Documentation
GCap Version 2.5.3.107

GATEWATCHER

Documentation version: V1

Translated from original manual version 1

Creation date: January, 2023

Contents

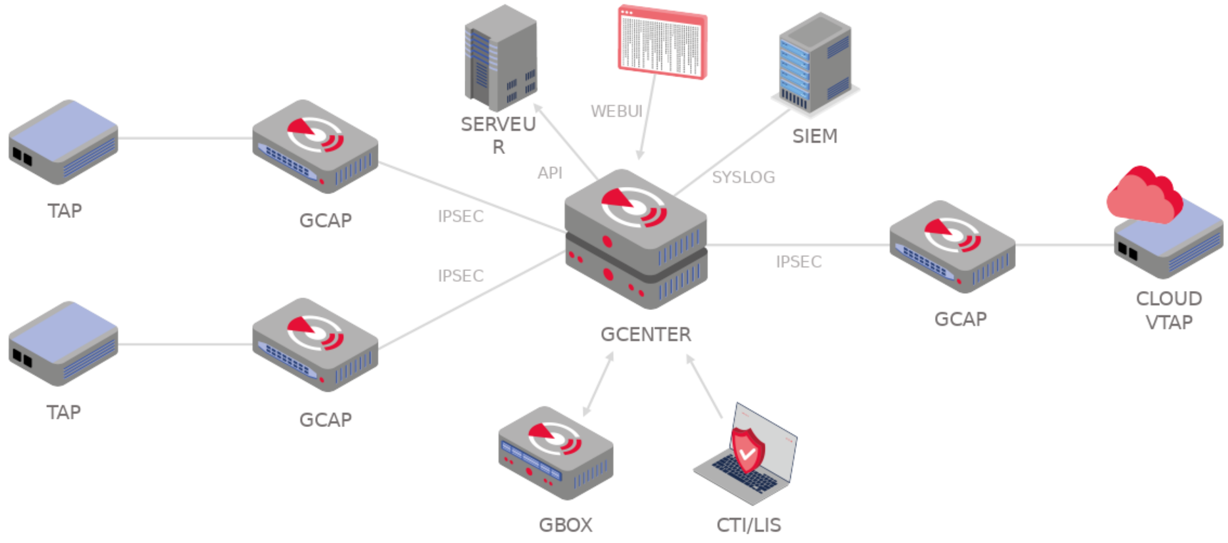
Chapter 1

Description

1.1 Introduction

The Trackwatch/Aioniq solution includes:

- One or more TAPs
- One or more GCaps
- A GCenter



1.2 TAP

A Test Access Point (TAP) is a passive device enabling the monitoring of a computer network by duplicating the flows in transit and redirecting them to an analysis and detection probe (the GCap).
It is possible to connect several TAPs to one GCap.

1.3 GCap

GCap is a probe-type component.
It enables:

- Capturing and analysing network traffic from TAPs
- Generating events, alerts, and metadata
- Rebuilding the files contained in the flow
- Communicating with the GCenter

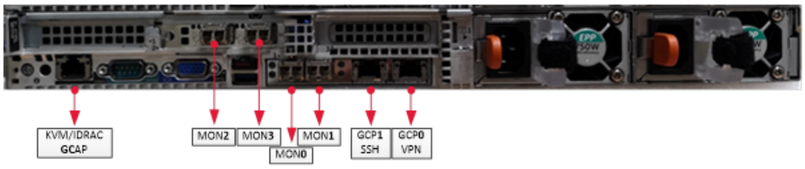
1.3.1 Different server models

For more information, please refer to the *Characteristics* section.

1.3.2 Description of the GCap inputs / outputs

The GCap detection probe features:

- A USB and VGA connector to directly access a keyboard and a monitor. This connection mode is deprecated in favour of KVM/IDRAC/XCC and should only be used as a last resort
- A USB connector accommodates the USB key enabling disk decryption (standard Linux Unified Key Setup)
- One RJ-45 connector to access the server management and configuration interface (KVM/IDRAC/XCC)
- Two RJ-45 connectors `gcp0` and `gcp1`
- RJ-45 and/or fibre connectors for monitoring `mon0`
- Two power supplies



1.3.2.1 Use of USB and VGA connectors

Connecting a keyboard and monitor enables direct access to the server's console interface.

Important:

This mode is deprecated. It should only be used during initial installation and for advanced diagnosis.

