



# **N2OS Configuration Reference Guide**

## Legal notices

*Information about the Nozomi Networks copyright and use of third-party software in the Nozomi Networks product suite.*

### Copyright

Copyright © 2013-2024, Nozomi Networks. All rights reserved. Nozomi Networks believes the information it furnishes to be accurate and reliable. However, Nozomi Networks assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of Nozomi Networks except as specifically described by applicable user licenses. Nozomi Networks reserves the right to change specifications at any time without notice.

### Third Party Software

Nozomi Networks uses third-party software, the usage of which is governed by the applicable license agreements from each of the software vendors. Additional details about used third-party software can be found at <https://security.nozominetworks.com/licenses>.

# Contents

<b>Chapter 1. Features control panel.....</b>	<b>7</b>
Features control panel.....	9
General.....	10
Retention.....	11
<b>Chapter 2. Edit sensor configuration.....</b>	<b>13</b>
Edit sensor configuration.....	15
<b>Chapter 3. Basic rules.....</b>	<b>17</b>
Basic configuration rules.....	19
<b>Chapter 4. Garbage Collector.....</b>	<b>31</b>
Configure Garbage Collector.....	33
<b>Chapter 5. Alerts.....</b>	<b>37</b>
Configure alerts.....	39
SIGN:MULTIPLE-ACCESS-DENIED.....	45
SIGN:MULTIPLE-UNSUCCESSFUL-LOGINS.....	46
SIGN:OUTBOUND-CONNECTIONS.....	47
SIGN:TCP-SYN-FLOOD.....	50
SIGN:UDP-FLOOD.....	52
SIGN:NETWORK-SCAN.....	53
<b>Chapter 6. Incidents.....</b>	<b>61</b>
INCIDENT:PORT-SCAN.....	63
<b>Chapter 7. Nodes.....</b>	<b>65</b>
Configure nodes.....	67
<b>Chapter 8. Assets.....</b>	<b>73</b>
Configure assets.....	75
<b>Chapter 9. Links.....</b>	<b>77</b>
Configure links.....	79
<b>Chapter 10. Variables.....</b>	<b>85</b>
Configure variables.....	87
<b>Chapter 11. Protocols.....</b>	<b>93</b>
Configure protocols.....	95
<b>Chapter 12. Vulnerability assessment.....</b>	<b>109</b>
Configure vulnerability assessments.....	111
<b>Chapter 13. Customize node identifier generation.....</b>	<b>115</b>
Customize node identifier generation.....	117

<b>Chapter 14. Decryption.....</b>	<b>119</b>
IEC 60870-5-7 / 62351-3/5 encrypted links.....	121
Configure IEC-62351-3.....	122
<b>Chapter 15. Trace.....</b>	<b>125</b>
Configure trace.....	127
Configure continuous trace.....	130
<b>Chapter 16. Time Machine.....</b>	<b>133</b>
Configure Time Machine.....	135
<b>Chapter 17. Retention.....</b>	<b>137</b>
Configure retention.....	139
<b>Chapter 18. Bandwidth throttling.....</b>	<b>145</b>
Configure bandwidth throttling.....	147
<b>Chapter 19. Synchronization.....</b>	<b>149</b>
Configure synchronization.....	151
<b>Chapter 20. Slow updates.....</b>	<b>157</b>
Configure slow updates.....	159
<b>Chapter 21. Session hijacking.....</b>	<b>161</b>
Configure session hijacking protection.....	163
<b>Chapter 22. Passwords.....</b>	<b>165</b>
Configure passwords.....	167
Set maximum attempts.....	168
Enable pbkdf2 sha512 password hashing algorithm.....	169
Set lock time.....	170
Set history.....	171
Set password digits.....	172
Set password lower.....	173
Set password upper.....	174
Set password symbol.....	175
Set password min length.....	176
Set password max length.....	177
Enable expire for inactive users.....	178
Set user lifetime.....	179
Enable expiration for admin users.....	180
Enable password expiration.....	181
Set password lifetime.....	182
<b>Chapter 23. Sandbox.....</b>	<b>183</b>
Sandbox archive processing.....	185
Set the number of workers.....	186
Set the size of the sandbox tmpfs partition.....	187

Set the size of the sandbox tmpfs temporary partition.....	188
Set the size of the sandbox tmpfs pipes partition.....	189
Configure how Threat Intelligence contents are handled.....	190
Set sandbox strategies for vulnerability assessment.....	191
Set the minimum percentage of free disk.....	192
Set the maximum number of files to retain.....	193
Set the size of the asynchronous processing queues.....	194
Set the interval for stale file analysis.....	195
Set the maximum number of extracted files.....	196
Set the maximum level of nested extracted files.....	197
Set the maximum size of a single extracted file.....	198
Set the maximum overall size of extracted files.....	199
Set the maximum time to be spent for each file extraction.....	200
Define maximum recursion depth allowed for VBA extraction.....	201
Enable or disable the adaptive algorithm for file unzipping.....	202
Configuring handling of sandboxed zipped files.....	203
Configure high throughput protection.....	204
<b>Chapter 24. Miscellaneous.....</b>	<b>207</b>
Configure SSH key update interval.....	209
Configure paranoid mode for user login authentication.....	210
Configure identity provider.....	211
Glossary.....	213



# Chapter 1. Features control panel





## Features control panel

The **Features control panel** page shows an overview of the current status of system features configuration and lets you fine tune specific values.

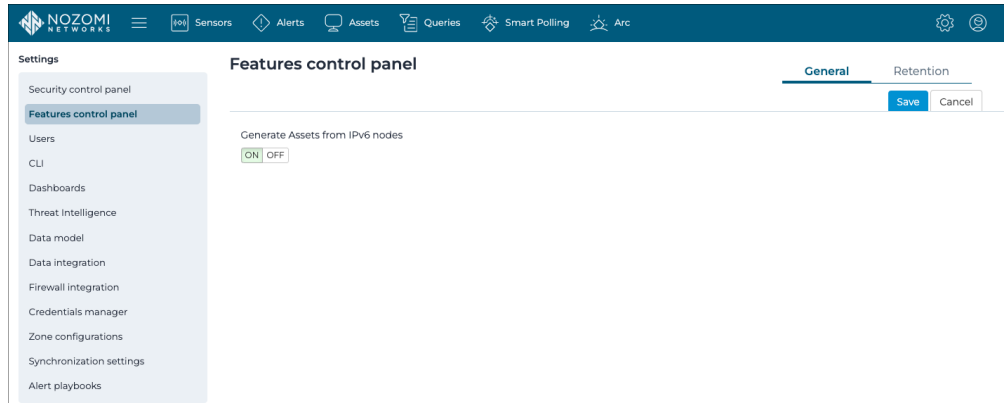


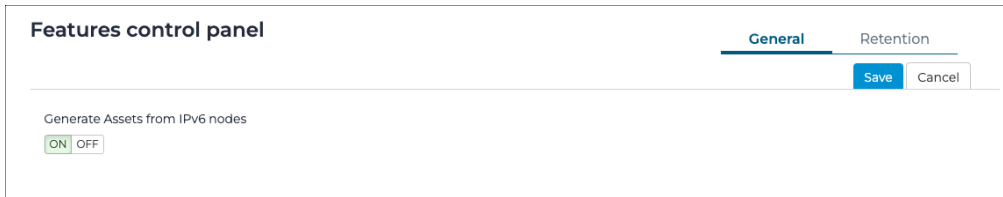
Figure 1. Features control panel page

The **Features control panel** page has these tabs:

- [General](#) (on page 10)
- [Retention](#) (on page 11)

## General

The **General** page lets you generate assets from IPv6 nodes.



The screenshot shows a web interface titled "Features control panel". It has two tabs: "General" (which is active) and "Retention". Below the tabs are two buttons: "Save" (in blue) and "Cancel" (in white). The main content area contains the text "Generate Assets from IPv6 nodes" followed by a toggle switch. The toggle switch is currently in the "ON" position, indicated by a green background and the word "ON" in white text, with "OFF" in grey text to its right.

Figure 2. General page

## Retention

The **Retention** page lets you select a specific number, or **Retention level**, for historical data persistence. In some cases, you can either completely disable the retention of a feature, or enable the advanced options that provide more specific settings.

**Features Control Panel** General **Retention** Save Cancel

**Alerts**  
 ⓘ When an alert is deleted, the related trace file is deleted too  
 Retention level: **up to 500,000 items**  
 Advanced options: **ON** | OFF

**Captured logs**  
 Retention level: **up to 25,000 items**

**CLI action requests**  
 Retention level: **up to 100,000 items**

**Extended network statistics**  
 COLLECTING **Disabled**  
 ⓘ • COLLECTING: enables historical data persistence up to chosen value  
 • DISABLED: disables historical data persistence

**Link events**  
 Warning: Enable this feature only in small environments  
 COLLECTING **Disabled**  
 ⓘ • COLLECTING: enables historical data persistence up to chosen value  
 • DISABLED: disables historical data persistence

**Node CPE changes**  
 Retention level: **up to 100,000 items**

**Node points**  
 Retention level: **up to 1,000,000 items**

**Reports saved locally**  
 Retention level: **up to 500 items**  
 Expiration: **90 days**

**Time machine snapshots**  
 Space retention level: **512 MB**  
 Retention level: **up to 50 items**

**Traces: generated continuous**  
 Space retention level: **2.25 GB**  
 Retention level: **up to 10,000 items**

**Traces: uploaded**  
 Retention level: **up to 10 items**

**Variables history**  
 Retention level: **up to 1,000,000 items**

**Audit**  
 Retention level: **up to 100,000 items**  
 Expiration: **Never**

**Captured URLs**  
 Warning: Enable this feature only in small environments  
 COLLECTING **Disabled**  
 ⓘ • COLLECTING: enables historical data persistence up to chosen value  
 • DISABLED: disables historical data persistence

**Dashboard configurations**  
 Retention level: **up to 10,000 items**

**Health logs**  
 Retention level: **up to 5,000 items**

**Node CPEs**  
 Retention level: **up to 100,000 items**

**Node CVEs**  
 Retention level: **up to 400,000 items**

**Files quarantine**  
 ⓘ Applicable to monitored files reconstructed for analysis  
 Retention level: **up to 50 items**

**Scheduled updates**  
 Retention level: **up to 10,000 items**

**Smart Polling execution history**  
 Retention level: **up to 100,000 items**

**Traces: generated**  
 Retention level: **up to 10,000 items**  
 Advanced options: **ON** | OFF  
 Space retention level: **2.25 GB**

**User sessions**  
 ⓘ Applicable to UI user sessions. Users need to reconnect as their session is purged  
 Retention level: **up to 10,000 items**

Figure 3. Retention page

### Expiration

This lets you select a specific number of days for historical data persistence. To let the data persist forever, you can set this to **Never**.

### Retention level

This lets you select a specific number of items for historical data persistence.

### Space retention level

This lets you select a specific space size for historical data persistence.



## Chapter 2. Edit sensor configuration



## Edit sensor configuration

In Central Management Console (CMC) and Guardian sensors, use the command-line Interface (CLI) to configure the Nozomi Networks solution.

You can access the [command-line interface \(CLI\)](#) in two ways:

- use the `cli` command in a text-console when connected to the sensor, either directly or through [secure shell \(SSH\)](#)
- in the web GUI, select **Administration > Settings > CLI**

### Examples:

A command issued via the `cli` command in a shell:

```
root@ch-lab-sg-guardian-stable:~ # cli
> conf.user configure vi gc old_ghost_nodes 3600
{"result"=>true}
success
```

A command issued (through pipe) to the `cli` command in a shell:

```
root@ch-lab-sg-guardian-stable:~ # echo conf.user configure vi gc
old_ghost_nodes 3600 | cli
> {"result"=>true}
success
>root@ch-lab-sg-guardian-stable:~ #
```

There are cases, on Remote Collector sensors for example, where `cli` command doesn't work; in those cases or to fine-tune user-defined configuration or mass-import rules from other systems it's required to manually edit the `/data/cfg/n2os.conf.user`. In this section we will see how to change and apply a configuration rule.

Please log into the shell console, either through [SSH](#), or shell console, and issue the following commands.

- Use `vi` or `nano` to edit `/data/cfg/n2os.conf.user`
- Edit a configuration rule with the text editor, see the next sections for some examples
- Write configuration changes to disk and exit the text editor

Next sections cover all the necessary details about the supported configuration rules.





## Chapter 3. Basic rules



## Basic configuration rules

### Set traffic filter

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure bpf_filter &lt;bpf_expression&gt;</code>
<b>Description</b>	Set the <i>Berkeley Packet Filter (BPF)</i> filter to apply on incoming traffic to limit the type and amount of data processed by the sensor.
<b>Parameters</b>	<code>bpf_expression</code> : the Berkeley Packet Filter expression to apply on incoming traffic. A <i>BPF</i> syntax reference can be accessed on the sensor at <a href="https://&lt;sensor_ip&gt;/#/bpf_guide">https://&lt;sensor_ip&gt;/#/bpf_guide</a>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Enable or disable management filters

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure mgmt_filters [on off]</code>
<b>Description</b>	With this rule you can switch off the filters on packets that come from/ to N2OS itself. Choose 'off' if you want to disable the management filters (default: on).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Enable or disable TCP/UDP deduplication

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe deduplication enabled [true false]</code>
<b>Description</b>	It can enable or disable the deduplication analysis that N2OS does on TCP/UDP packets. it can be either true, to enable the feature, or false, to disable it. (default: true)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set TCP deduplication time delta

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe deduplication tcp_max_delta &lt;delta&gt;</code>
<b>Description</b>	Set the desired maximum time delta, in milliseconds, to consider a duplicated TCP packet.
<b>Parameters</b>	<code>delta</code> : The value of the maximum time delta (default: 1)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set UDP deduplication time delta

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe deduplication udp_max_delta &lt;delta&gt;</code>
<b>Description</b>	Set the desired maximum time delta, in milliseconds, to consider a duplicated UDP packet.
<b>Parameters</b>	<code>delta</code> : The value of the maximum time delta (default: 1)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Rename fallback zones

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi zones default [private public] &lt;zone_name&gt;</code>
<b>Description</b>	Set the <i>private</i> or <i>public</i> fallback zone name, for nodes not matching any zone. Details on zones feature can be viewed in the <b>Graph</b> page in <b>Network.Remark</b> : zones can be configured through the GUI, which is the preferred way. Refer to <b>Zone configurations</b> , in the <b>Administrator Guide</b> .
<b>Parameters</b>	<code>zone_name</code> : the name of the <i>private</i> or <i>public</i> fallback zone
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Add Zone

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi zones create &lt;subnet&gt;[,&lt;subnet&gt;] &lt;zone_name&gt;</code>
<b>Description</b>	Add a new zone containing all the nodes in one or more specified subnetworks. More subnetworks can be concatenated using commas. The subnetworks can be specified using the CIDR notation (<ip>/<mask>) or by indicating the end IPs of a range (both ends are included: <low_ip>-<high_ip>). <b>Remark:</b> zones can be configured through the GUI, which is the preferred way. Refer to <b>Zone configurations</b> , in the <b>Administrator Guide</b> .
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>subnet</code>: The subnetwork or subnetworks assigned to the zone; both IPv4 and IPv6 are supported</li> <li>• <code>zone_name</code>: The name of the zone</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Assign a level to a zone

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi zones setlevel &lt;level&gt; &lt;zone_name&gt;</code>
<b>Description</b>	Assigns the specified level to a zone. All nodes pertaining to the given zone will be assigned the level. <b>Remark:</b> zones can be configured through the GUI, which is the preferred way. Refer to <b>Zone configurations</b> , in the <b>Administrator Guide</b>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>level</code>: The level assigned to the zone</li> <li>• <code>zone_name</code>: The name of the zone</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Set the nodes ownership for a zone

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>ids configure vi zones setis_public [true false] &lt;zone_name&gt;</code>
<b>Description</b>	Sets the specified nodes ownership for a zone. It can be either true, for public ownership, or false, for private ownership. All nodes belonging to the given zone are overwritten inheriting the value. <b>Remark:</b> zones can be configured through the GUI, which is the preferred way. Refer to <b>Zone configurations</b> , in the <b>Administrator Guide</b>
<b>Parameters</b>	zone_name: The name of the zone
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Assign a security profile to a zone

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi zones setsecprofile [low medium high paranoid] &lt;zone_name&gt;</code>
<b>Description</b>	Assigns the specified security profile to a zone. The visibility of the alerts generated within the zone will follow the configured security profile. Refer to <b>Security profile</b> , in the <b>Administrator Guide</b> .
<b>Parameters</b>	zone_name: The name of the zone
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Add custom protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe custom-protocol &lt;name&gt; [tcp udp] &lt;port&gt;</code>
<b>Description</b>	Add a new protocol specifying a port and a transport layer. Names shall always be unique, so when defining a custom protocol both for udp and tcp, use two different names.

