



Command Line Manual for Troubleshooting NATkit 3

This manual is for the network administrator or operator who installs, configures, verifies, and uses the Network Analysis Toolkit (NATkit) tool. The network administrator or operator should have the following skills:

- Basic Unix system administrator skills
- Basic network management skills

How to Enter a Command

Change directory to where the command script is located. Enter the command with the appropriate parameters. In most cases, commands are preceded by the rcmd prompt:

./rcmd.command



Note

Commands and directory names are CASE SENSITIVE.

For example:

```
cd /opt/CSCONsa/bin/NSA/supportutils  
./rcmd view-current-devs
```

NATkit Directory Information

The following directories are the directories you will potentially use while supporting NATkit:

Directory	Function
/opt/CSCONsa/bin/NSA/supportutils	Support Utilities (Command lines)
/opt/CSCONsa/bin/NSA/common/env	Common Configuration files
/opt/CSCONsa/htdocs/COMPANIES/ NATkitID#/out/logs	Log & Error files
/opt/CSCOPx/databases/NATkit	Database files

NATkit Device Manager

view-current-devs (View Current Devices)

Description:	To view managed and not responsive devices in NATkit database.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd view-current-devs [-m Domain ALL]
Parameters:	Domain: Any user defined domain on NATkit. ALL: For all domains defined on NATkit. This is the default option.

adddevices.sh <seedfile> (Add Devices to RME Database)

Description:	Add devices to RME database by importing a seedfile. The seedfile consists of devices with telnet and snmp access information in csv or dif format specified by RME.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd adddevices.sh <seedfile>
Parameters:	seedfile: File consisting list of devices in a format specified by RME

view-device-aliases (View Device Aliases)

Description:	A device could be associated with many aliases. To view all aliases associated with a managed device execute this script.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd view-device-aliases [-d DeviceList] [-m Domain ALL]
Parameters:	DeviceList: List of devices separated by a comma. Ex: 171.68.111.30,171.68.111.31 Domain: Any user defined domain on NATkit. ALL: For all Domains defined on NATkit. This is the default option

view-access-verifier (View Access Verification Results)

Description:	This script accesses the NATkit database and prints the Access Verification result for the devices specified.
Directory:	/opt/CSCONsa/bin/NSA/supportutils

Usage:	<code>./rcmd view-access-verifier [-d devicelist] [-m Domain ALL] [-t access type]</code>
Parameters:	<p>devicelist: List of devices separated by a comma. Ex: 171.68.111.30,171.68.111.31</p> <p>Domain: Any user defined domain on NATkit.</p> <p>ALL: For all Domains that are defined on NATkit. This is the default option.</p> <p>Access type: Values entered could be “Ping”, “SNMP” or “Telnet”.</p>

run_access_verifier (Run Access Verification – Old Version)

Description:	This older script is from Version 2.0 NATkit and runs Access Verification. It can also be executed on version 3.0. Once this script is executed, it schedules an Access Verification task and provides the task name, which can then be viewed via the view-scheduler script.
Directory:	<code>/opt/CSCONsa/bin/NSA/supportutils</code>
Usage:	<code>./rcmd run_access_verifier -d deviceList - Help</code>
Parameters:	<p>deviceList: List of devices separated by a comma OR a wild character % can be specified OR it could be for all devices by specifying ALL.</p> <p>Examples:</p> <p>171.68.111.30,171.68.111.31</p> <p>71.68%</p> <p>ALL</p>



Note Help provides the usage of the command.

verify_device_access (Run Access Verification – New Version)

Description:	This script is the new version of Access Verification for NATkits running Version 3.0 and higher. This script is a wrapper to call the java CLI for verifying the access for all devices. The output of this command can be seen on the screen.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd verify_device_access ALL SELECTED <xml_file_name> DEVICES telnet-id1 telnet-id2 telnet-id3 ... --help
Parameters:	ALL: Access Verification done for all devices. SELECTED: Access Verification done only on certain devices listed in the xml file. DEVICES: List of devices separated by a space.

sync-seed-file (Synchronize seedfile)

Description:	This script is used to synchronize seedfile from local RME database or remote NMS RME database to NATkit database.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd sync-seed-file {local} {<nms-hostname> ..}
Parameters:	local: This is the local RME database. This is the default option. nms-hostname: Host name of the remote NMS machine with which NATkit database needs to be synchronized.