1.3.2.2 Access to the server's management and configuration interface

Access to this management interface is via HTTPS:

- On a Dell server, this connector is called **iDRAC**. It is noted on the **KVM/IDRAC GCap** diagram
- On a Lenovo server, this connector is called **TSM**. This connector can be identified by a wrench symbol on the bottom of it.

1.3.2.3 Interface network `gcp0` and `gcp1`

These interfaces perform the following functions:

- Function 1: secure communication between the probe and GCenter through an IPSEC tunnel in order to:
  - Escalate information such as files, alerts, metadata, and so on, derived from analysing the monitored flows
  - Report information on the health of the probe to GCenter
  - Control the probe - analysis rules, signatures, and so on.
- Function 2: remote administration through the SSH protocol with access:
  - To the probe's command line interface (CLI)
  - To the graphical setup/configuration menu (deprecated)

In **single-interface configuration**, these functions are supported by the `gcp0` interface only.
In **dual-interface configuration**:

- Function 1 is handled by interface `gcp0`
- Function 2 is handled by interface `gcp1`

1.3.2.3.1 Configuration of the `gcp0` and `gcp1` network interfaces

For more information on these interfaces and their configuration, refer to the section *Network interfaces `gcp0` and `gcp1`*.

1.3.2.4 Capture and monitoring interfaces

These interfaces receive:

- Flows from the TAPs on the indicated interfaces (`mon0` and `monx`)
- The flow from previously recorded files (pcap files) on a dedicated `monvirt` interface

> **Note:**
>
> The number of capture interfaces varies depending on the specifications of each model.

1.3.2.4.1 Activating the capture and monitoring `monx` interfaces

For more information, please refer to the paragraph *Capture and monitoring interfaces: activation*.

1.3.2.4.2 Aggregating the capture and monitoring `monx` interfaces

For more information, see the paragraph *Capture and monitoring interfaces between TAP and GCap: aggregation capability*.

1.3.3 Electrical connection

The probe has two power supplies, each of which has the necessary power to operate the equipment.
It is strongly recommended that each power supply should be connected to a separate power supply.

1.3.4 USB connector and LUKS key

During installation, the contents of the disks (excluding /boot) are encrypted using the LUKS standard.
During this process, a unique encryption key is created and placed on the USB stick connected to the probe.
It is strongly recommended to make a copy of this key because, in the event of failure, the data on the disks will no longer be accessible.
Once the system is up and running, the USB stick should be removed and placed in a secure place (e.g. in a safe).

1.4 GCenter

The GCenter is the second component of the system working in conjunction with the GCap detection probe.
Its main functions include:

- The management of the GCap probe including managing the analysis rules, signatures, health status supervision, and so on.
- In-depth analysis of the files retrieved by the probe
- Administering the system
- Displaying the results of the various analyses in different dashboards
- Long-term data storage
- Exporting data to third-party solutions such as the Security Information and Entente Management (SIEM) system

For more information, please refer to the GCenter documentation.

1.5 Interconnection of subsets

1.5.1 Reminder of the GCap connections

Depending on the timing and configuration chosen, and looking from behind from left to right, the GCap is connected via:

- A network socket for connecting a KVM / iDRAC
- A USB and VGA connector for a keyboard and monitor
- Capture and monitoring interfaces `mon0`, `mon1`, `mon2`, `monx` for connecting TAPs
- The network interfaces `gcp0` and `gcp1`
  Depending on the chosen configuration - single or dual interface - it is possible to use these network interfaces for connecting to the GCenter.
- The connectors for the GCap power supplies

For more information on the connection description, please refer to the *Description of GCap inputs/outputs*.

Note:

Remember to connect the LUKS decryption key to the USB port.

1.5.2 Capture and monitoring interfaces `monx` between TAP and GCap: aggregation possibility

The GCap probe must read in a single flow; the network flow that has been captured in both directions:

- An uplink,
- A downlink.

To do this, the flows from each of the links must be aggregated into a single flow. There are 2 solutions for this:

- Either the flows were captured and aggregated by an aggregator TAP
- Or the flows were captured but not aggregated by a non-aggregating TAP

1.5.2.1 Capture mode with an aggregator TAP

In this situation, the GCap retrieves the flow aggregated by the TAP on a single `monx` capture interface.
This solution is preferable because it requires the least amount of GCap resources for the same flow.

1.5.2.2 Capture mode with a non-aggregating TAP: GCap mode with aggregation ("cluster")

This functionality is necessary if the Test Access Port (TAP) present in the architecture does not provide the interface aggregation functionality.
A **qualified TAP** is at least a passive or non-intelligent (simple) TAP. This means that it does not require its own power supply and does not actively interact with other components. Most passive TAPs do not have an embedded configuration.