<b>Parameters</b>	<ul style="list-style-type: none"> <li><b>name</b>: The name of the protocol, it will be displayed through the user interface; <b>DO NOT</b> use a protocol name already used by N2OS. E.g. one can use MySNMP, or Myhttp</li> <li><b>port</b>: The transport layer port used to identify the custom protocol</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Disabling a protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol &lt;name&gt; enable false</code>
<b>Description</b>	Completely disables a protocol. This can be useful to fine tune the sensor for specific needs.
<b>Parameters</b>	<b>name</b> : The name of the protocol to disable
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set IP grouping

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe ipgroup &lt;ip&gt;/&lt;mask&gt;</code>
<b>Description</b>	<p>This command permits to group multiple ip addresses into one single node. This command is particularly useful when a large network of clients accesses the SCADA/ICS system. To provide a clearer view and get an effective learning phase, you can map all clients to a unique node simply by specifying the netmasks (one line for each netmask). <a href="#">Configure trace (on page 127)</a> will still show the raw IPs in the provided trace files.</p> <p><b>Warning:</b> This command merges all nodes information into one in an irreversible way, and the information about original nodes is not kept.</p>
<b>Parameters</b>	<b>ip/mask</b> : The subnetwork identifier used to group the IP addresses
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console, execute both: <code>service n2osids stop</code> AND <code>service n2ostrace stop</code>

## Set IP grouping for Public Nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe ipgroup public_ips &lt;ip&gt;</code>
<b>Description</b>	This command permits to group all public IP addresses into one single node (for instance, use 0.0.0.0 as the 'ip' parameter). This command is particularly useful when the monitored network includes nodes that have routing to the Internet. The <a href="#">Configure trace (on page 127)</a> , will still show the raw IPs in the provided trace files. <b>Warning:</b> This command merges all nodes information into one in an irreversible way, and the information about original nodes is not kept.
<b>Parameters</b>	<code>ip</code> : The ip to map all Public Nodes to
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console, execute both: <code>service n2osids stop</code> AND <code>service n2ostrace stop</code>

## Skip Public Nodes Grouping for a subnet

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe ipgroup public_ips_skip &lt;ip&gt;/&lt;mask&gt;</code>
<b>Description</b>	This is useful when the monitored network has a public addressing that has to be monitored (i.e. public addressing used as private or public addresses that are in security denylists).
<b>Parameters</b>	<code>ip/mask</code> : The subnetwork identifier to skip
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console, execute both: <code>service n2osids stop</code> AND <code>service n2ostrace stop</code>

## Set special Private Nodes allowlist

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi private_ips &lt;ip&gt;/&lt;mask&gt;</code>
<b>Description</b>	This rule will set the <code>is_public</code> property of nodes matching the provided mask to false. This is useful when the monitored network has a public addressing used as private (e.g. violation of RFC 1918).



<b>Parameters</b>	<code>ip/mask</code> : The subnetwork identifier to treat as private; both IPv4 and IPv6 are supported
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set GUI logout timeout

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure users max_idle_minutes &lt;timeout_in_minutes&gt;</code>
<b>Description</b>	Change the default inactivity timeout of the GUI. This timeout is used to decide when to log out the current session when the user is not active.
<b>Parameters</b>	<code>timeout_in_minutes</code> : amount of minutes to wait before logging out. The default is 10 minutes.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable Syslog capture feature

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol syslog capture_logs [true false]</code>
<b>Description</b>	With this configuration rule you can enable (option true) the passively capture of the syslog events. It is useful when you want to forward them to a SIEM, for more details see <b>Syslog forwarder integration</b> in the <b>Administrator Guide</b> .
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable Guardian HA

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure guardian replica-of &lt;other_guardian_id&gt;</code>

<b>Description</b>	With this configuration rule you can enable the Guardian HA mode for two Guardians that sniff the same traffic and are connected to the same CMC. During normal operations, only the primary Guardian syncs with the CMC; if it stops synchronizing the secondary Guardian will start synchronize the records from the last primary Guardian update. This rule should only be configured on the secondary sensor.
<b>Parameters</b>	<code>other_guardian_id</code> : The id of the other Guardian, it can be found on the CMC with the query <code>appliances   where host == &lt;appliance_hostname&gt;   select id</code>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Disabling Vulnerability Assessment for some nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va_notification matching [id label zone type vendor]=&lt;value&gt; discard</code>
<b>Description</b>	<p>With this configuration rule you can disable Vulnerability Assessment for node matching the specified rules. The effect of this configuration rule is to discard the matching of CVE identifiers. The types are as follows.</p> <ul style="list-style-type: none"> <li>• <code>id</code>: the id of a node, it can be an IP address, a netmask in the CIDR format or a MAC address.</li> <li>• <code>label</code>: the label of a node.</li> <li>• <code>zone</code>: the zone in which a node is located.</li> <li>• <code>type</code>: the type of a node.</li> <li>• <code>vendor</code>: the vendor of a node.</li> </ul>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>value</code>: If a simple string is specified the match will be performed with an "equal to" case-sensitive criterion. The matching supports two operators:</li> <li>• <code>^</code>: starts with</li> <li>• <code>[]</code>: contains</li> </ul> <p>These operators must be specified right after the = symbol and their match is case-insensitive.</p> <p>Examples:</p> <ul style="list-style-type: none"> <li>• <code>va_notification matching id=192.168.1.123 discard</code></li> <li>• <code>va_notification matching id=192.168.1.0/24 discard</code></li> <li>• <code>va_notification matching label=^abc discard</code></li> </ul>
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	In a shell console execute: <code>service n2osva stop</code>
-----------------	--

### Enabling IPv6 Assets

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ipv6_assets [enabled disabled]</code>
<b>Description</b>	With this configuration rule you can enable assets generation also when nodes are IPv6. By default, this feature is disabled.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Change the maximum percentage of Variables in the Network Elements pool

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi machine_limits_variables_quota &lt;n&gt;</code>
<b>Description</b>	With this configuration rule you can change the maximum percentage of Variables in the Network Elements pool, the default is 0.6 meaning that no more than 60% of Network Elements can be Variables.
<b>Parameters</b>	n: the percentage of variables expressed as a number from 0.0 to 1.0, e.g. <code>vi machine_limits_variables_quota 0.7</code>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Tuning Backend Web Server workers

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure http_workers &lt;n&gt;</code>
<b>Description</b>	With this configuration rule you can change the number of Ruby Web Server workers. With a higher workers count the CMC/Guardian can handle more Web UI requests concurrently, at the expense of increased memory footprint.
<b>Parameters</b>	n: The new number of workers
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	In a shell console execute: <code>service webserver stop</code>
-----------------	---

### Configure how Threat Intelligence contents are handled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi contents &lt;json_value&gt;</code>
<b>Description</b>	<p>This command configures how Threat Intelligence contents are to be loaded. The JSON object can have the following attributes:</p> <ul style="list-style-type: none"> <li>• <code>load_contents</code> - this can be true/false to enable/disable the loading of contents;</li> <li>• <code>stix_backend_provider</code> - selects the engine to be used for STIX indicators: 'memory' (indicators are stored in memory - default) or 'db' (indicators are stored in the system database). With 'db', the application memory requirements become lower, but STIXVI (XML) indicators are not supported;</li> <li>• <code>loaded_content_types</code> - this is a JSON array of contents to be loaded.</li> </ul> <p>Contents available are:</p> <ul style="list-style-type: none"> <li>• <code>stix_indicators</code></li> </ul> <p>As an example, the following command will completely disable contents loading:</p> <pre>conf.user configure vi contents { "load_contents": false }</pre> <p>As a further example, the following command will allow only <code>stix_indicators</code> rules to be loaded:</p> <pre>conf.user configure vi contents { "loaded_content_types": [ "stix_indicators" ] }</pre> <p>As another example, the following command will configure the usage of the system database for storing STIX indicators related information, reducing the application memory footprint:</p> <pre>conf.user configure vi contents { "stix_backend_provider": "db" }</pre>
<b>Parameters</b>	<code>json_value</code> : A JSON object to configure how Threat Intelligence contents are loaded
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Configure which files detected on the networks are sent to sandbox

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi sandbox_extraction &lt;json_value&gt;</code>

<b>Description</b>	The json object can have the following attributes: * <b>disabled_protocols</b> - A JSON array of protocol that are disabled with regards files detection * <b>enabled_protocols</b> - A JSON array of protocol that are enabled with regards files detection * <b>disabled_file_extensions</b> - A JSON array of file extensions that are disabled with regards files detection * <b>enabled_file_extensions</b> - A JSON array of file extensions that are enabled with regards files detection
<b>Parameters</b>	<ul style="list-style-type: none"><li>• <b>json_value</b>: A json object to configure the handling of the detected files.</li></ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <b>service n2osids stop</b>



# Chapter 4. Garbage Collector





## Configure Garbage Collector

This section describes how to configure the Environment Garbage Collector (GC). The Garbage Collector lets the system discard nodes, assets, and links that are no longer useful, thus saving system resources.

### Clean up old ghost nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_ghost_nodes &lt;seconds&gt;</code>
<b>Description</b>	Set the threshold after which idle nodes that are also not confirmed and not learned are discarded by the garbage collector. <b>NOTE:</b> in Adaptive Learning, the GC works also if nodes are learned, since they all are.
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (the default is 3600, the equivalent of one hour).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old public nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_public_nodes &lt;seconds&gt;</code>
<b>Description</b>	Determines how long to keep public nodes that are inactive. Expressed in seconds.
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (default is 259200, the equivalent of three days).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old inactive nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_inactive_nodes &lt;seconds&gt;</code>

<b>Description</b>	Determines how long to keep nodes that are inactive. Expressed in seconds. Inactivity is calculated as the difference between the current time and the last activity time.  <b>Note:</b> When a node that has been deleted by the garbage collector appears again in the network it will be considered new, as a consequence, according to the learning mode, an alert could be raised. For a better result, use Adaptive Learning and choose a reasonably long interval for this setting.
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (by default it's disabled).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old inactive links

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_inactive_links &lt;seconds&gt;</code>
<b>Description</b>	Determines how long to keep links that are inactive. Expressed in seconds. Inactivity is calculated as the difference between the current time and the last activity time.  <b>Note:</b> When a link that has been deleted by the garbage collector appears again in the network it will be considered new, as a consequence, according to the learning mode, an alert could be raised. For a better result, use Adaptive Learning and choose a reasonably long interval for this setting.
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (by default it's disabled).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old ghost links

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_ghost_links &lt;seconds&gt;</code>

<b>Description</b>	Determines how long to wait before removing inactive ghost links. A ghost link is one that has not shown any application payload since its creation. This could be a connection attempt whose endpoint is not responding on the specified port; or it could be a link with a successful handshake but without application data transmitted (in this case, transferred data would still be greater than 0).
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (by default it's disabled).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old inactive variables

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc old_inactive_variables &lt;seconds&gt;</code>
<b>Description</b>	<p>Determines how long to keep variables that are inactive. Expressed in seconds. Inactivity is calculated as the difference between the current time and the last activity time.</p> <p><b>Note:</b> When a <i>variable</i> that has been deleted by the garbage collector appears again in the network it will be considered new, as a consequence, according to the learning mode, an alert could be raised. For a better result, use Adaptive Learning and choose a reasonably long interval for this setting.</p>
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which cleanup occurs (by default it's disabled).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Clean up old sessions

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi gc sessions_may_expire_after &lt;seconds&gt;</code>
<b>Description</b>	Determines how long to wait before a session is considered stale and its resources may be collected. Expressed in seconds.
<b>Parameters</b>	<code>seconds</code> : Number of seconds after which clean up may occur. By default, set to 100 seconds.
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

## Chapter 5. Alerts



## Configure alerts

### Enable or disable alert deduplication

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts deduplication enable [true false]</code>
<b>Description</b>	Enable or disable the alert deduplication capabilities. The feature is enabled by default.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Check the communication ports for alert deduplication

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts deduplication check_ports [true false]</code>
<b>Description</b>	This option lets you choose whether or not the ports must be taken into account when deduplicating alerts. Setting the value to <code>true</code> (default), two alerts must have the same source and destination ports for them to be considered duplicate and be deduplicated into a single alert (more selective deduplication effect). Setting the value to <code>false</code> , the user obtains a potentially broader deduplication effect.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Size of the buffer for alert deduplication

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts deduplication buffer_size &lt;num&gt;</code>
<b>Description</b>	By default, 2000 different alerts with their corresponding counter are stored in memory and written to the DB at a regular interval. This number can be tweaked depending on the appliance memory limits and on the number of different alerts that need to be deduplicated in the environment. The default should be generous enough for all production environments.
<b>Parameters</b>	<code>num</code> : Size of the deduplication buffer

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Update interval for alert deduplication

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts deduplication update_interval &lt;num&gt;</code>
<b>Description</b>	By default, the counter for every deduplicated alert is written to the database every 5 minutes (300 seconds). The deduplication counter of a single alert is also updated on the database when the maximum size of the duplication buffer is reached; the corresponding alert goes out of scope and is removed from the buffer, but the counter is updated to persist the information. Afterwards, if a similar new alert is raised, the counter for the previous alert will not be updated but the creation of a new alert will be triggered.
<b>Parameters</b>	num: Update interval for the alert deduplication in seconds
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable alert deduplication for specific alert types

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts deduplication disabled_types &lt;arr&gt;</code>
<b>Description</b>	By default, alert deduplication is on for all alert types. However, the feature can be disabled for some alert types. For example: <code>configure alerts deduplication disabled_types ["SIGN:MULTIPLE-UNSUCCESSFUL-LOGINS", "SIGN:MULTIPLE-ACCESS-DENIED"]</code>
<b>Parameters</b>	arr: Array of disabled alert types for the alert deduplication feature
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure maximum number of victims

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts max_victims &lt;num&gt;</code>



<b>Description</b>	Define the maximum number of victims that each alert can contain. Victims exceeding the given value are not stored. Default value is 10.
<b>Parameters</b>	num: Maximum number of victims stored for each alert
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure maximum number of attackers

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts max_attackers &lt;num&gt;</code>
<b>Description</b>	Define the maximum number of attackers that each alert can contain. Attackers exceeding the given value are not stored. Default value is 10.
<b>Parameters</b>	num: Maximum number of attackers stored for each alert
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Show/hide credentials

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts hide_username_on_alerts [true false]</code>
<b>Syntax</b>	<code>conf.user configure alerts hide_password_on_alerts [true false]</code>
<b>Description</b>	This flags determine whether usernames or passwords should be presented in the alert. By default, the credentials are visible. Affected alert types: SIGN: MULTIPLE-ACCESS-DENIED, SIGN: MULTIPLE-UNSUCCESSFUL-LOGINS, SIGN: PASSWORD: WEAK.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure maximum length of description

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts max_description_length &lt;nchars&gt;</code>

<b>Description</b>	Define the maximum number of characters that can contain the description of each incident. When an incident is appending an alert description, the append is performed only if the incident description length is smaller than the limit.
<b>Parameters</b>	nchars: Maximum number of characters allowed in the description of each incident
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure MITRE ATT&CK mapping rules

<b>Product</b>	Guardian
<b>Syntax for MITRE ATT&amp;CK for ICS mappings</b>	<code>conf.user configure alerts mitre_attack ics_mapping &lt;path&gt;</code>
<b>Syntax for MITRE ATT&amp;CK Enterprise mappings</b>	<code>conf.user configure alerts mitre_attack enterprise_mapping &lt;path&gt;</code>
<b>Description</b>	Customize the rules used to assign MITRE ATT&CK techniques to alerts by means of an external file. The file has the following format. Each line defines a rule; the rule specifies an alert type ID followed by a semicolon and a comma-separated list of MITRE ATT&CK technique IDs. For instance, the line <code>SIGN:PROGRAM:TRANSFER;T0843,T0853</code> instructs Guardian that alerts of type <code>SIGN:PROGRAM:TRANSFER</code> must be assigned both the <code>T0843</code> and <code>T0853</code> MITRE ATT&CK techniques.
<b>Parameters</b>	path: The path to the file containing the rules
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable storing of alerts not visible under the current security profiles

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts save_invisible_alerts [true false]</code>

<b>Description</b>	Alerts are not stored into the database, if they are not visible under the current security profile and they are not part of an incident. This command can change this behavior and allow the above alerts to be stored.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osalert stop</code>

### Configure red assertions behavior

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure assertion_element_monitoring [enabled disabled]</code>
<b>Description</b>	By default, an assertion is boolean; it is either red or green, and upon turning red the assertion may send alerts. This configuration allows the assertions to actively monitor lists of elements (such as nodes, labels, etc.), and send new alerts for any additional elements that break the assertion, and when an alert is closed and the triggering asserted element is still present. Disabled by default.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure packet rules alerts limit

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure packet_rules alerts_per_session [limit]</code>
<b>Description</b>	Alerts risen by packet rules have a limit per alert class and per session This limit, which is 1 by default, can be relaxed or removed entirely. A value of -1 means unlimited. A value of 0 will completely disable these alerts.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure how Threat Intelligence contents are handled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts contents &lt;json_value&gt;</code>

<b>Description</b>	<p>This command configures how Threat Intelligence contents are to be loaded. The JSON object can have the following attributes:</p> <ul style="list-style-type: none"> <li>• <code>load_contents</code> - this can be true/false to enable/disable the loading of contents;</li> <li>• <code>stix_backend_provider</code> - selects the engine to be used for STIX indicators: 'memory' (indicators are stored in memory - default) or 'db' (indicators are stored in the system database). With 'db', the application memory requirements become lower, but STIXv1 (XML) indicators are not supported;</li> <li>• <code>loaded_content_types</code> - this is a JSON array of contents to be loaded.</li> </ul> <p>Contents available are:</p> <ul style="list-style-type: none"> <li>• <code>stix_indicators</code></li> </ul> <p>As an example, the following command will disable completely contents loading:</p> <pre>conf.user configure alerts contents { "load_contents": false }</pre> <p>As a further example, the following command will allow only stix indicators rules to be loaded:</p> <pre>conf.user configure alerts contents { "loaded_content_types": [ "stix_indicators" ] }</pre> <p>As another example, the following command will configure the usage of the system database for storing STIX indicators related information, reducing the application memory footprint:</p> <pre>conf.user configure alerts contents { "stix_backend_provider": "db" }</pre>
<b>Parameters</b>	<p><code>json_value</code>: A JSON object to configure how Threat Intelligence contents are loaded</p>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osalert stop</code>

## SIGN:MULTIPLE-ACCESS-DENIED

In this section we will configure the Multiple Access Denied alert.