restart-sfsync-server (Restart NATkit Synchronization Server)

Description:	If the Seedfile synchronization from a remote NMS machine and NATkit is not functioning properly, this script is used to stop and restart the synchronization server on local NATkit machine. This script stops and restarts the following services: RmiRegistry, NATkitSyncServer, local NmsLsnr, NATkitFileCopyServer, NmsCfgMgrServer and NmsInvMgrServer. After stopping and restarting the above-mentioned services, sync-seed-file should be used to synchronize the seedfiles.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd restart-sfsync-server
Parameters:	none

devmgr-nms-sts (Show server status for seedfile synchronization)

Description:	Show server status for remote and local machines. The output of this command shows the status of the seedfile listener on either NATkit or the NMS machine where it can be concluded if the remote synchronization is happening properly.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd devmgr-nms-sts
Parameters:	none

devmgr-nms-sf-exp (Export seedfile from local RME or remote RME machine)

Description:	Export the device list and access information in local RME or remote RME machine to a file in RME version 1.0 format. The output of the command is displayed on the screen and can be redirected to a file. Once the file is exported then the file can be edited using Excel.
Directory:	/opt/CSCONsa/bin/NSA/supportutils

Usage:	<code>./rcmd devmgr-nms-sf-exp hostname > file_name</code>
Parameters:	hostname: Name of the local NATkit machine if remote integration is not done OR name of the remote RME machine if remote integration is done. file_name: Name of the file to which seedfile needs to be exported.

NATkit Download

download_settings (View/Edit download module attributes)

Description:	This script provides a menu to view or edit the download module settings. Download module will gather data for only those modules that have been selected. Options are presented from the script. Use -1 to exit the script in 3.0 version and option 3 to exit in 3.1 version.
Directory:	<code>/opt/CSCONsa/bin/NSA/supportutils</code>
Usage:	<code>./rcmd download_settings</code>
Parameters:	none

start_download (Download data to Cisco)

Description:	This script initiates the download process on NATkit.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd start_download
Parameters:	none

**Note**

This process can be initiated and run in the background mode

Example: ./rcmd start_download & (where & makes the process run in background).

NATkit Daily Report

start_dlyrpt (Daily Report)

Description:	To start the 24 hr daily report this script is executed. This script can be run in the background.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd start_dlyrpt
Parameters:	none

**Note**

This process can be initiated and run in the background mode.

Example: ./rcmd start_dlyrpt & (where & makes the process run in background).

NATkit Syslog Tasks

schedule-syslog (Schedule syslog on NATkit)

Description:	This script is used to schedule the syslog spooling utility from command line. In order to execute this script, the remote syslog directory should be NFS mounted on NATkit or NATkit should be configured to collect syslog locally. Custom correlation cannot be done from command line. Only Default or NO Correlation can be done from command line. If this script is executed with the help option then examples on how to run the script is given.
Directory:	/opt/CSCONSA/bin/NSA/supportutils
Usage:	/rcmd schedule-syslog [-start MM/DD/YYYY-hr:mn:sc OR now] [-end MM/DD/YYYY-hr:mn:sc OR never] [-pt Periodic Type ONCE/DAILY/WEEKLY/MONTHLY/HOURS/MINUTES/SECONDS] [-pu Periodic units 10/20/30/40 ...] [-file <filename>] [-n <Task Name>] [-correlation [NONE OR DEFAULT] [-help]
Parameters:	filename: Name of the syslog file to parse. Task Name: Input any name for this task. MM: month - [0 - 11] DD: day - [0 - 31] YYYY: year - [1900 - ...] hr: hours - [0 - 23] mn: minutes - [0 - 59] sc: seconds - [0 - 59]

view-syslog (View Syslog Messages)

Description:	This is an interactive script that is used to view syslog messages that are stored in NATkit database. Once the script is used with the format mentioned in Usage, the user is asked to select how to view the message.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	<code>./rcmd view-syslog -d <devicename> [-start <MM/DD/YYYY-hr:min:sec> -end <MM/DD/YYYY-hr:min:sec OR now> -message <message>]</code>
Parameters:	<p>devicename: Device name OR a wild card. EX: nsa-gw.cisco.com OR nsa%</p> <p>message: Syslog message OR a wild card. EX: SYS-5-CONFIG_I OR SYS-5-%</p> <p>MM: month - [0 - 11]</p> <p>DD: day - [0 - 31]</p> <p>YYYY: year - [1900 - ...]</p> <p>hr: hours - [0 - 23]</p> <p>mn: minutes - [0 - 59]</p> <p>sc: seconds - [0 - 59]</p>

