the host.
  - ncs_crash.dump: An NSO system dump generated when NSO terminates due to memory issues.
  - Core dump: When NSO is terminated for non-memory reasons, Linux can generate a core dump called core.<PID>

Certain conditions need to be met for Linux to generate a core dump. The ulimit configuration is the most common setting preventing a dump. See Linux Manual Page for a full list of requirements

Note: System logs are not collected by the NCS tech report, but can be useful for performance and crash related issues.

Enabling Logs and Setting Verbosity

Note: Changing the configuration settings in the ncs.conf file are applied by executing the `ncs --reload` command. `ncs --reload, it` reloads the values from the ncs.conf file and updates the running system, as well as closes and re-opens all log-files so any logging changes are applied. This does not interrupt services.

General Guidelines

- When specific configuration is not present in the ncs.conf file, NSO adopts the default behavior as specified in the `/opt/ncs/current/src/ncs/ncs_config/tailf-ncs-config.yang` file.
- When a log is specified as enabled by default, it means the log is enabled even if the configuration to enable it is missing.
- Some logs are disabled by default, but during the first installation of NSO, ncs.conf has specific instructions to enable the log.
- When specific configuration is not present in the ncs.conf file, you can add the configuration as shown under the `logs container`, meaning between `<logs>` and `</logs>` in the ncs.conf file.

Internal

ncs.log

This log is enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **change the content of <ncs-log>**.

```
<ncs-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/ncs.log</name>
    <enabled>true</enabled>
  </file>
</ncs-log>
```

After editing ncs.conf, execute **ncs --reload**.

audit.log

This log is enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **change the content of <audit-log>**.

```
<audit-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/audit.log</name>
    <enabled>true</enabled>
  </file>
</audit-log>
```

After editing ncs.conf, execute **ncs --reload**.

audit-log-commit and audit-log-commit-defaults

This log is not enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **Add the content after <audit-log>**.

```
<audit-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/audit.log</name>
    <enabled>true</enabled>
  </file>
</audit-log>
<audit-log-commit>true</audit-log-commit>
<audit-log-commit-defaults>true</audit-log-commit-defaults>
```

After editing ncs.conf, execute **ncs --reload**.

devel.log

This log is enabled by default at INFO verbosity. To enable and change verbosity for this log, **open /etc/ncs/ncs.conf** and **change the content of <developer-log>**.

```
<developer-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/devel.log</name>
    <enabled>true</enabled>
  </file>
</developer-log>
<developer-log-level>trace</developer-log-level>
```

After editing ncs.conf, execute **ncs --reload**.

ncs-java-vm.log

This log is enabled by default at INFO verbosity. It is possible to set verbosity for individual elements managed by java-vm. Verbosity is altered from the NSO CLI which can be accessed through SSH or **ncs_cli -C -noaaa**

To increase the verbosity across all java elements under com.tailf:

config
java-vm java-logging logger com.tailf level level-trace
commit no-networking

To increase the verbosity for a specific NED package:

config

java-vm java-logging logger com.tailf.packages.ned.<NED-name> level level-trace

commit no-networking

To increase the verbosity for the SSHJ client used in java NED packages:

config

java-vm java-logging logger net.schmizz.sshj level level-error

commit no-networking

Note: Cisco recommends setting logging for the SSHJ client to level-error. It is disabled by default.

To revert the logging for a specific java element:

config

no java-vm java-logging logger com.tailf

commit no-networking

To view current java-vm logging settings:

show running-config java-vm java-logging

ncs-python-vm.log

This log is enabled by default at INFO verbosity. Verbosity is altered from the NSO CLI which can be accessed through SSH or **ncs_cli -C -noaaa**.

To set verbosity for logs of all Python VMs.

config
python-vm logging level level-debug
commit no-networking

To revert back:
config
no python-vm logging level level-debug
commit no-networking

To view current python-vm logging settings:
show running-config python-vm logging

upgrade.log

This log is enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **change the content of <upgrade-log>**.

```
<upgrade-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/upgrade.log</name>
    <enabled>true</enabled>
  </file>
</upgrade-log>
```

After editing ncs.conf, execute **ncs --reload**.

raft.log

This log is enabled by default at INFO verbosity. To enable and set verbosity for this log, **open /etc/ncs/ncs.conf** and **change the content of <raft-log>**.

```
<raft-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/raft.log</name>
    <enabled>true</enabled>
  </file>
  <level>trace</level>
</raft-log>
```

After editing ncs.conf, execute **ncs --reload**.