The detection is enabled by default and works accordingly to the following parameters.

### Set interval and threshold - 1

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi multiple_events protocol &lt;protocol&gt; &lt;interval&gt; &lt;threshold&gt;</code>
<b>Description</b>	Set the detection configuration for a specific protocol.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <b>protocol</b>: Name of the protocol to configure. Can be 'all' to apply the configuration globally.</li> <li>• <b>interval</b>: maximum time in seconds for the event to happen in order to trigger the detection. Default: 30[s] for OT devices, 15[s] for the rest."</li> <li>• <b>threshold</b>: number of times for the event to happen in order to trigger the detection. Default: 20 for OT devices, 40 for the rest.</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the detection of a multiple access denied alert for the SMB protocol with an interval of 10 seconds and threshold of 35 attempts with the following command:

```
conf.user configure vi multiple_events protocol smb 10 35
```

## SIGN:MULTIPLE-UNSUCCESSFUL-LOGINS

In this section we will configure the Multiple Unsuccessful Logins alert.

The detection is enabled by default and works accordingly to the following parameters.

### Set interval and threshold - 2

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi multiple_events protocol &lt;protocol&gt; &lt;interval&gt; &lt;threshold&gt;</code>
<b>Description</b>	Set the detection configuration for a specific protocol.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <b>protocol</b>: Name of the protocol to configure. Can be 'all' to apply the configuration globally.</li> <li>• <b>interval</b>: maximum time in seconds for the event to happen in order to trigger the detection. Default: 30[s] for OT devices, 15[s] for the rest."</li> <li>• <b>threshold</b>: number of times for the event to happen in order to trigger the detection. Default: 20 for OT devices, 40 for the rest.</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the detection of a multiple unsuccessful login alert for the SMB protocol with an interval of 10 seconds and threshold of 35 attempts with the following command:

```
conf.user configure vi multiple_events protocol smb 10 35
```

## SIGN:OUTBOUND-CONNECTIONS

In this section we will configure the outbound connections limit.

Guardian can detect a sudden increase of outbound connections from a specific learned source node. An alert is raised by default when 100 new outbound connections are observed over a 60-seconds interval.

By default, the detection is only performed when the node is being protected. Optionally, the detection can also be performed when the node is being learned.

Optionally, we can prevent the system from creating additional destination nodes in order to preserve resources. Such nodes creation limit is disabled by default.

Some of the configuration parameters listed below can be applied either globally or to individual nodes. The configuration of an individual node has higher priority and overrides the global configuration.

### Perform detection when source node is being learned

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi outbound_connections_limit learning [true false]</code>
<b>Description</b>	Specify whether the detection has to be performed also when the source node is being learned or only when it is being protected. Select <code>true</code> for detection also when the source node is learned, or <code>false</code> for detection only when the source node is being protected. By default <code>false</code> .
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable/disable nodes creation limit

<b>Product</b>	Guardian
<b>Syntax global</b>	<code>conf.user configure vi outbound_connections_limit enabled [true false]</code>
<b>Syntax individual node</b>	<code>conf.user configure vi node &lt;ip&gt; outbound_connections_limit enabled [true false]</code>
<b>Description</b>	Enable (option <code>true</code> ) or disable (option <code>false</code> ) the destination nodes creation limit.
<b>Parameters</b>	<code>ip</code> : The IP of the source node
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Set connections count

<b>Product</b>	Guardian
<b>Syntax global</b>	<code>conf.user configure vi outbound_connections_limit connections &lt;count&gt;</code>
<b>Syntax individual node</b>	<code>conf.user configure vi node &lt;ip&gt; outbound_connections_limit connections &lt;count&gt;</code>
<b>Description</b>	Set the outbound connections limit, in number of connections.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip</code>: The IP of the source node</li> <li>• <code>count</code>: The amount of outbound connections from a node to be observed in order to trigger the detection (default: 100)</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set observation interval

<b>Product</b>	Guardian
<b>Syntax global</b>	<code>conf.user configure vi outbound_connections_limit interval &lt;value&gt;</code>
<b>Syntax individual node</b>	<code>conf.user configure vi node &lt;ip&gt; outbound_connections_limit interval &lt;value&gt;</code>
<b>Description</b>	Set the outbound connections observation interval, in seconds.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip</code>: The IP of the source node</li> <li>• <code>value</code>: The time interval during which the new outbound connections are observed.</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



For example, we can configure the outbound connections limit to prevent a source node from creating additional destination nodes when 70 outbound connections are observed during a 30-seconds interval with the following configuration commands:

```
conf.user configure vi outbound_connections_limit enabled true
conf.user configure vi outbound_connections_limit connections 70
conf.user configure vi outbound_connections_limit interval 30
```

## SIGN:TCP-SYN-FLOOD

In this section we will configure the TCP SYN flood detection.

A node is considered to be under a TCP SYN flood attack when:

- The number of incoming connection attempts during the observation interval is greater than the detection counter
- And, during the observation interval, the ratio between established connections and total number of connection attempts falls below the trigger threshold

A TCP SYN flood attack is considered terminated when:

- The number of incoming connection attempts during the observation interval returns below the detection counter
- Or, during the observation interval, the ratio between established connections and total number of connection attempts returns above the exit threshold

The detection of flooding is not guarded by the duplication detection. In other words, duplicated packets can still trigger a flooding alert. This is because the detection of duplication is based on SYN numbers, which do not change during a flooding event; deduplicating these packets will cause false negatives as it will be inhibiting the flooding detection on duplicate packets.

### Set detection counter

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi tcp_syn_flood_detection counter &lt;value&gt;</code>
<b>Description</b>	Set the connection attempts counter, in number of connections.
<b>Parameters</b>	<code>value</code> : The amount of connection attempts to be observed in order to trigger the detection (default: 100)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set observation interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi tcp_syn_flood_detection interval &lt;value&gt;</code>
<b>Description</b>	Set the observation interval, in seconds.
<b>Parameters</b>	<code>value</code> : The time interval during which the connection attempts are observed, in seconds (default: 10).
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Set trigger threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi tcp_syn_flood_detection trigger_threshold &lt;value&gt;</code>
<b>Description</b>	Set the trigger threshold.
<b>Parameters</b>	<code>value</code> : The ratio between established connections and connections attempts, which when it is reached triggers the flood detection (default: 0.1).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set exit threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi tcp_syn_flood_detection exit_threshold &lt;value&gt;</code>
<b>Description</b>	Set the exit threshold.
<b>Parameters</b>	<code>value</code> : The ratio between established connections and connections attempts, which when it is reached terminates the flood detection (default: 0.4).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, with the commands below the TCP SYN flood detection would trigger when 200 connection attempts are observed during a 15-second observation interval and the ratio between established connections and connection attempts falls below 0.3. Then the detection would terminate when the ratio returns above 0.5.

```
conf.user configure vi tcp_syn_flood_detection counter 200
conf.user configure vi tcp_syn_flood_detection interval 15
conf.user configure vi tcp_syn_flood_detection trigger_threshold 0.3
conf.user configure vi tcp_syn_flood_detection exit_threshold 0.5
```

## SIGN:UDP-FLOOD

In this section we will configure the UDP flood detection.

The detection is enabled by default and it triggers when a victim receives 20'000 UDP packets per second for at least 10 seconds.

### Enable/disable detection

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi udp_flood_detection enabled [true false]</code>
<b>Description</b>	Enable (option <code>true</code> ) or disable (option <code>false</code> ) the UDP flood detection.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set detection threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi udp_flood_detection packets_per_second &lt;threshold&gt;</code>
<b>Description</b>	Set the UDP flood detection threshold, in packets per second.
<b>Parameters</b>	<code>threshold</code> : The amount of UDP packets per second to be transmitted to a victim for at least 10 seconds in order to trigger the detection (default: 20000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the UDP flood detection to trigger when a victim receives 40'000 UDP packets per second for at least 10 seconds with the following configuration command:

```
vi udp_flood_detection packets_per_second 40000
```

## SIGN:NETWORK-SCAN

### DDOS Defense

In this section we will configure the detection of a DDOS attack.

The detection is enabled by default and an alert is raised at most every 5 minutes, when under one minute more than 20 nodes have been created.

#### Set analysis interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ddos_defense interval &lt;threshold&gt;</code>
<b>Description</b>	Set the analysis interval for the detection.
<b>Parameters</b>	<code>threshold</code> : The analysis interval is measured in minutes. Default: one minute.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

#### Set max created nodes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ddos_defense max_created_nodes &lt;max_nodes&gt;</code>
<b>Description</b>	Number of created nodes that, if created in less time than the analysis interval, will trigger the alert.
<b>Parameters</b>	<code>max_nodes</code> : Number of created nodes that trigger the detection. Default: 20.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

#### Set alert threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ddos_defense alert_threshold &lt;threshold&gt;</code>
<b>Description</b>	Interval to wait in order to raise an additional alert.

<b>Parameters</b>	<code>threshold</code> : Minutes to wait for another alert to be raised. Default: 5 minutes.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set alert threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ddos_defense alert_threshold &lt;threshold&gt;</code>
<b>Description</b>	Interval to wait in order to raise an additional alert.
<b>Parameters</b>	<code>threshold</code> : Minutes to wait for another alert to be raised. Default: 5 minutes.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### TCP Port Scan

In this section we will configure the detection for the TCP Port scan.

The detection is enabled by default and an alert is emitted according to the configuration parameters described below.

### Set attempts threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp attempts_threshold &lt;threshold&gt;</code>
<b>Description</b>	Set the number of scan attempts that will trigger the alert.
<b>Parameters</b>	<code>threshold</code> : Number of scan attempts that will trigger the alert. Default: 100.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set observation interval

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp interval &lt;interval&gt;</code>
<b>Description</b>	Set the analysis interval for the detection algorithm.
<b>Parameters</b>	<code>interval</code> : Analysis interval in seconds for the detection algorithm. Default: 10 seconds.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set trigger threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp trigger_threshold &lt;threshold&gt;</code>
<b>Description</b>	Set the trigger threshold for the detection algorithm. An alert is raised only if the ratio between the number of established connections and total attempts is smaller than the trigger threshold.
<b>Parameters</b>	<code>threshold</code> : Trigger threshold as described above for the detection algorithm. Default: 0.1.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set out of sequence threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp out_of_sequence_threshold_number &lt;threshold&gt;</code>
<b>Description</b>	Set the number of out of sync fragments which trigger this feature of the detection algorithm.
<b>Parameters</b>	<code>threshold</code> : Number of out of sync fragments. Default: 10.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set out of sequence interval

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp out_of_sequence_interval &lt;interval&gt;</code>
<b>Description</b>	Set the analysis interval of the out of sync recognition feature of the detection algorithm.
<b>Parameters</b>	<code>interval</code> : Analysis interval in seconds. Default: 10 seconds.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set out of sequence max rate

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp out_of_sequence_threshold_max_rate &lt;rate&gt;</code>
<b>Description</b>	Set the period of time during which additional alerts due to out of sync fragments are not raised.
<b>Parameters</b>	<code>rate</code> : Timespan in minutes to mute additional alerts due to out of sync fragments. Default: 5 minutes.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set ignored port ranges

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_tcp ignore_ports &lt;port_ranges&gt;[ , &lt;port_ranges&gt; ]</code>
<b>Description</b>	Set the victims' ports or port ranges which must not participate in the detection algorithm.
<b>Parameters</b>	<code>port_ranges</code> : ports can be entered as a list of comma separated values and ranges as a pair of ports separated by a dash. Example: 1000,1200-1300,1500. Default: none.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



For example, we can configure the detection for the TCP Port scan with the following commands:

```
conf.user configure vi port_scan_tcp attempts_threshold 50
conf.user configure vi port_scan_tcp interval 20
conf.user configure vi port_scan_tcp trigger_threshold 0.2
conf.user configure vi port_scan_tcp
out_of_sequence_threshold_number 15
conf.user configure vi port_scan_tcp out_of_sequence_interval 20
conf.user configure vi port_scan_tcp
out_of_sequence_threshold_max_rate 10
conf.user configure vi port_scan_tcp ignore_ports
1000,1200-1300,1500
```

### UDP Port Scan

In this section we will configure the detection for the UDP Port scan.

The detection is enabled by default and an alert is emitted according to the configuration parameters described below.

#### Set fast threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_udp fast_threshold &lt;threshold&gt;</code>
<b>Description</b>	Set the number of attempts which will trigger the alert for the fast detection algorithm.
<b>Parameters</b>	threshold: Attempts triggering the alert for the fast detection algorithm. Default: 500.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

#### Set slow interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_udp slow_interval &lt;interval&gt;</code>
<b>Description</b>	Set the analysis interval for the slow detection algorithm.
<b>Parameters</b>	interval: Analysis interval for the slow detection algorithm. Default: 60 seconds.

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set fast interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_udp fast_interval &lt;interval&gt;</code>
<b>Description</b>	Set the analysis interval for the fast detection algorithm.
<b>Parameters</b>	interval: Analysis interval for the fast detection algorithm. Default: 1 second.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set fast different ports threshold

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_udp fast_different_ports_threshold &lt;threshold&gt;</code>
<b>Description</b>	Set the number of different ports that should be tested by the attacker for the fast detection algorithm to trigger the alert.
<b>Parameters</b>	threshold: Minimum number of different ports to be tested by the attacker to trigger the alert for the fast detection algorithm. Default: 250.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set unreachable ratio

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi port_scan_udp unreachable_ratio &lt;ratio&gt;</code>
<b>Description</b>	The slow detection algorithm will issue an alert only if the ratio between the number of unreachable requests and the total requests is greater than this value.

<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>ratio</code>: Critical ratio for the slow detection algorithm to trigger an alert. An alert is raised if the ratio between the number of unreachable requests and the total requests is greater than the critical ratio. Default: 0.1.</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the detection for the UDP Port scan with the following commands:

```
conf.user configure vi port_scan_udp slow_threshold 200
conf.user configure vi port_scan_udp slow_interval 30
conf.user configure vi port_scan_udp fast_threshold 400
conf.user configure vi port_scan_udp
fast_different_ports_threshold 150
conf.user configure vi port_scan_udp fast_interval 3
conf.user configure vi port_scan_udp unreachable_ratio 0.2
```

## Ping Sweep

In this section we will configure the detection for the ICMP/Ping Sweep scan.

The detection is enabled by default and an alert is emitted when more than 100 request are issued in less than 5 seconds with a total number of recorded victims equal to 100.

### Set request number

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ping_sweep max_requests &lt;threshold&gt;</code>
<b>Description</b>	Set the number of requests that will trigger the alert.
<b>Parameters</b>	<code>threshold</code> : Number of request that will raise the alert. Default: 100.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi ping_sweep interval &lt;interval&gt;</code>

<b>Description</b>	Set the interval during which the maximum number of requests should be issued in order to trigger the alert.
<b>Parameters</b>	<code>interval</code> : Interval in seconds for the maximum requests to be issued. Default: 5 seconds.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the detection for the ICMP/Ping Sweep scan with an analysis interval of 10 seconds for a threshold of 200 requests with 150 victims recorded with the following commands:

```
conf.user configure vi ping_sweep max_requests 200
conf.user configure vi ping_sweep interval 10
```

### Treck Stack

In this section we will configure the detection for the Treck TCP/IP Fingerprint scan via ICMP 165.

The detection is enabled by default and an alert is emitted at most once every 20 minutes.

### Set alert interval

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi treck_stack once_every &lt;threshold&gt;</code>
<b>Description</b>	Set the minimum interval between two raised alerts, in minutes.
<b>Parameters</b>	<code>threshold</code> : Minutes to wait for another alert to be raised. Default: 20 minutes.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

For example, we can configure the detection for the Treck TCP/IP Fingerprint Scan via ICMP 165 with an interval between two emitted alerts of one hour (60 minutes) with the following command:

```
conf.user configure vi treck_stack once_every 60
```

## Chapter 6. Incidents



## INCIDENT:PORT-SCAN

### Port scan incident

In this section we will configure the parameters of a Port Scan Incident.

The detection is enabled by default and an incident is raised when more than 6 correlated alerts are triggered, independently from their creation time.