WAN Switches

schedule-trapd (Schedule trapd task)

Description:	This script is used to schedule the trapd spooling utility from command line. In order to execute this script, the remote trapd log directory should be NFS mounted on NATkit.
Directory:	/opt/CSCONsa/bin/NSA/supportutils

Usage:	<pre> /rcmd schedule-trapd [-start MM/DD/YYYY-hr:mn:sc OR now] [-end MM/DD/YYYY-hr:mn:sc OR never] [-pt Periodic Type ONCE/DAILY/WEEKLY/ MONTHLY/HOURS/MINUTES/SECONDS] [-pu Peridoic units 10/20/30/40 ...] [-file <filename>] [-n <Task Name>] [-h <Help>] </pre>
Parameters:	<pre> filename: Name of the trapd log file to parse. Task Name: Input any name for this task. MM: month - [0 - 11] DD: day - [0 - 31] YYYY: year - [1900 - ...] hr: hours - [0 - 23] mn: minutes - [0 - 59] sc: seconds - [0 - 59] </pre>

view-trapd (View trapd messages from NATkit)

Description:	This is an interactive script that is used to view trapd messages that are stored in NATkit database. Once the script is executed in the format mentioned in Usage, the user is asked to select how to view the message.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd view-trapd [-start MM/DD/YYYY hr:mn:sc OR yesterday] [-end MM/DD/YYYY hr:mn:sc OR now] [-p pagebreak size] [-help]
Parameters:	pagebreak size: number of messages to be view in one page or window. MM: month - [0 - 11] DD: day - [0 - 31] YYYY: year - [1900 - ...] hr: hours - [0 - 23] mn: minutes - [0 - 59] sc: seconds - [0 - 59]

schedule-wan-cli (Schedule WAN CLI)

Description:	This is an interactive script which when executed in the format mentioned in the Usage, asks the user to select the WAN Switch nodes and the command profile to run the cli. So, execute the script and wait for the script to prompt for the node and the command profile selection.
Directory:	/opt/CSCONsa/bin/NSA/supportutils

Usage:	/rcmd schedule-wan-cli [-start MM/DD/YYYY-hr:mn:sc OR now] [-end MM/DD/YYYY-hr:mn:sc OR never] [-pt Periodic Type ONCE/DAILY/WEEKLY/MONTHLY/HOURS/MINUTES/SECONDS] [-pu Periodic units 10/20/30/40 ...] [-n <Task Name>] [-h <Help>]
Parameters:	Task Name: Input any name for this task. MM: month - [0 - 11] DD: day - [0 - 31] YYYY: year - [1900 - ...] hr: hours - [0 - 23] mn: minutes - [0 - 59] sc: seconds - [0 - 59]

wan_node_discovery (WAN Node Discovery)

Description:	Discover other WAN devices by logging into the devices specified as an input to this script.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd wan_node_discovery device_name1 <device_name2> <device_name3> ... <device_namen>
Parameters:	device_name1: Host name of any WAN device that is already managed in the RME database. Make sure that the telnet information for this device is correct in the NATkit database.

NATkit Scheduler

view-scheduler (View scheduled tasks on NATkit)

Description:	View the details of all the scheduled tasks on NATkit. The output of this script is displayed on the screen.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd view-scheduler
Parameters:	none

change-task-status (Change Task Status)

Description:	This is an interactive script where the user is asked to select the task that is already scheduled on NATkit to change the task context, activate or delete the task.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd change-task-status
Parameters:	none

stop-scheduler (Stop NATkit scheduler)

Description:	This script is used to stop the NATkit scheduler.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd stop_scheduler
Parameters:	none

start_scheduler (Start NATkit scheduler)

Description:	This script is used to start the NATkit scheduler.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd start_scheduler
Parameters:	none

NATkit Poller

plr_change_task_status (Change Poller task status) plr_schedule_polling pir_schedule_polling

Description:	This is an interactive script where the user is asked to select the poller task that is already scheduled on NATkit to change the task status to running, suspend or delete the task.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd plr_change_task_status
Parameters:	none