Connection between TAP and GCap

Unlike network interfaces where traffic is both TX (emission) and RX (reception), capture interfaces are unidirectional. Therefore, they can only receive flow, hence the following connection.
Each physical fibre link handles two links:

- An uplink, i.e. a TX link
- A downlink, i.e. an RX link

The TAP (without aggregation) is connected to the network via 2 physical links called `commutateur X` and `commutateur Y`.
The `commutateur X` link connects the switch and the **X** input TAP and enables duplicating half the network flow.
The TX link is:

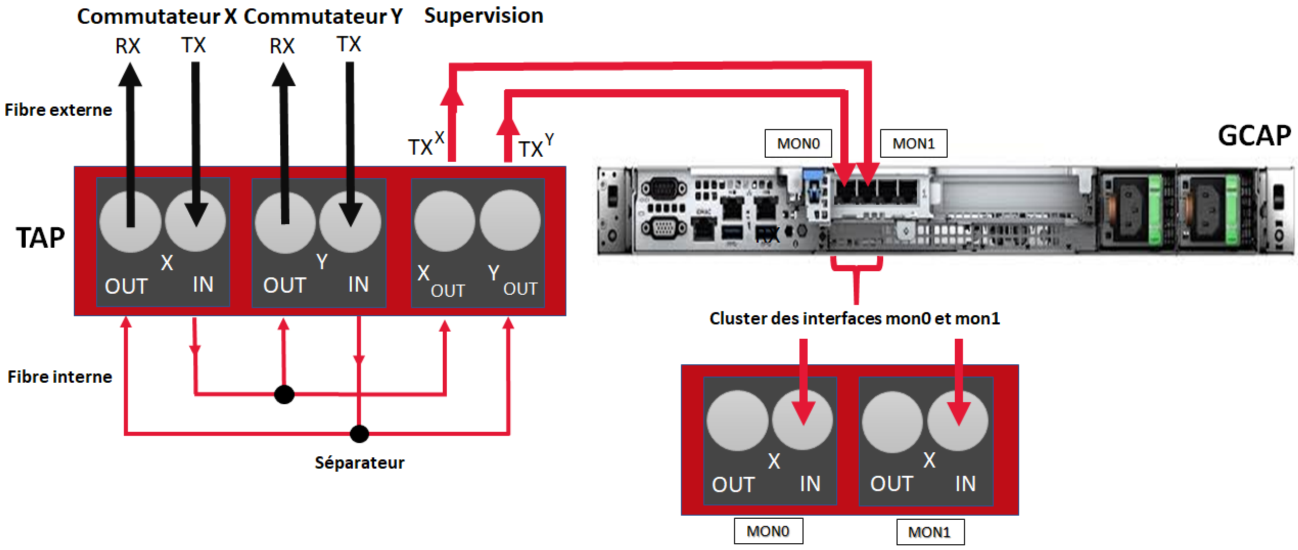
- Connected to **IN** of the **X** connector
- The flow of the TX link is copied to **OUT** of the **Y** connector: this is connected to the RX link of the `commutateur Y` physical link
- The flow from the TX link is also copied to the **Xout** link which is sent to the input port of the GCap (**IN** link of the `mon1` port)

The `commutateur Y` link connects the switch and the **Y** input TAP and enables duplicating the other half the network flow.
The TX link is:

- Connected to **IN** of the **Y** connector
- The flow of the TX link is copied to **OUT** of the **X** connector: this is connected to the RX link of the `commutateur X` physical link
- The flow from the TX link is also copied to the **Yout** link which is sent to the input port of the GCap (**IN** link of the `mon0` port)

Aggregation of interfaces (or clustering)

By defining an aggregation of two interfaces, the GCap aggregates these two flows into a single one, thus enabling a correct flow interpretation.
If the GCap has this functionality, this is not neutral in terms of resources allocated to this processing, hence the configuration with an aggregator TAP should be preferred.



1.5.2.3 Using and configuring interface aggregation

To implement interface aggregation, refer to the *Procedure for managing capture interface aggregation*.

1.5.3 Transferring rules between GCenter and GCap: single-tenant vs. multi-tenant

For more information, please refer to the paragraph *Capture and monitoring interfaces: single-tenant vs multi-tenant*.