For example, we can configure the parameters for the Port Scan Incident with the following command, where we identify the minimum number of alerts for the incident to be triggered, and the maximum time interval in milliseconds in which they need to occur:

```
conf.user configure alerts incidents portscan {"min_alerts": 25,
"max_time_interval": 1500}
```

### Configure the port scan incident

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure alerts incidents portscan &lt;json_obj&gt;</code>
<b>Description</b>	Configure the port scan incident by providing the configuration in a JSON object.
<b>Parameters</b>	<code>json_obj</code> : JSON object containing the keys 'min_alerts' and 'max_time_interval', which are respectively the minimum number of alerts which trigger the detection and the maximum time interval in which they need to occur.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osalert stop</code>





# Chapter 7. Nodes



## Configure nodes

### Set node label

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>ids configure vi node &lt;ip&gt; label &lt;label&gt;</code>
<b>Syntax erase</b>	<code>ids configure vi node &lt;ip&gt; label</code>
<b>Description</b>	Set the label to a node, the label will appear in the <b>Graph</b> , in the <b>Nodes</b> , in the <b>Process &gt; Variables</b>
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>ip</code>: The IP address of the node</li> <li><code>label</code>: The label that will be displayed in the user interface</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set default live traffic label formatting

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi default_node_live_label &lt;operation&gt;[:&lt;param&gt;][,&lt;operation&gt;[:&lt;param&gt;]]</code>
<b>Description</b>	The default formatting operation(s) applied to labels coming from live traffic. More operations can be applied sequentially. See also the "Set protocol-specific live traffic label formatting" configuration for details
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>operation</code>: The operation to apply</li> <li><code>param</code>: The specific operation parameter</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set protocol-specific live traffic label formatting

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi node_live_label &lt;protocol&gt; &lt;operation&gt;[:&lt;param&gt;][,&lt;operation&gt;[:&lt;param&gt;]]</code>

<b>Description</b>	Protocol-specific formatting operation(s) applied to labels coming from live traffic. More operations can be applied sequentially.
<b>Parameters</b>	<p><b>protocol</b>: The name of the protocol</p> <p><b>operation</b>: The operation to apply. There are several operations categories:</p> <ul style="list-style-type: none"> <li>• Invalid character replace operations (the default <b>param</b> is ' '): <ul style="list-style-type: none"> <li>◦ <b>utf8</b>: only printable utf8 characters</li> <li>◦ <b>ascii</b>: only printable ascii characters</li> <li>◦ <b>alnum</b>: only alphanumeric characters <b>a-zA-Z0-9</b></li> <li>◦ <b>alnum_underscore</b>: only <b>alnum</b> + underscore</li> </ul> </li> <li>• String operations:<b>prefix</b>: it keeps only the prefix of a string identified by <b>param</b>. The default <b>param</b> is ' . '</li> <li>• Validation operations: <ul style="list-style-type: none"> <li>◦ <b>strict</b>: checks if the label is changed from it's first value or, if present, from the last <b>mark</b> operation. If changed, the label will be set to empty</li> <li>◦ <b>mark</b>: resets the <b>strict</b> history to the current operation (the label will not be changed)</li> </ul> </li> </ul> <p><b>param</b>: The specific operation parameter as above.</p> <p>For an example, see the table below.</p>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

Operations	Input label	Output label	Comment
utf8:-	labçl çtestç.	lab-l -test-.	The <b>utf8</b> operation replaces not allowed characters with the set parameter 'l'
alnum	testl	testl	Unchanged because all characters are valid
alnum	lab,l	lab l	The <b>alnum</b> operation replaces a not allowed character

Operations	Input label	Output label	Comment
alnum,strict	lab,1		The <code>strict</code> operation detects a change between the initial input 'lab,1' and the <code>alnum</code> output 'lab 1', the label is cleared
alnum,mark,utf8,strict	lab,1	lab 1	The <code>mark</code> operation sets <code>alnum</code> output (lab 1) as default, for the following <code>strict</code> operation. The <code>utf8</code> operation has not effect, and <code>strict</code> detects no changes so has no effect either
prefix	host_n.domain1.it	host_n	The <code>prefix</code> operation keeps only the hostname part

### Set node Device ID with priority

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi node &lt;ip&gt; device_id_with_priority &lt;device_id&gt;;&lt;priority&gt;</code>
<b>Description</b>	Adds the Device ID to the set of node Device IDs. The final Device ID, used for node grouping under Assets is the one with the highest priority
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip</code>: The IP address of the node</li> <li>• <code>device_id</code>: The device id</li> <li>• <code>priority</code>: the priority of the Device ID. If missing, it will be set to the lowest priority value</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

## Override node Device ID

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi node &lt;ip&gt; device_id_override &lt;device_id&gt;</code>
<b>Description</b>	Adds the Device ID to the set of node Device IDs, giving it the maximum priority value. This Device ID will be used for node grouping under Assets
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip</code>: The IP address of the node</li> <li>• <code>device_id</code>: The device id (with the maximum priority)</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Enable or disable node

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi node &lt;ip&gt; state [enabled disabled]</code>
<b>Description</b>	This directive permits to disable a node. This setting has effect in the graph: a disabled node will not be displayed.
<b>Parameters</b>	<code>ip</code> : The IP address of the node
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Enable or disable same ip node separation

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure check_multiple_mac_s_same_ip enable [true false]</code>
<b>Description</b>	<p>This directive permits to enable the separation of L3 nodes with same IP but different MAC address. The nodes with the desired IP addresses will be treated as L2 nodes and appear as distinct assets. If the nodes already exist as L3 nodes upon the application of the configuration, they will be deleted and the new logic will start to execute with empty statistics.</p> <p>The values of <code>true</code> or <code>false</code> enables, respectively disables, the feature.</p>

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Configure same ip node separation

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure check_multiple_macs_same_ip ip &lt;ip_address&gt;</code>
<b>Description</b>	Selects the ip of the nodes which should be separated as per the strategy described in the previous box.
<b>Parameters</b>	<code>ip_address</code> : The IP of the node to be configured
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Delete node

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi node &lt;ip&gt; :delete</code>
<b>Description</b>	Delete a node from the environment
<b>Parameters</b>	<code>ip</code> : The IP of the node to delete
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Define a cluster

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi cluster &lt;ip&gt; &lt;name&gt;</code>
<b>Description</b>	This command permits to define an High Availability cluster of observed nodes. In particular, this permits to: accelerate the learning phase by joining the learning data of two sibling nodes, and to group nodes by cluster in the graph.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip</code>: The IP of the node</li> <li>• <code>name</code>: The name of the cluster</li> </ul>

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



## Chapter 8. Assets



## Configure assets

### Hide built-in asset types

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi hide_built_in_asset_types true</code>
<b>Description</b>	Hides built-in asset types visible from the dropdown in the asset configuration modal
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



## Chapter 9. Links



## Configure links

### Set link last activity check

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :check_last_activity &lt;seconds&gt;</code>
<b>Syntax erase</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :check_last_activity :delete</code>
<b>Description</b>	Set the last activity check on a link, an alert will be raised if the link remains inactive for more than the specified seconds
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1, ip2</code>: The IPs of the two nodes involved in the communication</li> <li>• <code>protocol</code>: The protocol</li> <li>• <code>seconds</code>: The communication timeout</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set link persistency check

<b>Product</b>	Guardian
<b>Syntax</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :is_persistent [true false]</code>
<b>Description</b>	Set the persistency check on a link, if a new handshake is detected an alert will be raised
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1, ip2</code>: The IPs of the two nodes involved in the communication</li> <li>• <code>protocol</code>: The protocol</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set link alert on SYN

<b>Product</b>	Guardian
<b>Syntax</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :alert_on_syn [true false]</code>

<b>Description</b>	Raise an alert when a TCP SYN packet is detected on this link
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1</code>, <code>ip2</code>: The IPs of the two nodes involved in the communication</li> <li>• <code>protocol</code>: The protocol</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set link to track availability

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :track_availability &lt;seconds&gt;</code>
<b>Syntax erase</b>	<code>vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :track_availability :delete</code>
<b>Description</b>	Notify the link events when the link communication is interrupted or resumed.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1</code>, <code>ip2</code>: The IPs of the two nodes involved in the communication</li> <li>• <code>protocol</code>: The protocol</li> <li>• <code>seconds</code>: Interval to checking if the link is available or not</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Delete link

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi link &lt;ip1&gt; &lt;ip2&gt; :delete</code>
<b>Description</b>	Delete a link
<b>Parameters</b>	<code>ip1</code> , <code>ip2</code> : The IPs identifying the link
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.



## Delete protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; :delete</code>
<b>Description</b>	Delete a protocol from a link
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1, ip2</code>: The IPs identifying the link</li> <li>• <code>protocol</code>: The protocol of the link to delete</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Learn function code

<b>Product</b>	Guardian
<b>Learn</b>	<code>ids configure vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; fc &lt;func_code&gt;</code>
<b>Learn or Unlearn</b>	<code>ids configure vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; fc &lt;func_code&gt; is_learned [true false]</code>
<b>Description</b>	Learn or unlearn a function code from a protocol
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1, ip2</code>: The IPs identifying the link</li> <li>• <code>protocol</code>: The protocol of the link</li> <li>• <code>func_code</code>: The function code to be learned or unlearned</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

## Delete function code

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi link &lt;ip1&gt; &lt;ip2&gt; &lt;protocol&gt; fc &lt;func_code&gt; :delete</code>
<b>Description</b>	Delete a function code from a protocol

<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>ip1</code>, <code>ip2</code>: The IPs identifying the link</li> <li>• <code>protocol</code>: The protocol of the link</li> <li>• <code>func_code</code>: The function code to be deleted</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Enable `link_events` generation

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi link_events [enabled disabled]</code>
<b>Description</b>	Enable or disable the generation of <code>link_events</code> records, this feature can have an impact on performance, enable it carefully
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Disabling the persistence of links

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi persistence skip_links true</code>
<b>Description</b>	With this configuration rule you can disable the persistence of links, thus saving disk space in cases with a large number of links.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable link ports collection for the specified set of protocols

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi enable_link_ports &lt;protocol_name&gt;[ , &lt;protocol_name&gt;]</code>

<b>Description</b>	For enabled protocols, the set of source and destination ports found in the underlying sessions is collected and shown as links attributes ( <code>from_ports</code> and <code>to_ports</code> ). By default enabled only on unrecognized protocols (i.e. links named <code>other</code> ). If the command is used, the total list of protocols to be enabled shall be specified (and so including <code>other</code> too if applicable).
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>protocol_name</code>: The name of the protocol to enable</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set max number of collected link ports

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi max_link_ports &lt;max_collected_ports&gt;</code>
<b>Description</b>	When the ports collection is enabled for links, sets the maximum number of collected ports for each link.
<b>Parameters</b>	<code>max_collected_ports</code> : The maximum number of collected ports for each link. Default 32. Max 128.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



# Chapter 10. Variables



## Configure variables

### Enable or disable default variable history

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable default history [enabled disabled]</code>
<b>Description</b>	<p>Set if the variable history is enabled or not, when not set it's disabled. The amount of the history maintained can be configured in "Variable history retention" section in <a href="#">Configure retention (on page 139)</a>.</p> <p><b>Note:</b> Enabling this functionality can negatively affect Guardian's performance, depending on the amount of variables and the update rate.</p>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable or disable variable history

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable &lt;var_key&gt; history [enabled disabled]</code>
<b>Description</b>	<p>Define the amount of samples shown in the graphical history of a variable. Set if the variable history is enabled or not, when not set it's disabled. The amount of the history maintained can be configured in "Variable history retention" section in <a href="#">Configure retention (on page 139)</a>.</p> <p><b>Note:</b> Enabling this functionality can negatively affect Guardian's performance, depending on the amount of variables and the update rate.</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>var_key</code>: The variable identifier</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable label

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable &lt;var_key&gt; label &lt;label&gt;</code>
<b>Description</b>	Set the label for a variable, the label will appear in the <code>Process</code> sections

<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>label</code>: The label displayed in the user interface</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable unit of measure

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable &lt;var_key&gt; unit &lt;unit&gt;</code>
<b>Description</b>	Set a unit of measure on a variable.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>unit</code>: The unit of measure displayed in the user interface</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable offset

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable &lt;var_key&gt; offset &lt;offset&gt;</code>
<b>Description</b>	The offset of the variable that will be used to map the 0 value of the variable.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>offset</code>: The offset value used to calculate the final value of the variable</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.



### Set variable scale

<b>Product</b>	Guardian
<b>Syntax</b>	<code>ids configure vi variable &lt;var_key&gt; scale &lt;scale&gt;</code>
<b>Description</b>	The scale of the variable that is used to define the full range of the variable.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>scale</code>: the scale value used to calculate the final value of the variable</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable last update check

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>vi variable &lt;var_key&gt; :check_last_update &lt;seconds&gt;</code>
<b>Syntax remove</b>	<code>vi variable &lt;var_key&gt; :check_last_update :delete</code>
<b>Description</b>	Set the last update check on a variable, if the variable value is not updated for more than the specified seconds an alert is raised
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>seconds</code>: The timeout after which a stale variable alert will be raised</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable quality check

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>vi variable &lt;var_key&gt; :check_quality &lt;seconds&gt;</code>
<b>Syntax remove</b>	<code>vi variable &lt;var_key&gt; :check_quality :delete</code>
<b>Description</b>	Set the quality check on a variable, if the value quality remains invalid for more than the specified seconds an alert is raised

<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>seconds</code>: The maximum amount of consecutive seconds the variable can have an invalid quality</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set variable to alert on quality

<b>Product</b>	Guardian
<b>Syntax set</b>	<code>vi variable &lt;var_key&gt; :alert_on_quality &lt;quality&gt;</code>
<b>Syntax remove</b>	<code>vi variable &lt;var_key&gt; :alert_on_quality :delete</code>
<b>Description</b>	Raise an alert when the variable has one of the specified qualities. Possible values are: invalid, not topical, blocked, substituted, overflow, reserved, questionable, out of range, bad reference, oscillatory, failure, inconsistent, inaccurate, test, alarm. Multiple values can be separated by comma.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>quality</code>: The alert quality</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set a variable critical state

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure cs variable &lt;id&gt; &lt;var_key&gt; [&lt; &gt; =] &lt;value&gt;</code>
<b>Description</b>	<p>Define a new custom critical state on a single variable that will raise on violation of defined range.</p> <p>For instance, if the &gt; operator is specified, the variable will have to be higher than value to trigger the critical state.</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>id</code>: A unique ID for this critical state</li> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>value</code>: The variable value to check for</li> </ul>

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set a multiple critical state

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure cs multi &lt;id&gt; variable &lt;ci&gt; &lt;var_key&gt; [&lt; &gt; =] &lt;value&gt;[ ^ variable &lt;ci&gt; &lt;var_key&gt; [&lt; &gt; =] &lt;value&gt;]</code>
<b>Description</b>	Creates a multi-valued critical state, that is an expression of "variable critical states", described above. The syntax is and AND (^) expression of the single-variable critical state.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>id</code>: A unique ID for this critical state</li> <li>• <code>ci</code>: Enumerate the variables <code>c1</code>, <code>c2</code>, <code>c3</code>, ..., etc</li> <li>• <code>var_key</code>: The variable identifier</li> <li>• <code>value</code>: The variable value to check for</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Control variables extraction at the protocol level

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol &lt;name&gt; variables_extraction [disabled enabled advanced global]</code>
<b>Description</b>	<p>It allows the application of a variable extraction policy different from the global policy on a protocol basis. Note that if the Global policy is set to Disabled, it prevails on any protocol-specific setting. However, protocol specific policies prevail.</p> <p>Choices are whether variables extraction is <code>disabled</code>, <code>enabled</code>, enabled with advanced heuristics (<code>advanced</code>) or if it should inherit the global policy (<code>global</code>)</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>name</code>: The name of the target protocol</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Control variables extraction at the global level for all zones

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi variables_extraction [disabled enabled advanced global]</code>
<b>Description</b>	<p>Same as for the protocol level variables extraction, except it sets the policy for the global level.</p> <p>Choices are whether variables extraction is <code>disabled</code>, <code>enabled</code>, enabled with advanced heuristics (<code>advanced</code>) or if it should inherit the global policy (<code>global</code>)</p>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Control variables extraction at the global level for specific zones

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure vi variables_extraction [disabled enabled advanced global] &lt;zones&gt;</code>
<b>Description</b>	<p>Same as for the protocol level variables extraction, except it sets the policy for the global level for the specified zones.</p> <p>Choices are whether variables extraction is <code>disabled</code>, <code>enabled</code>, enabled with advanced heuristics (<code>advanced</code>) or if it should inherit the global policy (<code>global</code>)</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>zones</code>: Name of the zones for which the extraction should be enabled. If unspecified, the extraction is enabled for all the zones. and values are separated by a comma; for example: <code>[plant1,plant2]</code> or <code>[zone1,zone2,zone3]</code>. Brackets are required</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

# Chapter 11. Protocols



## Configure protocols

### Configure iec104s encryption key

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec104s tls private_key &lt;ip&gt; &lt;location&gt;</code>
<b>Description</b>	Add a private key associated to the device running iec104s. For more information, see <a href="#">Configure IEC-62351-3 (on page 122)</a> .
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>ip</code>: The IP of the device</li> <li><code>location</code>: The absolute location of the key</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set variable namespace for iec101 protocol decoder