(Schedule a new poller task)

Description:	This is an interactive script where the user is asked to select the device category, device name, MIB and the time to schedule the poller task.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd plr_schedule_polling
Parameters:	none

plr_view_polled_values (View data collected by poller task)

Description:	This is an interactive script to view the data that is collected by the scheduled poller tasks. The user is asked to select the scheduled poller task and provide the start and end time to view the collected data by that task.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd plr_view_polled_values
Parameters:	none

plr_view_scheduled_tasks

Description:	This script shows all the poller tasks that are currently scheduled on NATkit.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd plr_view_scheduled_tasks
Parameters:	none

NATkit Purger

purge-data (Delete or purge data from NATkit database)

Description:	This script deletes data from NATkit database. The database stores data for 14 days. The number of days the database stores the data can be set using <code>purge_settings</code> script.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd purge-data
Parameters:	none

purger_settings (Edit and View purger settings)

Description:	This is an interactive script to view and edit the purger settings. By default all the modules on NATkit store the data for 14 days in the database. By running this script the user can change the number of days the front end stores the data.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd purger_settings
Parameters:	none

NATkit Database

db_table_backup (Extract or Restore data to NATkit database)

Description:	This script is used to extract or restore data to NATkit database. When the data is extracted from the database it is stored in /opt/CSCONsa/bin/NSA/database/.backup directory. When the script is called to restore the data back to the database then all the files in the “.backup” directory is restored into the NATkit database.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd /opt/CSCONsa/bin/NSA/db_table_backup [-m Maintenance_mode] [-help]
Parameters:	Maintenance_mode: backup or restore backup: Backup data from NATkit database. restore: Restore data back to NATkit database. Before running this ou need to recreate the database. But before recreating the database please make sure that the /opt/CSCONsa/bin/NSA/database/.backup has the required table files. Then stop the database engine by running this command : /etc/init.d/NATkitdmgtd stop and then recreate the database /opt/CSCONsa/bin/NSA/supportutils/rcmd /opt/CSCONsa/bin/NSA/database/create.database -n NATkitID

NATkit Device Discovery

start-discovery (Schedule Phase 1 Device Discovery)

Description:	This script is used to start the device discovery process. Before scheduling the device discovery process the discovery parameters should be set in the respective configuration files specified in directory mentioned below.
Directory:	To schedule the task: /opt/CSCONsa/bin/NSA/supportutils To edit configuration files: /opt/CSCOPx/objects/CSCOad/nsa/NATkit/data/config
Usage:	./rcmd start-discovery
Parameters:	none



Note

Before scheduling device discovery the following files in the directory mentioned above need to be edited to have the correct parameters.

param.txt : Edit this file to include the Discovery method, Include filters, Exclude filters, start time.

stop-discovery (Stop Phase 1 Device Discovery)

Description:	This script is used to stop phase 1 device discovery.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	./rcmd stop-discovery
Parameters:	none

start-discovery2 (Start Phase 2 Device Discovery)

Description:	This script is used to start the phase 2 device discovery. Once the devices have been discovered this script can be used to build the seedfile with all the possible passwords and read_write community strings.
Directory:	To schedule the task: /opt/CSCONsa/bin/NSA/supportutils To edit configuration files: /opt/CSCOPx/objects/CSCOad/nsa/NATkit/data/config
Usage:	/rcmd start-discovery2
Parameters:	none

**Note**

Before scheduling phase 2 device discovery the following files in the directory mentioned above need to be edited. logpass.txt: Edit this file to include the possible password for the devices.

logpassprompt.txt: Edit this file to include the possible login prompts on the devices.

stop-discovery2 (Start Phase 2 Device Discovery)

Description:	This script is used to stop phase 2 device discovery.
Directory:	/opt/CSCONsa/bin/NSA/supportutils
Usage:	/rcmd stop-discovery2
Parameters:	none

NATkit RME Daemon Process

dmgtd start (Start Daemon Manager)

Description:	This script is used to start the daemon manger process.
Directory:	/etc/init.d/dmgtd
Usage:	/dmgtd start
Parameters:	none

dmgtd stop (Stop Daemon Manager)

Description:	This script is used to stop the daemon manger process.
Directory:	/etc/init.d/dmgtd
Usage:	/dmgtd stop
Parameters:	none