xpath.trace

This log is not enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **change the content of <xpath-trace-log>**.

```
<xpath-trace-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/xpath.trace</filename>
</xpath-trace-log>
```

After editing ncs.conf, execute **ncs --reload**.

ncserr.log

This log records a limited amount of information. NSO maintains 5 error files, each with a maximum size of 1MB by default. In the rare situation where an issue occurs that creates more than 5MB in log-data, you need to increase the maximum size. This log is enabled by default. To alter the max size of this log to 10MB per file, **open /etc/ncs/ncs.conf** and **change the content of <error-log>**.

```
<error-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/ncserr.log</filename>
  <max-size>S10M</max-size>
</error-log>
```

After editing ncs.conf, execute **ncs --reload**.

transerr.log

This log is not enabled by default, but enabled in ncs.conf on first install. To enable this log **open /etc/ncs/ncs.conf** and **change the content of <transaction-error-log>**.

```
<transaction-error-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/transerr.log</filename>
</transaction-error-log>
```

After editing ncs.conf, execute **ncs --reload**.

progress.trace

This log is not enabled by default, but enabled in ncs.conf on first install. To enable this log **open /etc/ncs/ncs.conf** and **change the content of <progress-trace>**.

```
<progress-trace>
  <enabled>true</enabled>
  <dir>${NCS_LOG_DIR}</dir>
</progress-trace>
```

After editing ncs.conf, execute **ncs --reload**.

ncs-smart-licensing.log

This log is not enabled by default. The log is enabled from the NSO CLI which can be accessed through SSH or **ncs_cli -C -noaaa**. To enable this log:

config

smart-license smart-agent stdout-capture enabled

commit no-networking

To revert the logging change:

config

no smart-license smart-agent stdout-capture enabled

commit no-networking

Northbound

localhost:xxxx.access

This log is enabled by default. The name of this log varies based on the HTTP port. By default 8080 and 8888. To enable this log **open /etc/ncs/ncs.conf** and **change the content of <webui-access-log>**.

```
<webui-access-log>
  <enabled>true</enabled>
  <dir>${NCS_LOG_DIR}</dir>
</webui-access-log>
```

After editing ncs.conf, execute **ncs --reload**.

traffic.trace

This log is not enabled by default. traffic.trace logs are generated in a directory such as /var/log/ncs/trace_20240628_010010/. To enable this log **open /etc/ncs/ncs.conf** and **change the content of <webui-access-log>**.

```
<webui-access-log>
  <enabled>true</enabled>
  <dir>${NCS_LOG_DIR}</dir>
  <traffic-log>true</traffic-log>
</webui-access-log>
```

After editing ncs.conf, execute **ncs --reload**.

netconf.log

This log is enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **Add the content after <netconf-log>**.

```
<netconf-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/netconf.log</name>
    <enabled>true</enabled>
  </file>
</netconf-log>
```

After editing ncs.conf, execute **ncs --reload**

Additional option: Insert <log-reply-status>true</log-reply-status> after </file> to have NSO log the rpc-reply status "ok" or "error".

netconf-trace.log

This log is not enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **change the content of <netconf-trace-log>**.

```
<netconf-trace-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/netconf-trace.log</filename>
</netconf-trace-log>
```

After editing ncs.conf, execute **ncs --reload**.

json-rpc.log

This log is not enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **Add the content after <jsonrpc-log>**.

```
<jsonrpc-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/json-rpc.log</name>
    <enabled>true</enabled>
  </file>
</jsonrpc-log>
```

After editing ncs.conf, execute **ncs --reload**.

Southbound

Device NED Trace

This log is not enabled by default. The log is enabled from the NSO CLI which can be accessed through SSH or **ncs_cli -C -noaaa**.

To enable a trace for a device:

```
config
devices device <devicename> trace raw
devices device <devicename> ned-setting <ned-id> logger level debug
commit no-networking
```

To view all log-settings applied to a device, use **show devices device <devicename> active-settings**.

To clear the content of a device-trace file, use **devices device <devicename> clear-trace**.

To disable the device trace:

```
config

no devices device <devicename> trace

commit no-networking
```

audit-network.log

This log is not enabled by default. To enable this log, **open /etc/ncs/ncs.conf** and **Add the content after <audit-network-log**>.

```
<audit-network-log>
  <enabled>true</enabled>
  <file>
    <name>${NCS_LOG_DIR}/audit-network.log</name>
    <enabled>true</enabled>
  </file>
</audit-network-log>
```

After editing ncs.conf, execute **ncs --reload**.