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec101 var_namespace &lt;namespace&gt;</code>
<b>Description</b>	iec101 CA size can vary across implementations, with this configuration rule the user can customize the setting for its own environment
<b>Parameters</b>	<code>size</code> : Namespace to be used in variable definition. It can be <code>la</code> or <code>ca</code> (default: <code>ca</code> )
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set CA size for iec101 protocol decoder

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec101 ca_size &lt;size&gt;</code>
<b>Description</b>	iec101 CA size can vary across implementations, with this configuration rule the user can customize the setting for its own environment
<b>Parameters</b>	<code>size</code> : The size in bytes of the CA. Accepted values: 1,2 (default: 1)

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set LA size for iec101 protocol decoder

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec101 la_size &lt;size&gt;</code>
<b>Description</b>	iec101 LA size can vary across implementations, with this configuration rule the user can customize the setting for its own environment
<b>Parameters</b>	<code>size</code> : The size in bytes of the LA. Accepted values: 1,2 (default: 1)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set IOA size for iec101 protocol decoder

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec101 ioa_size &lt;size&gt;</code>
<b>Description</b>	iec101 IOA size can vary across implementations, with this configuration rule the user can customize the setting for its own environment
<b>Parameters</b>	<code>size</code> : The size in bytes of the IOA. Accepted values: 1,2,3 (default: 2)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osids stop</code>

### Set a dictionary file

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol &lt;protocol&gt; dictionary &lt;dictionary_file_name&gt;</code>
<b>Description</b>	Based on the dictionary file set with this command, friendly names are associated to the extracted variables, for the specific protocol in scope.
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>protocol</code>: The protocol can be can-bus or mvb</li> <li><code>dictionary_file_name</code>: The path for the dictionary file</li> </ul>
<b>Where</b>	<b>CLI</b>



<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Set an arbitrary amount of bytes to skip before decoding iec101 protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec101 bytes_to_skip &lt;amount&gt;</code>
<b>Description</b>	Based on the hardware configuration iec101 can be prefixed with a fixed amount of bytes, with this setting Guardian can be adapted to the peculiarity of the environment.
<b>Parameters</b>	amount: The amount of bytes to skip
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable the Red Electrica Espanola semantic for iec102 protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec102 ree [enabled disabled]</code>
<b>Description</b>	There is a standard from Red Electrica Española which changes the semantic of the iec102 protocol, after enabling (choosing option enabled) this setting the iec102 protocol decoder will be compliant to the REE standard.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the subnet in which the iec102 protocol will be enabled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec102 subnet &lt;subnet&gt;</code>
<b>Description</b>	The detection of iec102 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific subnet
<b>Parameters</b>	subnet: A subnet in the CIDR notation
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Enable iec102 on the specified port

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec102 port &lt;port&gt;</code>
<b>Description</b>	The detection of iec102 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific port
<b>Parameters</b>	port: The TCP port
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the subnet in which the iec103 protocol will be enabled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec103 subnet &lt;subnet&gt;</code>
<b>Description</b>	The detection of iec103 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific subnet
<b>Parameters</b>	subnet: A subnet in the CIDR notation
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable iec103 on the specified port

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec103 port &lt;port&gt;</code>
<b>Description</b>	The detection of iec103 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific port
<b>Parameters</b>	port: The TCP port
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Force iec101 semantics inside iec103 protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec103 force_iec101_semantics true</code>
<b>Description</b>	Forces change of semantics for iec103 protocol to use ASDUs of iec101
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Allow to recognize as iec103 very fragmented sessions

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol iec103 accept_on_fragmented true</code>
<b>Description</b>	Allow to accept as iec103 those packets that are always incomplete, thus allowing situations where the protocol is heavily fragmented to be recognized.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable the detection of plain text passwords in HTTP payloads

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol http detect_uri_passwords [true false]</code>
<b>Description</b>	Guardian is able to detect if plain text passwords and login credentials are present in HTTP payloads, such as strings containing <code>ftp://user:password@example.com</code> . The feature is disabled by default. Choose <code>true</code> to enable the feature and <code>false</code> to disable it.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the subnet in which the tg102 protocol will be enabled

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>conf.user configure probe protocol tg102 subnet &lt;subnet&gt;</code>
<b>Description</b>	The detection of tg102 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific subnet
<b>Parameters</b>	subnet: A subnet in the CIDR notation
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the port range in which the tg102 protocol will be enabled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol tg102 port_range &lt;src_port&gt;-&lt;dst_port&gt;</code>
<b>Description</b>	The detection of tg102 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific port range
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>src_port</code>: The starting port of the range</li> <li>• <code>dst_port</code>: The ending port of the range</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the subnet in which the tg800 protocol will be enabled

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol tg800 subnet &lt;subnet&gt;</code>
<b>Description</b>	The detection of tg800 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific subnet
<b>Parameters</b>	subnet: A subnet in the CIDR notation
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the port range in which the tg800 protocol will be enabled

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>conf.user configure probe protocol tg800 port_range &lt;src_port&gt;-&lt;dst_port&gt;</code>
<b>Description</b>	The detection of tg800 can lead to false positives, this rules give the possibility to the user to enable the detection on a specific port range
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>src_port</code>: The starting port of the range</li> <li>• <code>dst_port</code>: The ending port of the range</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable variable extraction for a Siemens S7 area and type

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol s7 exclude &lt;area&gt; &lt;type&gt;</code>
<b>Description</b>	For performance reasons or to reduce noise it's possible to selectively exclude variables extraction for some areas and type.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>area</code>: The area, some examples are: DB, DI, M, Q</li> <li>• <code>type</code>: The type of the <i>Variable</i>, some examples are: INT, REAL, BYTE</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable the full TLS inspection mode

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe tls-inspection enable [true false]</code>
<b>Description</b>	<p>TLS inspection is normally performed only on https and iec104s traffic. Enabling (choosing the option <code>true</code>) the full inspection mode provides the following additional features:</p> <ul style="list-style-type: none"> <li>• TLS traffic found on any TCP port is inspected</li> <li>• an alert is raised when TLS-1.0 is used (when this mode is disabled, this is an https only check)</li> <li>• an alert is raised on expired certificates</li> <li>• an alert is raised on weak cipher suites</li> <li>• session ID, cipher suite and certificates are extracted into the relative link events</li> </ul>
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Enable or disable the persistence of the connections for Ethernet/IP Implicit

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol ethernetip-implicit persist-connection [true false]</code>
<b>Description</b>	The Ethernet/IP Implicit decoder of Guardian is able to detect handshakes that are then used to decode variables. In some scenarios these handshakes are not common but it's very important to persist them so that Guardian can continue to decode variables after a reboot or an upgrade. By enabling (choosing option true) this option Guardian will store on disk the data needed to autonomously reproduce the handshake phase after a reboot.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable or disable fragmented packets for modbus protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol modbus enable_full_fragmentation [true false]</code>
<b>Description</b>	Modbus protocol is usually not fragmented, so this option is by default disabled (option false). If fragmented modbus packets can be present in the network, then full fragmentation can be enabled (choosing option true) to avoid generation of unexpected alerts.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable or disable the support for LACBUS-RTU

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol modbus lacbus-rtu-protocol [true false]</code>

<b>Description</b>	LACBUS-RTU is a protocol that extends Modbus. By default, this protocol is not parsed by Guardian. It can be enabled using this configuration, in which case LACBUS-RTU transmissions are still tagged as 'modbus', but the function codes specific to LACBUS-RTU will be processed, and the corresponding variables extracted.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Import the ge-egd produced data XML file for variables extraction

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol ge-egd produced-data-xml &lt;path&gt;</code>
<b>Description</b>	The ge-egd protocol can extract process variables only after the XML file describing the produced data for the involved nodes is imported. Multiple imports are allowed as long as the XML files do not provide overlapping information for any producer node.
<b>Parameters</b>	<code>path</code> : The path of the produced data XML file to import
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable file extraction for SMB protocol

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol smb file_extraction false</code>
<b>Description</b>	The SMB protocol decoder is able to extract files and analyze them for malware in a sandbox. If not needed, the user can disable such feature and improve the performance of the system especially in environments where SMB file transfer is heavily used.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Activate the extraction of GE asset information from modbus registers

<b>Product</b>	Guardian
----------------	----------

<b>Syntax</b>	<code>conf.user configure probe protocol modbus ge_asset_info_from_registers true</code>
<b>Description</b>	Some General Electric devices send asset information (product name, firmware version, serial number, label, and FPGA version) encoded in register values with the Modbus protocol. By enabling this setting, Guardian is instructed to extract this data and enrich the corresponding nodes with it. This data is also used to produce CPEs for the corresponding devices.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable byte scan of traffic by unrecognized protocols in order to detect executables

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe enable-executable-extraction [true false]</code>
<b>Description</b>	When this option is enabled, the traffic from unrecognized protocols is scanned in order to detect the presence of executables. If an executable is found, it is extracted and sent to Sandbox for analysis. This option is enabled by default. However, when a lot of traffic is flowing through unknown protocols, disabling it can improve performance.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Configure ARP flood

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe protocol arp flood &lt;json_value&gt;</code>



<p><b>Description</b></p>	<p>A json object to configure ARP flood</p> <ul style="list-style-type: none"> <li>• <b>enable</b> - Enable or disable flood detection. Possible values <code>true</code> or <code>false</code> by default is <code>false</code>.</li> <li>• <b>packet_limit</b> - Define the number of packets per seconds (n) that trigger a flood detection (default = 1000).</li> <li>• <b>packet_increase_limit</b> - Define the increase in the number of packets per seconds that trigger a flood detection (default = 1000). The difference between <code>packet_limit</code> and <code>packet_increase_limit</code> is that the first one checks the number of packets per second while the second checks the increase in the number of packets per second. For example if <code>packet_limit</code> = 3000 and <code>packet_increase_limit</code> = 500 if the arp packets increase by 100 each second no alert is triggered until they reach the threshold of 3000 packets per second, but if they increase in one second of 600, for example they pass from 1000 to 1600, then the flood is triggered.</li> <li>• <b>packet_ratio_limit</b> - Define the ratio between the number of arp and total packets that trigger the flood . The flood is triggered when the ratio is greater than the specified value. In order to enable this detection method there is the need to enable also the counting of the total packets that can be done with <code>probe packet-monitor enable true</code>. A value of 0.1 means that the flood is triggered when the number of arp packets is 10% of the total packets. Default value is 1.0 (all packets are arp) in such a way that this threshold is disabled by default.</li> <li>• <b>strict_mode</b> - When <code>true</code> all the above checks have to be verified at the same time to trigger the detection, while when <code>strict_mode</code> is <code>false</code> it is enough for a single check to trigger the flood. Possible values <code>true</code> or <code>false</code> by default is <code>false</code>.</li> <li>• <b>repeat_mode</b> - Define how subsequent triggers are handled. Default value is <code>mute_for_period</code>. Possible values are: <ul style="list-style-type: none"> <li>◦ <code>repeat</code>: alerts are raised at each detection</li> <li>◦ <code>mute_for_period</code>: after the first detection alerts are muted for a specified period</li> <li>◦ <code>mute_until_passes</code> no subsequent alerts are raised until the flood passes</li> </ul> </li> <li>• <b>mute_seconds</b> - The meaning depends on the repeat mode (default = 300 seconds): <ul style="list-style-type: none"> <li>◦ <code>repeat_mode == mute_for_period</code>: It defines the number of seconds for which the alert is muted</li> <li>◦ <code>repeat_mode == mute_until_passes</code>: It defines the number of second for which the flood has to stay undetected in order to be considered passed</li> </ul> </li> </ul> <pre> conf.user configure probe protocol arp flood {"enable": true}  conf.user configure probe protocol arp flood {"enable": true, "packet_limit": 5}'  conf.user configure probe protocol arp flood {"enable": true, "packet_limit": 100000, "packet_increase_limit": 100000, "packet_ratio_limit": 0.2}  conf.user configure probe packet-monitor enable true </pre>
---------------------------	---

<b>Where</b>	CLI
<b>To apply</b>	It is applied automatically

### Enable or disable the counting of all the packets to be used as reference for arp flood

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure probe packet-monitor enable [true false]</code>
<b>Description</b>	When this option is enabled, all the packets are counted to be used as reference for the relative ratio limit (default = false)

### Set the maximum number of subnets per MAC address that can be stored in the ARP table reconstructed from the processed traffic

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure arp_tables max_subnets_per_mac &lt;size&gt;</code>

<p><b>Description</b></p>	<p>This setting controls the maximum number of subnets per MAC address that can be stored in the ARP table reconstructed from the processed traffic.</p> <ul style="list-style-type: none"> <li>• <b>enable</b> - Enable or disable flood detection. Possible values <code>true</code> or <code>false</code> by default is <code>false</code>.</li> <li>• <b>packet_limit</b> - Define the number of packets per seconds (n) that trigger a flood detection (default = 1000).</li> <li>• <b>packet_increase_limit</b> - Define the increase in the number of packets per seconds that trigger a flood detection (default = 1000). The difference between <code>packet_limit</code> and <code>packet_increase_limit</code> is that the first one checks the number of packets per second while the second checks the increase in the number of packets per second. For example if <code>packet_limit</code> = 3000 and <code>packet_increase_limit</code> = 500 if the arp packets increase by 100 each second no alert is triggered until they reach the threshold of 3000 packets per second, but if they increase in one second of 600, for example they pass from 1000 to 1600, then the flood is triggered.</li> <li>• <b>packet_ratio_limit</b> - Define the ratio between the number of arp and total packets that trigger the flood . The flood is triggered when the ratio is greater than the specified value. In order to enable this detection method there is the need to enable also the counting of the total packets that can be done with <code>probe packet-monitor enable true</code>. A value of 0.1 means that the flood is triggered when the number of arp packets is 10% of the total packets. Default value is 1.0 (all packets are arp) in such a way that this threshold is disabled by default.</li> <li>• <b>strict_mode</b> - When <code>true</code> all the above checks have to be verified at the same time to trigger the detection, while when <code>strict_mode</code> is <code>false</code> it is enough for a single check to trigger the flood. Possible values <code>true</code> or <code>false</code> by default is <code>false</code>.</li> <li>• <b>repeat_mode</b> - Define how subsequent triggers are handled. Default value is <code>mute_for_period</code>. Possible values are: <ul style="list-style-type: none"> <li>◦ <b>repeat</b>: alerts are raised at each detection</li> <li>◦ <b>mute_for_period</b>: after the first detection alerts are muted for a specified period</li> <li>◦ <b>mute_until_passes</b> no subsequent alerts are raised until the flood passes</li> </ul> </li> <li>• <b>mute_seconds</b> - The meaning depends on the repeat mode (default = 300 seconds): <ul style="list-style-type: none"> <li>◦ <b>repeat_mode == mute_for_period</b>: It defines the number of seconds for which the alert is muted</li> <li>◦ <b>repeat_mode == mute_until_passes</b>: It defines the number of second for which the flood has to stay undetected in order to be considered passed</li> </ul> </li> </ul> <pre> conf.user configure probe protocol arp flood {"enable": true}  conf.user configure probe protocol arp flood {"enable": true, "packet_limit": 5}'  conf.user configure probe protocol arp flood {"enable": true, "packet_limit": 100000, "packet_increase_limit": 100000, "packet_ratio_limit": 0.2}  conf.user configure probe packet-monitor enable true </pre>
---------------------------	--

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

# Chapter 12. Vulnerability assessment



## Configure vulnerability assessments

### Configure the loading of Threat Intelligence Contents

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va contents &lt;json_value&gt;</code>
<b>Description</b>	<p>This command allows Threat Intelligence Contents to be either completely disabled, or selectively loaded. The JSON object can have the following attributes:</p> <ul style="list-style-type: none"> <li>• <code>load_contents</code> - this can be true/false to enable/disable the loading of contents;</li> <li>• <code>loaded_content_types</code> - this is a JSON array of contents to be loaded.</li> </ul> <p>The available content types are:</p> <ul style="list-style-type: none"> <li>• <code>cpe_items</code></li> <li>• <code>microsoft_hotfixes</code></li> <li>• <code>vulnass</code></li> </ul> <p>As an example, the following command will disable completely contents loading:</p> <pre>conf.user configure va contents { "load_contents": true }</pre> <p>As a further example, the following command will allow only <code>cpe_items</code> to be loaded:</p> <pre>conf.user configure va contents { "loaded_content_types": [ "cpe_items" ] }</pre>
<b>Parameters</b>	<code>json_value</code> : A JSON object to configure how contents are loaded
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osva stop</code>

### Configure CVE matching

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va cve enable [true false if_not_sync]</code>
<b>Description</b>	<p>By default, the sensors only match CVEs if they are not connected to an upstream (i.e. a CMC or Vantage). The CVE matching will happen upstream. This behavior can be configured using this configuration line, where 'true' forces the CVE matching even if the sensor is connected upstream, 'false' disables it in any case, and 'if_not_sync' restores the default behavior.</p>

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osva stop</code>

### Enables the management of Microsoft Hotfixes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va hotfixes_enabled &lt;flag&gt;</code>
<b>Description</b>	Please consider that when this is set to true hotfixes are loaded and used to set CVEs status whereas when this flag is set to false, hotfixes are not loaded nor used by CVE calculation.
<b>Parameters</b>	<code>flag</code> : The management of Microsoft Hotfixes is enabled by default
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable the Microsoft Hotfixes resolution capabilities

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va use_hotfix_resolution &lt;flag&gt;</code>
<b>Description</b>	Please consider that disabling the Microsoft Hotfixes resolution feature means that CVEs for Microsoft Windows machines will not be automatically closed through Smart Polling, and as a consequence those nodes might be assigned by Guardian a large number of obsolete CVEs.
<b>Parameters</b>	<code>flag</code> : Microsoft Hotfixes resolution is enabled by default
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable the CPE computation for a specific node

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va cpe disable &lt;node_id&gt; [true false]</code>
<b>Description</b>	Please consider that, when this command is used, the vulnerabilities assessment engine is completely disabled for that specific node and no CVEs will be assigned to the node itself.
<b>Parameters</b>	<code>node_id</code> : Node ID of the node targeting the rule



<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Disable End Of Life CPEs calculation

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure va use_eol_cpe_calculation false</code>
<b>Description</b>	By default, when CVE associated to CPES calculation is performed, CPE that are referring to products that reached End Of Life are not taken into account. To disable this behaviour use this configuration.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osva stop</code>



# Chapter 13. Customize node identifier generation



## Customize node identifier generation

All the entities that communicate in a network are called *nodes* and a Guardian assigns to each node a unique identifier, or *NodeID* in short. Generally, the NodeID is just an ip address (or a mac address), but in some special network topologies, extra information must be included in a NodeID to further differentiate nodes.