1.6 Redundant GCaps: high availability

1.6.1 Introduction

High Availability enables two GCaps to be installed in redundancy so that the flows captured are not lost in the event of a GCap failure or shutdown.
To implement high availability, two GCaps must be installed on a network that communicate with a single GCenter. In the event of a failure on one of the two GCaps, the other one takes over to allow the service to continue to function while it is being repaired.

1.6.2 How high availability works

1.6.2.1 Prerequisites

Configuration must be identical on both GCaps, otherwise the messages exchanged will not be considered valid.

1.6.2.2 Basic assumption

A GCap can be either `leader` or `follower`.
Only the GCap `leader` can send eve logs and files to the GCenter.
The GCap `follower` stores the eve logs and files on its file system.

Note:

Retention time is 1 hour for the GCap *follower*.

When a GCap becomes the `leader`, it sends all the eve logs and files it stored on its file system.
There is no preemption mechanism: if a GCap is the `leader`, it will remain so as long as its status is `healthy`.

1.6.2.3 Electing the leader

The GCaps communicate with each other, electing the GCap `leader` and the GCap `follower`.
Electing the `leader` GCap is the one with the lowest ID: partly linked to the start date.

> **Avertissement:**
>
> If the GCaps fail to communicate, then they both become `leader'.
> If this happens, the data is duplicated on the system.

> **Note:**
>
> This behaviour is normal because a GCap cannot stop operating without the certainty that another GCap is `leader`.

1.6.2.4 Failure of a GCap

If the `leader` GCap fails, the `follower` GCap automatically becomes the `leader` GCap.

When the failed GCap becomes functional again, it was and remains the `follower`.

If the `follower` GCap fails, the `leader` GCap remains the `leader`.

When the failed GCap becomes functional again, it becomes the `follower`.

1.6.3 Using and configuring high availability

For more information, please refer to *Redundant GCaps: high availability*.

Chapter 2

Operation

2.1 GCap

2.1.1 GCap functions

The functions of the GCap include:

- Connecting to the TAP and retrieving duplicate packets from the network flow seen by the TAP,
- Rebuilding the files from the corresponding packets using a detection engine, also referred to as Sigflow,
- Intrusion detection (vulnerabilities...) is performed by several detection engines:
  - The first is the Sigflow engine. It is located in the GCap
  - The others are located in GCenter. It recovers the network flow sent by the GCap to perform this analysis:
    - The second is the Codebreaker engine
    - The third is the Malcore engine
    - The fourth is the Retroact engine
- The transmission of files, codes and events to GCenter,
- Communication between GCap and GCenter including receiving configuration files, rulesets, and the like.

2.1.2 The Sigflow engine

Sigflow performs:

- The recovery of network flows entering the Gcap via the `monx` capture interfaces,
- Intrusion detection, statistical analysis of network flows to reduce the number of false positives and identify possible protocol malformations, SQL injection attempts, and so on.
- The creation of alerts or log files

The use of rules enables the Sigflow engine to define what to monitor, hence to raise alerts.
For more information, please refer to the table *Managing the detection engine*.

2.1.2.1 Filtering of the captured flow

Certain parts of the captured flow cannot be detected or reconstructed: for example, encrypted flows.
If nothing is done, the system will monopolise resources to achieve a result known in advance.
To avoid this, it is possible to create rules to filter the flow to be captured.

Note:

To **display** the packet filtering rules, use the `show advanced-configuration packet-filtering` command.
To **specify** the packet filtering rules, use the `set advanced-configuration packet-filtering` command.

2.1.2.2 Configuration rules

Note:

To **display** the packet filtering rules, use the `show advanced-configuration local-rules` command.
To **specify** the local rules, use the `set advanced-configuration local-rules` command.

2.1.2.2.1 Sigflow rules for detection

The Sigflow configuration rules are defined:

- In GCenter and transferred from GCenter with access via the `show config-files rules-scirius` command
- Or locally on the GCap with access via the `{show,set} advanced-configuration local-rules` command

2.1.2.3 Sigflow configuration rules for rebuilding files

The Sigflow configuration rules are defined:

- In GCenter and transferred from GCenter with access via the `show config-files rules-files` command
- Or locally on the GCap with access via the `{show,set} advanced-configuration local-rules` command

2.1.2.4 Sigflow configuration rules for managing the thresholds for the raising alarms

The Sigflow configuration rules are defined:

- In GCenter with access via the `show config-files threshold` command
- Or locally on the GCap with access via the `{show,set} advanced-configuration local-rules` command

2.1.3 Counters of GCap activity

In order to view this information, the *`show eve-stats'* command enables the following counters to be viewed:

- Counter `Alerts` - Number of Sigflow alerts found
- The counters `Files` - Files extracted by Sigflow
- The counters `Codebreaker samples` - Files analysed by Codebreaker
- Counters `Protocols` - List of protocols seen by Sigflow
- Counters `Detection Engine Stats` - Sigflow statistics (*monitoring-engine*)

For more information, please refer to the table *Monitoring the detection engine*.