**Note:** NodeIDs generated with different settings will cause inconsistencies and should not coexist. These options should be manually set at sensor deploy time or on a Guardian with a clean configuration.

### Include VLAN number in NodeID

Nodes can have their NodeID "decorated" with the VLAN ID of their zone.

<b>Product</b>	Guardian
<b>Syntax</b>	<code>nodeid_factory zone</code>
<b>Description</b>	Nodes included in a zone, which has a non-zero VLAN id, will get a NodeID of the form <code>ip@vlan</code> .

### Include Remote Collector/Arc provenance in NodeID

Packets forwarded by Remote Collectors or Arc sensors carry a special "provenance" attribute that the Guardian uses to track precisely where the traffic was captured. The configuration directive `nodeid_factory include_capture` by default will use a standard NodeID for nodes seen by local capture devices, and append an explicative suffix `_from:...` to nodes appearing in remotely captured traffic.

<b>Product</b>	Guardian
<b>Syntax</b>	<code>nodeid_factory include_capture [local-traffic-tag] [format-string]</code>
<b>Description</b>	Enable decoration of NodeIDs with packet provenance information.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>the optional <code>local-traffic-tag</code> is the provenance name for locally captured traffic: leave empty or use <code>no_localhost</code> to disable NodeID decoration on local traffic.</li> <li>the <code>format-string</code> is the template for decorating remotely captured NodeIDs. A pair of curly braces <code>{}</code> will be expanded to the actual provenance. The default format is <code>"_from: {}"</code></li> </ul>

<b>Notes</b>	<p>For packets captured by a Remote Collector, the default provenance is the ip of the Remote Collector itself. Alternatively, a Guardian can use the <code>site</code> of the Remote Collector (and fall back to the ip, when the site is undefined), by adding the directive <code>remote_capture_forward_packet_src true</code> to the Guardian configuration.</p> <p>For packets captured by Arc sensors, the provenance is a unique identifier of the Arc instance, followed by the ip that uploaded the data to the Guardian (so, it may not belong to the machine where Arc is running).</p> <p>The Guardian may use both <code>include_capture</code> and <code>zone</code> if the configuration contains both <code>nodeid_factory zone</code> and <code>nodeid_factory include_capture</code> in this order.</p>
--------------	--

# Chapter 14. Decryption





## IEC 60870-5-7 / 62351-3/5 encrypted links

*IEC TC57 (POWER SYSTEMS management and associated information exchange) develops the standards 60870 and 62351. IEC 60870 part 5 (by WG3) describes systems used for telecontrol. IEC 62351 (by WG15) handles the security of TC 57 series.*

IEC TC57 WG15 recommends the combination of IEC 62351-3 and 5 to secure IEC 60870-5-104 links:

- IEC 62351-3 is a TLS profile to secure power systems related communication.
- IEC 62351-5 is an application security protocol applicable to IEC 60870-5-101, 104, and derivatives. Its implementation in terms of ASDUs (i.e., real encapsulation) is outlined in IEC 60870-5-7.

In order to decrypt IEC 62351-3 (TLS) traffic, you must meet these conditions:

- The private key for each TLS server (e.g. RTU, PLC) must be available; it is used to derive session keys.
- All the equipment where decryption is needed must operate using the **TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA** (0x00002f) cipher suite. Often, this step is accomplished by forcing either the client or the server to confine itself to that specific cipher suite.

## Configure IEC-62351-3

### About this task

The following steps assume we're decoding the communication of a TLS server with the address 192.168.1.26.

### Procedure

1. Upload the TLS server's private key to `/data/cfg`. The file name must match the server's address. In our case, the file must be named **192.168.1.26.key**.

Your key should be similar to the following:

```
-----BEGIN RSA PRIVATE KEY-----
MIICXQIBAAKBgQDIi4LBdRJTWgQBF91dRzYFPNb6ZsHDp9tuucB2vYkrfWXR/FLG
zFsEA+ws5BcKC/JxTy6Ud3IJL3dkPMLHEpIwvNH0EAWJ0wRXe4TLKD6vZH91Bu4X
yY/hI9W4MM4CD/WB/s/+QXnCGwXXSItXbNDnB6mbpNRKWrFCXrREFp09eQIDAQAB
AoGALcymrPXGnLfoXt0La30S2jldw0vqXnucQnpLeRKqFgN0WHNumBYdhxo60rpQ
hKvbz4LM2X053t0nQA4cUYhtMjoJDZk6uaj5e2lX5qDDCWqb5RsDiL+UcaMwA9wN
Dtu/zo49GdMKnPBx4Gcf8bTEngS7JyGJ9aQMzxUGUMFSgH0CQQD84Bg6vwpRwUY9
bigXQgUtggpmJ1VUVcIaZ8QNzbqwjB7C60fy06ci6kHs4Mi8Cc50UUjfinC3VSI
SjeVmdaLAKeAywXh7F0XtzWwPfu89gCLZh+/7cB8hDlmlg5FfDbZU7Nxq1ERGCoh
930GMnVaHhXjmye/ORYaVS6LnEUbeezAiwJBAJKTwveqaTcuVEyfYZW5qw6p5iU4
f3mXCAKldYmNYTic7xAbUAP9z1K/vt7zn0eCN62swDTzJkreihV/Mo2lekCQQDG
P8xJdadk5BNiDUoL/oohA0fr5law10xxoyX/EaB0t5975wGGUyTma22twCJr1nwX
ekykdgzouTbjNiikesHZAKBxp4B10nVSWAaYw88P61mr5QGL3oChZYQx15r1i9GI
LTWmncbyQu1Nm8df8wNdMgfvqWgbrlfJJqSIJtZhk5+U
-----END RSA PRIVATE KEY-----
```

2. In Guardian's Features Control Panel, enable link events; this provides visibility to the TLS decoded handshakes; for example:

192.168.0.104	192.168.1.26	tcp/19998	TCP_SYN:FIRST	3	52792	19998
192.168.0.104	192.168.1.26	tcp/19998	UP	4	52792	19998
192.168.0.104	192.168.1.26	iecl04s	SSLV2_CLIENT_HELLO(TLS-1.0)	5	52792	19998
192.168.1.26	192.168.0.104	iecl04s	TLS_SERVER_HELLO(TLS-1.0)	6	19998	52792
192.168.1.26	192.168.0.104	iecl04s	TLS_SERVER_HELLO(TLS_RSA_WITH_AES_128_CBC_SHA)	7	19998	52792
192.168.1.26	192.168.0.104	iecl04s	TLS_HANDSHAKE_CERTIFICATE	8	19998	52792
192.168.0.104	192.168.1.26	iecl04s	TLS_HANDSHAKE_CERTIFICATE	9	52792	19998
192.168.0.104	192.168.1.26	iecl04s	TLS_HANDSHAKE_CLIENT_KEY_EXCHANGE	10	52792	19998
192.168.0.104	192.168.1.26	iecl04s	TLS_CHANGE_CIPHER_SPEC(client)	11	52792	19998
192.168.1.26	192.168.0.104	iecl04s	TLS_CHANGE_CIPHER_SPEC(server)	12	19998	52792
192.168.0.104	192.168.1.26	iecl04s	DOWN	13	52792	19998

3. Specify the key file's location by defining it in the **CLI**. To continue our example, we would use the following string:

```
conf.user configure probe protocol iecl04s tls private_key
192.168.1.26 /data/cfg/192.168.1.26.key
```

4. Repeat these steps for each applicable TLS server key.
5. Run the following command in a shell console:

```
service n2osids stop
```



# Chapter 15. Trace



## Configure trace

### Trace size and timeout

A trace is a sequence of packets saved to the disk in the PCAP file format. The number of packets in a trace is fixed, this way when a trace of N packets is triggered Guardian starts to write to disk the N/2 packets that were sniffed before the trace was triggered, after that it tries to save another N/2 packets and then finalize the write operation, at this point the trace can be downloaded. To avoid a trace being pending for too much time there is also a timeout, when the time expires the trace is saved also if the desired number of packets has not been reached.

Trace files are stored in directory `/data/traces`, which employs disk based storage. In order to improve performance though, in machines with larger memory configurations this directory is backed by RAM based storage.

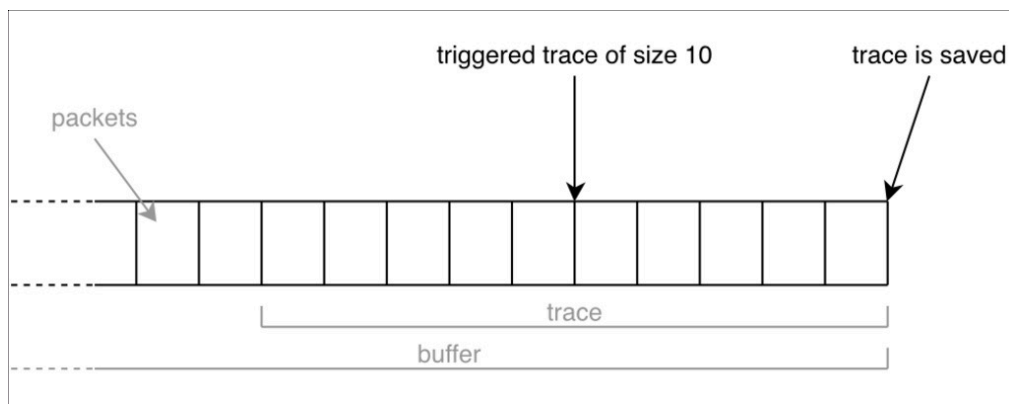


Figure 4. A schematic illustration of the trace saving process

### Set max trace packets

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure trace trace_size &lt;size&gt;</code>
<b>Description</b>	The maximum number of packets that will be stored in the trace file.
<b>Parameters</b>	size: Default value 5000
<b>Where</b>	CLI
<b>To apply</b>	It is applied automatically

### Set trace request timeout

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure trace trace_request_timeout &lt;seconds&gt;</code>

<b>Description</b>	The time in seconds after which the trace will be finalized also if the <code>trace_size</code> parameter is not fulfilled
<b>Parameters</b>	seconds: Default value 60
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set max pcaps to retain

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure trace max_pcaps_to_retain &lt;value&gt;</code>
<b>Description</b>	The maximum number of PCAP files to keep on disk, when this number is exceeded the oldest traces will be deleted. Both automatic alert traces and user-requested traces are included. This is a runtime machine setting used for self protection prevailing on the retention settings as described in the Configuring retention section
<b>Parameters</b>	value: Default value 100000
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set minimum free disk percentage

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure trace min_disk_free &lt;percent&gt;</code>
<b>Description</b>	The minimum percentage of disk free under which the oldest traces will be deleted. If the traces directory is memory backed, this configuration cannot be overridden and the default value will always be used.
<b>Parameters</b>	percent: Default value 10 if traces storage is disk backed, 5 if memory backed. Enter without % sign
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set maximum occupied space

<b>Product</b>	Guardian
----------------	----------



<b>Syntax</b>	<code>conf.user configure retention trace_request occupied_space &lt;max_occupied_bytes&gt;</code>
<b>Description</b>	The maximum traces occupation on disk in bytes. If the traces directory is memory backed, this configuration cannot be overridden and the default value will always be used.
<b>Parameters</b>	<code>max_occupied_bytes</code> : Default value is half of disk size if traces storage is disk backed, 95% of available space if memory backed
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

## Configure continuous trace

### Set max continuous trace occupation in bytes

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure continuous_trace max_bytes_per_trace &lt;size&gt;</code>
<b>Description</b>	The maximum size in bytes for a continuous trace file.
<b>Parameters</b>	size: Default value 100000000
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set max pcaps to retain

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure continuous_trace max_pcaps_to_retain &lt;value&gt;</code>
<b>Description</b>	The maximum number of PCAP files to keep on disk, when this number is exceeded the oldest traces will be deleted. This is a runtime machine setting used for self protection prevailing on the retention settings as described in the Configuring retention section
<b>Parameters</b>	value: Default value 100000
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set minimum free disk percentage

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure continuous_trace min_disk_free &lt;percent&gt;</code>
<b>Description</b>	The minimum percentage of disk free under which the oldest continuous traces will be deleted
<b>Parameters</b>	percent: Default value 10, enter without % sign
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Set maximum occupied space

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention continuous_trace &lt;occupied_space&gt;</code>
<b>Description</b>	The maximum continuous traces occupation on disk in bytes
<b>Parameters</b>	<code>occupied_space</code> : Default value is half of disk size
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



## Chapter 16. Time Machine



## Configure Time Machine

In this section we will configure the Nozomi Networks Solution Time Machine functionality.

### Set snapshot interval

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure tm snap interval &lt;interval_seconds&gt;</code>
<b>Description</b>	Set the desired interval between snapshots, in seconds.
<b>Parameters</b>	<code>interval_seconds</code> : The amount of seconds between snapshots (default: 3600, minimum: 3600)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	<code>service n2osjobs stop</code>

### Enable or disable automatic snapshot for each alert

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure tm snap on_alert [true false]</code>
<b>Description</b>	It can enable (option true) or disable (option false) the possibility to take a snapshot on alert. By default snapshots are taken only for VI alerts; it is possible to explicitly set the alerts that will trigger automatic snapshots via <code>on_alert_trigger</code>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	<code>service n2osjobs stop</code>

### Configure how alerts trigger automatic snapshots

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure tm snap on_alert_trigger &lt;json_value&gt;</code>

<b>Description</b>	<p>Configures the alert triggers for automatic snapshots. The JSON object must have the following attribute:</p> <p><code>type_ids</code> - A JSON array of the alert type IDs that will trigger an automatic snapshot. These type IDs may be literals or wildcarded ones (the asterisk can be used to match any substring).</p> <p>For example, the following command will configure the system to automatically take a snapshot whenever a VI:NEW-NODE or VI:NEW-LINK alert occurs:</p> <pre>conf.user configure tm snap on_alert_trigger {"type_ids": ["VI:NEW-NODE", "VI:NEW-LINK"]}</pre> <p>As a second example, the command below will configure the system to take a snapshot on all VI or SIGN alerts:</p> <pre>conf.user configure tm snap on_alert_trigger {"type_ids": ["VI:*", "SIGN:*"]}</pre>
<b>Parameters</b>	<code>json_value</code> : A JSON object describing the alert automatic snapshot triggers (default: {"type_ids": ["VI:*"]})
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	<code>service n2osjobs stop</code>

### Set maximum number of network elements allowed in a diff

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure tm diff max_results_network_elements &lt;num_elements&gt;</code>
<b>Description</b>	When comparing time machine snapshots that are too different, it is possible to overtax the system resources (memory, CPU). By setting a limit on the number of network elements that are allowed to be reported in a diff, the system is protected by such effects. When this threshold is crossed, the diff job is aborted and the appropriate error message is shown to the user.
<b>Parameters</b>	<code>num_elements</code> : Maximum number of network elements that may be reported by a diff (default: 10000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



# Chapter 17. Retention



## Configure retention

Retention of historical data is controlled for each persisted entity by a configuration entry. Modify it to extend or reduce the default retention.

By default, the CMC retains 500,000 alerts. Note that retaining large numbers of alerts can impair performance. We recommend limiting the number of alerts generated rather than retaining more data. If you want to retain more alerts, we recommend an iterative approach of incrementally increasing this value and evaluating the system's performance. In some cases, you may want to send alerts to a different system using our data integration features instead of retaining the alerts in the sensor.

### Alerts retention

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure retention alert rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of alerts to retain. <b>NOTE:</b> When an alert is deleted, the related trace file is deleted too.
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 500000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Alerts advanced retention

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure retention alert.out_of_security_profile rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of alerts out of security profile to retain. By default, this feature is disabled. <b>NOTE:</b> <ul style="list-style-type: none"> <li>• This retention has a higher priority than <code>retention alert rows &lt;rows_to_retain&gt;</code> and will be executed before it.</li> <li>• When an alert is deleted, the related trace file is deleted too.</li> </ul>
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (disabled by default)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

<b>Note</b>	You can also change this configuration from the Web UI.
-------------	---

### Trace retention size

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention trace_request occupied_space &lt;max_occupied_bytes&gt;</code>
<b>Description</b>	The maximum traces occupation on disk in bytes. If the traces directory is memory backed, this configuration cannot be overridden and the default value will always be used.
<b>Parameters</b>	<code>max_occupied_bytes</code> : Default value is half of disk size if traces storage is disk backed, 95% of available space if memory backed
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Trace retention rows

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention trace_request rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of traces to retain.
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 10000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Trace advanced retention

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure retention trace_request.&lt;generation_cause&gt; rows &lt;rows_to_retain&gt;</code>

<b>Description</b>	Set the amount of traces retained considering their generation cause. By default, these options are disabled. <b>NOTE:</b> This retention has a higher priority than <code>retention trace rows &lt;rows_to_retain&gt;</code> and will be executed before it. Moreover, These advanced retention options depend on each other, thus they must be configured all together or none.
<b>Parameters</b>	<ul style="list-style-type: none"> <li>• <code>generation_cause</code>: Can be any of: <ul style="list-style-type: none"> <li>• <code>by_alerts_high</code>: traces generated by high risk alerts</li> <li>• <code>by_alerts_medium</code>: traces generated by medium risk alerts</li> <li>• <code>by_alerts_low</code>: traces generated by low risk alerts</li> <li>• <code>by_user_request</code>: traces generated by a request from the user</li> </ul> </li> <li>• <code>rows_to_retain</code>: The number of rows to keep</li> </ul>
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

For example, we can configure the trace retention with the following command:

```
conf.user configure retention trace_request 10000
```

and also set up the advanced retention with:

```
conf.user configure retention trace_request 10000

conf.user configure retention trace_request.by_alerts_high 5000
conf.user configure retention trace_request.by_alerts_medium 1000
conf.user configure retention trace_request.by_alerts_low 1000
conf.user configure retention trace_request.by_user_request 3000
```

### Continuous trace retention size

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention continuous_trace occupied_space &lt;max_occupied_bytes&gt;</code>
<b>Description</b>	Set max occupation in bytes for continuous traces
<b>Parameters</b>	<code>max_occupied_bytes</code> : the number of bytes to keep (default: half of disk size)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2ostrace stop</code>

<b>Note</b>	You can also change this configuration from the Web UI.
-------------	---

### Continuous trace retention rows

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention continuous_trace rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of continuous traces to retain
<b>Parameters</b>	<code>rows_to_retain</code> : the number of rows to keep (default: 10000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Link events retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention link_event rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of link events to retain
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 2500000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Captured urls retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention captured_urls rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of captured "urls" (http queries, dns queries, etc) to retain
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 10000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

<b>Note</b>	You can also change this configuration from the Web UI.
-------------	---

### Variable history retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention variable_history rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the amount of variable historical values to retain
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 1000000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Node CVE retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention node_cve rows &lt;rows_to_retain&gt;</code>
<b>Description</b>	Set the maximum amount of node_cve entries to retain
<b>Parameters</b>	<code>rows_to_retain</code> : The number of rows to keep (default: 100000)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	In a shell console execute: <code>service n2osva stop</code>
<b>Note</b>	You can also change this configuration from the Web UI.

### Uploaded traces retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention input_pcap rows &lt;files_to_retain&gt;</code>
<b>Description</b>	Set the amount of PCAP files to retain
<b>Parameters</b>	<code>files_to_retain</code> : The number of files to keep (default: 10)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

<b>Note</b>	You can also change this configuration from the Web UI.
-------------	---

### File quarantine retention

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure retention quarantine number_of_files &lt;files_to_retain&gt;</code>
<b>Description</b>	Set the number of files to retain. When a new file is added to a sensor, Nozomi deletes the oldest quarantined file if the file exceeds this limit and the sensor needs to free disk space.
<b>Parameters</b>	<code>files_to_retain</code> : The number of files to keep (default: 50)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.



# Chapter 18. Bandwidth throttling



## Configure bandwidth throttling

It is possible to limit the bandwidth that a sensor's management port has at its disposal (so for access and updates) by specifying the maximum amount of allowed traffic.

### Limit traffic shaping bandwidth

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure system traffic_shaping bandwidth &lt;max_bandwidth&gt;</code>
<b>Description</b>	Set the maximum outbound bandwidth that the sensor's management interface can use. Inbound data is still unlimited. By default this works on all the outbound traffic and you can exclude some IPs. Instead if you use 'system traffic_shaping include ' it will limit only the outbound traffic of the inclusions and everything else will be unlimited.
<b>Parameters</b>	<code>max_bandwidth</code> : the bandwidth limit. The following units are supported: b, Kb, Mb, Gb. When no unit is specified, b is intended by default (i.e. bits per second). When setting a limit in decimal notation make sure you add the all the leading zeros, and the unit (e.g., write 0.015Mb, not .015Mb). (default: no limitation).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	Update rules with <code>n2os-firewall-update</code> . On a fresh installation a reboot is necessary.

For example, we can set a limit of two megabits with the following configuration command:

```
conf.user configure system traffic_shaping bandwidth 2Mb
```

Note that this command affects only the sensor on which it is executed, its effects are not propagated to other sensors.

It is possible to exclude from the limitation of the bandwidth specific hostnames or IPs.

### Exclude IP from traffic shaping

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure system traffic_shaping exclude &lt;ip&gt;</code>
<b>Description</b>	Set the IP to exclude from the limitation. This command and 'system traffic_shaping include' are mutually exclusive.
<b>Parameters</b>	<code>ip</code> : the IP to exclude. It can be a single IP, a class of IPS or an hostname (e.g. 192.168.12.34 or 192.168.0.0/16 or <code>vantage.nozominetworks.io</code> ).

<b>Where</b>	<b>CLI</b>
<b>To apply</b>	Update rules with <code>n2os-firewall-update</code> . On a fresh installation a reboot is necessary.

For example, we can exclude an IP with the following configuration command:

```
conf.user configure system traffic_shaping exclude 192.168.12.34
```

Note that this command affects only the sensor on which it is executed, its effects are not propagated to other sensors.

Or to include in the limitation of the bandwidth only specific hostnames or IPs.

### Include IP to traffic shaping

<b>Product</b>	Guardian
<b>Syntax</b>	<code>conf.user configure system traffic_shaping include &lt;ip&gt;</code>
<b>Description</b>	Set the IP to include in the limitation. This command and 'system traffic_shaping exclude' are mutually exclusive.
<b>Parameters</b>	<code>ip</code> : the IP to include. It can be a single IP, a class of IPS or an hostname (e.g. 192.168.12.34 or 192.168.0.0/16 or <code>vantage.nozominetworks.io</code> ).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	Update rules with <code>n2os-firewall-update</code> . On a fresh installation a reboot is necessary.

For example, we can include an IP with the following configuration command:

```
conf.user configure system traffic_shaping include 192.168.12.34
```

Note that this command affects only the sensor on which it is executed, its effects are not propagated to other sensors.

Inclusions and exclusions are mutually exclusive and you cannot use both simultaneously.

# Chapter 19. Synchronization



## Configure synchronization

In this section we will configure the synchronization between sensors at different levels.

### Set the global synchronization interval (notification message)

<b>Product</b>	CMC
<b>Syntax</b>	<code>conf.user configure cmc sync interval &lt;interval_seconds&gt;</code>
<b>Description</b>	<p>Set the desired global synchronization interval for the in-scope sensor. Configuration is defined on the parent sensor; synchronization starts at child sensors and flows upstream.</p> <p>Each and every sync takes place following a notification message sent by the child sensor, stating that the child sensor is ready to synchronize data to its parent. The notification messages act as global synchronization settings, working together with the following settings as well.</p> <p><b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), the setting must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).</p>
<b>Parameters</b>	<code>interval_seconds</code> : the number of seconds between status notifications (default: 60)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the DB synchronization interval

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure cmc sync_db_interval &lt;interval_seconds&gt;</code>
<b>Description</b>	<p>Set the desired interval between DB synchronizations for the in-scope sensor. Configuration is done on the parent sensor; synchronization starts at child sensors and flows upstream. The setting applies to each DB element subject to synchronization (e.g., Alerts, Assets, Audit logs, and Health logs). As the interval expires, the DB entries are synchronized at the next notification message.</p> <p><b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>interval_seconds</code>: the number of seconds between DB synchronizations (default: 60). This parameter only makes sense when set higher than the global synchronization interval.</li> </ul>
<b>Where</b>	<b>CLI</b>

<b>To apply</b>	It is applied automatically
-----------------	-----------------------------

### Set the filesystem synchronization interval

<b>Product</b>	CMC
<b>Syntax</b>	<code>conf.user configure cmc sync_fs_interval &lt;interval_seconds&gt;</code>
<b>Description</b>	<p>Set the desired interval between filesystem synchronizations for the sensor in scope, from its child sensors. The setting applies to each filesystem element subject to synchronization (e.g., nodes, links, and variables). As the interval expires, the filesystem entries are synchronized at the next notification message. In case the CMC is All-In-One, this interval will be used as default value for the <a href="#">Set the configurations merge interval (on page 154)</a>.</p> <p><b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).</p>
<b>Parameters</b>	<ul style="list-style-type: none"> <li><code>interval_seconds</code>: the number of seconds between filesystem synchronizations (default: 10800 [3 hours] if the CMC is multi-context, 720 (12 minutes) if the CMC is All-In-One). This parameter only makes sense when set higher than the global synchronization interval.</li> </ul>
<b>Where</b>	CLI
<b>To apply</b>	It is applied automatically

### Set the binary files synchronization interval

<b>Product</b>	CMC
<b>Syntax</b>	<code>conf.user configure cmc sync_binary_files_interval &lt;interval_seconds&gt;</code>
<b>Description</b>	<p>Set the desired interval between binary files synchronizations for the sensor in scope, from its child sensors. The setting applies to each binary file element subject to synchronization (e.g., PDF reports). As the interval expires, the binary file entries are synchronized at the next notification message.</p> <p><b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).</p>
<b>Parameters</b>	<p><code>interval_seconds</code>: the number of seconds between binary files synchronizations (default: 60). This parameter only makes sense when set higher than the global synchronization interval.</p>



<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the rows to be sent at every DB synchronization for each DB element

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure cmc sync record_per_loop &lt;number_of_record_per_loop&gt;</code>
<b>Description</b>	The system allows the user to customize the synchronization, in particular the number of records to be sent at each phase. A synchronization phase is composed of 50 steps for each DB element, every one sending <code>number_of_record_per_loop</code> rows, which means that the system sends, by default, 2500 rows every time.
<b>Parameters</b>	<code>number_of_record_per_loop</code> : the number of DB rows sent per single request (default: 50)
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Synchronize only visible alerts

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure cmc sync send_only_visible_alert [true false]</code>
<b>Description</b>	Set whether to synchronize all alerts from the child sensors to the in-scope parent sensor (false), or to synchronize only visible alerts (as defined in the Security Profile) (true). Default: false. <b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the alert rules execution policy

<b>Product</b>	CMC
----------------	-----

<b>Syntax</b>	<code>conf.user configure alerts execution_policy alert_rules [upstream_only upstream_prevents local_prevents]</code>
<b>Description</b>	Set the desired execution policy for the alert rules. <b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically
<b>Note</b>	You can also change this configuration from the Web UI.

### Set the configurations merge interval

<b>Product</b>	CMC
<b>Syntax</b>	<code>conf.user configure cmc merge interval &lt;interval_seconds&gt;</code>
<b>Description</b>	Periodically, CMC All-In-One merge all the filesystem elements received by the connected Guardians. This operation strictly depends on the filesystem synchronization. It is possible to define a custom interval, however it is suggested to specify a similar value as the one set for the filesystem synchronization. <b>Note:</b> In a multi-level deployment (e.g., one with root CMC, local CMC, and Guardian), if the setting is applicable, it must be applied at each parent level (e.g., at the root CMC as well as at the local CMC).
<b>Parameters</b>	<code>interval_seconds</code> : the number of seconds between two merging actions (default: it follows the value of the <a href="#">Set the configurations merge interval (on page 154)</a> ).
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable the PostgreSQL advisory locks for assets synchronization

<b>Product</b>	CMC
<b>Syntax</b>	<code>conf.user configure cmc save_assets_with_advisory_lock [true false]</code>

<b>Description</b>	Enable this feature to avoid potential database deadlocks on assets. This option shall be applied on mid-level CMC if the bulk asset synchronization is not enabled. <b>Note:</b> This option applies only on the CMC it is configured on. It is enabled by default.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Enable content modifications on a Guardian or CMC to be local only

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure threat_intelligence local_contents enable [true false]</code>
<b>Description</b>	Enable this feature to be able to modify contents on a machine even when it has an upstream connection. All changes will be local only, and changes from upstream will not be pushed down. If the feature is afterwards disabled, all the changes from the upstream will be propagated, and overwrite all the modifications done when the feature was enabled.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



## Chapter 20. Slow updates



## Configure slow updates

In this section we show how to configure a sensor for receiving firmware updates from an upstream sensor at a determined speed. This option is suitable for those scenarios where a limited bandwidth is available, and a normal firmware update procedure would result in a timeout. For example, one may want to configure a remote collector constrained by a 50Kbps bandwidth to receive the updates at 24Kbps. This configuration will prevent the saturation of the communication channel and thus it will allow to send data traffic while receiving an update.

### Enable or disable slow update

<b>Products</b>	CMC, Guardian, Remote Collector
<b>Syntax</b>	<code>software_update_slow_mode [true false]</code>
<b>Description</b>	This is a global switch that enables (true) or disables (false) the feature. When the feature is disabled all the other switches are ignored.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the transfer chunk size

<b>Products</b>	CMC, Guardian, Remote Collector
<b>Syntax</b>	<code>software_update_slow_mode_chunk_size &lt;size_in_bytes&gt;</code>
<b>Description</b>	The update bundle is split into multiple fixed-sized chunks. Chunks are individually transmitted, verified and reassembled. In case of a failed delivery, only invalid chunks are retransmitted. Essentially, big chunks are more suitable for high-speed networks, while smaller chunks are to be preferred for more effective bandwidth limitation.
<b>Parameters</b>	<code>size_in_bytes</code> : the chunk size in bytes. Values are normalized to stay in the range [128, 10485760]. Default value is 4096.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

### Set the transfer speed

<b>Products</b>	CMC, Guardian, Remote Collector
<b>Syntax</b>	<code>software_update_slow_mode_max_speed &lt;speed_in_bps&gt;</code>

<b>Description</b>	Sets the maximum allowed speed for update transfer.
<b>Parameters</b>	<code>speed_in_bps</code> : The maximum allowed speed in bytes per second. Values lower than 1024 are normalized to 1024. The default value is 4096. Notice that small chunks add some slight overhead, so generally the transfer speed will remain consistently below the declared limit.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically



# Chapter 21. Session hijacking



## Configure session hijacking protection

Web management interface protects itself from session hijacking attacks binding web session to ip addresses and browser configurations. When it detects differences on these parameters it automatically destroys the session and records the error in the audit log. This feature is enabled by default and it can be disabled using this configuration:

### Disable session hijacking protection

<b>Products</b>	CMC, Guardian
<b>Syntax</b>	<code>conf.user configure ui session protection [true false]</code>
<b>Description</b>	Enable (option true, default behavior) or disable (option false) session hijacking protection.
<b>Where</b>	<b>CLI</b>
<b>To apply</b>	It is applied automatically

When closing sessions the web management interface will record in the audit log this error text

```
Session hijacking detected, closing session
```

and the details of the affected session.



## Chapter 22. Passwords



## Configure passwords

*This topic describes N2OS password parameters and their default values. Modifying the default password requires that you change the configuration of the sensor.*

## Set maximum attempts

### Product

Guardian

```
conf.user configure password_policy maximum_attempts <attempts>
```

### Description

Set the maximum number of attempts for inserting a password, before the system locks.

### Parameters

`attempts`: The maximum number of attempts allowed (default: 3)

### Where

**CLI**

### To apply

It is applied automatically.



## Enable pbkdf2 sha512 password hashing algorithm

### Product

Guardian

```
conf.user configure password_policy hashing_pbkdf2 [true|false]
```

### Description

Enable or disable the pbkdf2 sha512 password hashing algorithm (default: false).

### Where

CLI

### To apply

It is applied automatically.

## Set lock time

### Product

Guardian

```
conf.user configure password_policy lock_time <minutes>
```

### Description

Set the maximum number of attempts for inserting a password, before the system locks.

### Parameters

**minutes:** The number of minutes for which a user account is locked out after having failed to login for the maximum attempts (default: 5)

### Where

**CLI**

### To apply

It is applied automatically.

## Set history

### Product

Guardian

```
conf.user configure password_policy history <number>
```

### Description

Set the number of historical passwords that are required to be unique (default: 3).

### Parameters

**number:** The number of unique passwords to be used.

### Where

**CLI**

### To apply

It is applied automatically.

## Set password digits

### Product

Guardian

```
conf.user configure password_policy digit <number>
```

### Description

Sets the minimum number of digits that are required to be contained in a password (default: 1).

### Parameters

`number`: The number of digits required.

### Where

**CLI**

### To apply

It is applied automatically.

## Set password lower

### Product

Guardian

```
conf.user configure password_policy lower <number>
```

### Description

Sets the minimum number of lowercase characters that are required to be contained in a password (default: 1).

### Parameters

`number`: The number of lowercase characters required.

### Where

**CLI**

### To apply

It is applied automatically.

## Set password upper

### Product

Guardian

```
conf.user configure password_policy upper <number>
```

### Description

Sets the minimum number of uppercase characters that are required to be contained in a password (default: 1).

### Parameters

`number`: The number of uppercase characters required.

### Where

**CLI**

### To apply

It is applied automatically.

## Set password symbol

### Product

Guardian

```
conf.user configure password_policy symbol <number>
```

### Description

Sets the minimum number of symbol characters that are required to be contained in a password (default: 0).

### Parameters

**number:** The number of symbol characters required.

### Where

**CLI**

### To apply

It is applied automatically.

## Set password min length

### Product

Guardian

```
conf.user configure password_policy min_password_length <number>
```

### Description

Sets the minimum length required for a password (default: 12).

### Parameters

**number:** The minimum length.

### Where

**CLI**

### To apply

It is applied automatically.



## Set password max length

### Product

Guardian

```
conf.user configure password_policy max_password_length <number>
```

### Description

Sets the maximum length required for a password (default: 128).

### Parameters

`number`: The maximum length.

### Where

**CLI**

### To apply

It is applied automatically.

## Enable expire for inactive users

### Product

Guardian

```
conf.user configure password_policy inactive_user_expire_enable [true|false]
```

### Description

Enable or disable the expiration for inactive users (default: false).

### Parameters

`number`: The maximum length.

### Where

CLI

### To apply

It is applied automatically.

## Set user lifetime

### Product

Guardian

```
conf.user configure password_policy inactive_user_lifetime <number>
```

### Description

Sets the required inactive days after which a user is forced as disabled (default: 60).

### Parameters

`number`: The number of days.

### Where

**CLI**

### To apply

It is applied automatically.

## Enable expiration for admin users

### Product

Guardian

```
conf.user configure password_policy admin_can_expire [true|false]
```

### Description

Enable or disable the expiration for inactive admin users (default: false).

### Where

**CLI**

### To apply

It is applied automatically.

## Enable password expiration

### Product

Guardian

```
conf.user configure password_policy password_expire_enable [true|false]
```

### Description

Enable or disable the expiration for passwords (default: false).

### Where

**CLI**

### To apply

It is applied automatically.

## Set password lifetime

### Product

Guardian

```
conf.user configure password_policy password_lifetime <number>
```

### Description

Sets the required number of days after which a password change is enforced (default: 90).

### Parameters

`number`: The number of days.

### Where

**CLI**

### To apply

It is applied automatically.

## Chapter 23. Sandbox





## Sandbox archive processing

*A description of sandbox archive processing.*

When a sandbox file contains an archive, the archive is uncompressed and each extracted file is processed to check the presence of malware using the Yara rules and indicators. The process is repeated recursively for each extracted file to eventually check for malware in nested archives. The 'archive' configurations commands listed below permits to control how this unpacking and checking process is performed. Since the unpacking and checking can consume significant resources, the tuning of these parameters can be important when a large number of potentially large files is present in the sandbox.

## Set the number of workers

```
conf.user configure sandbox dispatcher number_of_workers <value>
```

### Product

Guardian

### Description

Service will start one dispatcher application and as many worker applications as requested. By default, Sandbox will instead start in standalone mode.

### Parameters

value: Maximum value of 8.

### Where

CLI

### To apply

It is applied automatically.

## Set the size of the sandbox tmpfs partition

```
conf.user configure sandbox tmpfs sandbox <value>
```

### Product

Guardian

### Description

The tmpfs in-memory partition should be big enough to host two times the maximum number of retained files.

### Parameters

value: Size in MB of the partition. Defaults to 400MB.

### Where

**CLI**

### To apply

It is applied upon the reboot of the sensor.

## Set the size of the sandbox tmpfs temporary partition

```
conf.user configure sandbox tmpfs tmp_sandbox <value>
```

### Product

Guardian

### Description

For high throughputs, the tmpfs in-memory partition should be big enough to host all the files that can fit in the sandbox tmpfs partition described in the previous command.

### Parameters

`value`: Size in MB of the partition. Defaults to 100MB.

### Where

**CLI**

### To apply

It is applied upon the reboot of the sensor.

## Set the size of the sandbox tmpfs pipes partition

```
conf.user configure sandbox tmpfs pipes <value>
```

### Product

Guardian

### Description

10MB should be allocated for every worker that has been configured.

### Parameters

`value`: Size in MB of the partition. Defaults to 10MB.

### Where

CLI

### To apply

It is applied upon the reboot of the sensor.

# Configure how Threat Intelligence contents are handled

```
conf.user configure sandbox contents <json_value>
```

## Product

Guardian

## Description

This command configures how Threat Intelligence contents are to be loaded. The JSON object can have the following attributes:

- `load_contents` - this can be true/false to enable/disable the loading of contents;
- `stix_backend_provider` - selects the engine to be used for STIX indicators: 'memory' (indicators are stored in memory - default) or 'db' (indicators are stored in the system database). With 'db', the application memory requirements become lower, but STIXv1 (XML) indicators are not supported;
- `loaded_content_types` - this is a JSON array of contents to be loaded.

Contents available are:

- `stix_indicators`
- `yara_rules`

As an example, the following command will disable completely contents loading:

```
conf.user configure sandbox contents { "load_contents": false }
```

As a further example, the following command will allow only yara rules to be loaded:

```
conf.user configure sanbox contents { "loaded_content_types":  
[ "yara_rules" ] }
```

As another example, the following command will configure the usage of the system database for storing STIX indicators related information, reducing the application memory footprint:

```
conf.user configure sanbox contents { "stix_backend_provider": "db" }
```

## Parameters

`json_value`: A JSON object to configure how Threat Intelligence contents are loaded

## Where

CLI

## To apply

It is applied automatically.

## Set sandbox strategies for vulnerability assessment

### Product

Guardian

```
conf.user configure sandbox strategies <json_value>
```

### Description

The schema for the configuration options is:

```
{"enabled_strategies": ["yara_rules", "stix_indicators"]}
```

### Parameters

`json_value`: Define which strategies will be used to analyze files.

### Where

CLI

### To apply

It is applied automatically.

## Set the minimum percentage of free disk

### Product

Guardian

```
conf.user configure sandbox min_disk_free <value>
```

### Description

Minimum free disk percentage that should be observed in the tmpfs `/var/sandbox` folder, where `n2os_ids` will write the captured files.

### Parameters

value: Default value 10

### Where

CLI

### To apply

It is applied automatically.



## Set the maximum number of files to retain

### Product

Guardian

```
conf.user configure sandbox max_files_to_retain <value>
```

### Description

Maximum number of files to retain in the tmpfs /var/sandbox folder, where n2os\_ids will write the captured files. The actual number may be up to two times higher under heavy loading in order to improve the performance of the n2os\_sandbox process.

### Parameters

value: Default value 250

### Where

CLI

### To apply

It is applied automatically.

# Set the size of the asynchronous processing queues

## Product

Guardian

```
conf.user configure sandbox queues queue_length <value>
```

## Description

Length of the asynchronous queues processing and analysing the captured files. This number is applied to both the standalone, dispatcher and worker applications. This figure should be increased to the number of files which are expected to be handled by the application every 250ms. After the application of this setting, also the size of the tmpfs folder partitions needs to be carefully tuned as described in the corresponding commands.

## Parameters

value: Default value 200

## Where

**CLI**

## To apply

It is applied automatically.

## Set the interval for stale file analysis

### Product

Guardian

```
conf.user configure sandbox dispatcher files_timeout <value>
```

### Description

Stale files which have not been analyzed and cleaned by the worker applications under severe loading are garbage collected by default every hour. This is an exceptional situation which should not even occur under heavy loading, but this setting is provided as an additional protection mechanism.

### Parameters

value: Default value 3600 seconds

### Where

CLI

### To apply

It is applied automatically.

## Set the maximum number of extracted files

### Product

Guardian

```
conf.user configure archive max_number_of_files <value>
```

### Description

Defines the maximum number of files that can be extracted and processed from an archive file. If the maximum number of extracted files is reached for a file, then additional files nested inside it are neither expanded nor processed.

### Parameters

value: Default value 100

### Where

**CLI**

### To apply

It is applied automatically.

## Set the maximum level of nested extracted files

### Product

Guardian

```
conf.user configure archive max_levels <value>
```

### Description

Defines the maximum level of nested archives that are extracted and processed from an archive file. Files nested in archives at a level deeper than the specified one are neither extracted nor processed.

### Parameters

value: Default value 3

### Where

CLI

### To apply

It is applied automatically.

# Set the maximum size of a single extracted file

## Product

Guardian

```
conf.user configure archive max_single_size <size_in_bytes>
```

## Description

Defines the maximum size in bytes of a file that can be extracted and processed from a compressed archive file. If a file is larger than the limit it is neither extracted nor processed.

## Parameters

size\_in\_bytes: Default value 200000000 (200M)

## Where

**CLI**

## To apply

It is applied automatically.

## Set the maximum overall size of extracted files

### Product

Guardian

```
conf.user configure archive max_overall_size <size_in_bytes>
```

### Description

Defines the overall maximum size of the files that can be extracted from a single compressed archive file. If the overall size of the files extracted from a file exceed this limit, the remaining files in the archive are neither extracted nor processed.

### Parameters

size\_in\_bytes: Default value 400000000 (400M)

### Where

CLI

### To apply

It is applied automatically.

# Set the maximum time to be spent for each file extraction

## Product

Guardian

```
conf.user configure archive max_wait_msec <time_in_millisecs>
```

## Description

Defines the maximum time to be spent during a file extraction from a compressed archive file. If the spent time exceeds the limit, the extraction is aborted.

## Parameters

`time_in_millisecs`: Default value 10000 millisecs

## Where

**CLI**

## To apply

It is applied automatically.



# Define maximum recursion depth allowed for VBA extraction

## Product

Guardian

```
conf.user configure archive vba_macro_extraction max_recursion <value>
```

## Description

VBA archives unzipping could require a high number of recursive steps to complete full recursion: this could lead to possible performance issues due to high number of memory allocations. This parameter limits the number of recursions allowed to unzip a single file.

## Parameters

value: Default value 100

## Where

CLI

## To apply

It is applied automatically.

# Enable or disable the adaptive algorithm for file unzipping

## Product

Guardian

```
conf.user configure archive auto_switch_off <flag>
```

## Description

Unzipping is a very expensive process which is automatically disabled when Sandbox is under heavy loading. Instead of discarding files, Sandbox will disable unzipping for some files and process only the unzipped file with STIX and Yara indicators.

## Parameters

`flag`: Default to true. false to disable the feature

## Where

CLI

## To apply

It is applied automatically.

# Configuring handling of sandboxed zipped files

## Product

Guardian

```
conf.user configure sandbox unzipping <json_value>
```

## Description

The json object can have the following attributes: \* modes - array of unzipping modes which should be enabled. By default all of them are enabled and are executed in the described order. Possible values are: fast, for fast unzipping, macro, for macro extraction and analysis, upx, for upx decompression, full, for extensive and advanced archive decompression. An empty array can be used to completely disable the unzipping functionalities of Sandbox.

```
conf.user configure sandbox unzipping {"modes": ["macro", "upx",  
"full"]}
```

## Parameters

json\_value: A json object to configure how zipped files are handled by Guardian

## Where

CLI

## To apply

It is applied automatically.

## Configure high throughput protection

```
conf.user configure sandbox extraction <json_value>
```

### Product

Guardian

### Description

This option lets you define which files have to be analysed by Sandbox. The `enable_...` suboptions enable only the files that satisfy the specific criteria, while the `disable_...` suboptions disable only the files that match the specific criteria and enable all the others. In the case that several criteria are specified they are applied with an AND sequence (i.e. all of them have to be satisfied). The user should note in the following that advertised file extensions are considered. If an attacker hides behind a [joint photographic experts group \(JPEG\)](#) file extension a malicious executable, there is no way for Sandbox to understand that the file is an executable without performing an in-depth analysis on the file itself. For this reason, we highly discourage the use of the file extension attribute in the [JavaScript Object Notation \(JSON\)](#) below. Protocols, zones and node criteria are instead encouraged, when even the auto switch off adaptive algorithm cannot provide a sufficient protection against high throughputs.

The json object can have the following attributes:

- `enabled_protocols` - only files extracted from these protocols will be analysed
- `disabled_protocols` - files extracted from these protocols will be excluded from the analysis
- `enabled_file_extensions` - only files extracted with these advertised extensions will be analysed
- `disabled_file_extensions` - files extracted with these advertised extensions will be excluded from the analysis
- `enabled_zones` - only files extracted from the specified zones (both source and destination) will be analysed.
- `disabled_zones` - files extracted from the specified zones (both source and destination) will be excluded from the analysis.
- `enabled_src_zones` - only files extracted from the specified source zones will be analysed.
- `disabled_src_zones` - files extracted from the specified source zones will be excluded from the analysis.
- `enabled_dst_zones` - only files extracted from the specified destination zones will be analysed.
- `disabled_dst_zones` - files extracted from the specified destination zones will be excluded from the analysis.
- `enabled_node_types` - only files extracted from nodes of the specified types will be analysed (both source and destination).
- `disabled_node_types` - files extracted from nodes of the specified types will be excluded from the analysis (both source and destination).
- `enabled_src_node_types` - only files extracted from packets with source nodes of the specified types will be analysed.
- `disabled_src_node_types` - files extracted from packets with source nodes of the specified types will be excluded from the analysis.
- `enabled_dst_node_types` - only files extracted from packets with destination nodes of the specified types will be analysed.
- `disabled_dst_node_types` - files extracted from packets with destination nodes of the specified types will be excluded from the analysis.
- `enabled_node_ids` - only files extracted from nodes with the specified ids will be analysed (both source and destination).
- `disabled_node_ids` - files extracted from nodes with the specified ids will be excluded from the analysis (both source and destination).
- `enabled_src_node_ids` - only files extracted from packets with source nodes contained in the specified ids will be analysed.
- `disabled_src_node_ids` - files extracted from packets with source nodes contained in the specified ids will be excluded from the analysis.
- `enabled_dst_node_ids` - only files extracted from packets with destination nodes contained in the specified ids will be analysed.
- `disabled_dst_node_ids` - files extracted from packets with destination nodes contained in the specified ids will be excluded from the analysis.

```
conf.user configure sandbox extraction {"enabled_protocols": ["http"]}
```

### Parameters

`json_value`: A *JSON* object to configure which files are not analysed by Sandbox.

### Where

**CLI**

### To apply

It is applied automatically.

## Chapter 24. Miscellaneous





## Configure SSH key update interval

```
conf.user configure ssh_key_update interval <seconds>
```

### Product

Guardian

### Description

See **Add an SSH key for an admin user**, in the **Maintenance Guide**.

### Parameters

seconds: Number of seconds between propagations.

### Where to apply

CLI

### How to apply

It is applied automatically.

## Configure paranoid mode for user login authentication

```
conf.user configure authentication paranoid_mode [true|false]
```

### Product

Guardian

### Description

Paranoid mode in authentication is enabled by default. It is used to control the disclosure of information about the existence of a user during the login authentication process and to normalize the login response time. When this setting is disabled, after several failed login attempts, the user is warned with a message about the remaining attempts before the account gets locked. As a consequence, the user information can be leaked. However, when this setting is enabled, there is no warning message and thus no potential for user information leak.

### Where to apply

CLI

### How to apply

In a shell console execute: `service webserver stop`

## Configure identity provider

```
conf.user configure cmc identity_provider_url <ip>
```

### Product

*Central Management Console (CMC)*, Guardian

### Description

Expose identity provider endpoint.

### Parameters

`ip`: The address of the identity provider.

### Where to apply

**CLI**

### How to apply

It is applied automatically.



# Glossary



**Berkeley Packet Filter**

The BPF is a technology that is used in some computer operating systems for programs that need to analyze network traffic. A BPF provides a raw interface to data link layers, permitting raw link-layer packets to be sent and received.

**Central Management Console**

The Central Management Console (CMC) is a Nozomi Networks product that has been designed to support complex deployments that cannot be addressed with a single sensor. A central design principle behind the CMC is the unified experience, that lets you access information in the similar method to the sensor.

**Command-line interface**

A command-line processor uses a command-line interface (CLI) as text input commands. It lets you invoke executables and provide information for the actions that you want them to do. It also lets you set parameters for the environment.

**JavaScript Object Notation**

JSON is an open standard file format for data interchange. It uses human-readable text to store and transmit data objects, which consist of attribute–value pairs and arrays.

**Joint Photographic Experts Group**

JPEG, or JPG, is a method of lossy compression that is used for digital images. The degree of compression can be adjusted, allowing a selectable tradeoff between storage size and image quality.

**Secure Shell**

A cryptographic network protocol that let you operate network services securely over an unsecured network. It is commonly used for command-line execution and remote login applications.

**Universal Serial Bus**

Universal Serial Bus (USB) is a standard that sets specifications for protocols, connectors, and cables for communication and connection between computers and peripheral devices.

**Variable**

In the context of control systems, a variable can refer to process values that change over time. These can be temperature, speed, pressure etc.