2.2 GCap configuration

2.2.1 Configuring a GCap and its Sigflow engine

To analyse the captured flow, the following steps must be taken:

- Synchronise the date and time of the GCap on GCenter: see *Overview of the date and time management*
- Manage `gcp0` and `gcp1` interfaces: see *Overview of managing `gcp0` and `gcp1` network interfaces*
- Managing Capture Interfaces: see *Overview of managing capture and monitoring interfaces*
- Manage single-tenant vs. multi-tenant configuration of `monx` interfaces: see *Capture and Monitoring Interfaces: Single-tenant vs. multi-tenant*
- Managing the aggregation of capture interfaces: see *Capture and monitoring interfaces: aggregation*
- Pairing the GCap with GCenter
  A GCap must be paired with a GCenter.
  Data exchange only starts when the VPN tunnel (IPsec) is established between the two devices.
- Activation of the Sigflow monitor engine (by default it is deactivated)

2.2.2 Overview of date and time management

When connecting for the first time, the date and time of the GCap and GCenter must be identical in order to set up the IPsec tunnel.

2.2.2.1 CLI commands

Displaying the current date and time is accomplished with the *show datetime* command in the CLI. Modifying the current date and time is accomplished with the *set datetime* command in the CLI.

2.2.2.2 Use case procedures

For implementation, refer to the *Procedure for modifying the GCap date and time*. Thereafter, the GCap date and time are synchronised with the GCenter date and time after the IPsec tunnel is established.

2.2.3 Overview of the management of the gcp0 and gcp1 network interfaces.

There are two management interfaces. They are called `gcp0` and `gcp1` respectively.
These interfaces perform the following functions:

- Function 1: secure communication between the probe and GCenter through an IPSEC tunnel in order to:
  - Escalate information such as files, alerts, metadata, and so on, derived from analysing the monitored flows
  - Report information on the health of the probe to GCenter
  - Control the probe - analysis rules, signatures, and so on.
- Function 2: remote administration through the SSH protocol with access:
  - To the probe's command line interface (CLI)
  - To the graphical setup/configuration menu (deprecated)

2.2.3.1 CLI commands

Managing the network interfaces is done using the CLI commands listed in the *Manage the network* table.

2.2.3.2 View or configure

To view or configure the network interfaces, refer to the *Procedure for managing the network settings of gcp0 and gcp1 interfaces*.

2.2.3.2.1 Single interface configuration.

In **single-interface configuration**, function 1 and function 2 are supported by the interface `gcp0` only. To toggle from dual-interface to single-interface configuration, refer to the *Procedure for switching to single-interface configuration*.

2.2.3.2.2 Dual-interface configuration

In **dual-interface configuration**:

- Function 1 is handled by interface `gcp0`
- Function 2 is handled by interface `gcp1`

Important:

This dual-interface configuration is mandatory if using the **MPL mode** on the GCenter.

The aim of this situation is to ensure that the management flow and the interconnection flow between the GCap and GCenter are separated from each other.

Note:

It is not possible to invert the 2 network interfaces.

To toggle from single-interface to dual-interface configuration, refer to the *Procedure for switching to dual-interface configuration*.

2.2.4 Overview of managing the capture and monitoring interfaces

The capture interfaces on GCap are, by default, four in number.
These interfaces receive the flows from the TAPs on the specified interfaces:

- `mon0` for the first TAP
- `mon1` for the second TAP
- `mon2` for the third TAP
- `mon3` for the fourth TAP

For more information regarding the capture interfaces, refer to the *Capture and monitoring interfaces section*.

Note:

The number of capture interfaces varies depending on the specifications of each model.

In some special cases, it is possible to use GCaps with eight interfaces instead of four.
In addition, there is also a monvirt virtual capture interface enabling `.pcap` file replay directly on the GCap.
In order for the GCap to capture the flow, one or more interfaces must be activated.

2.2.4.1 CLI commands

Managing the capture interfaces is done using the CLI commands listed in the *Manage the network* table.

2.2.4.2 Use case procedures

To view or configure the capture interfaces, refer to the *Procedure for managing capture interface settings* `monx`.

2.2.5 Capture and monitoring interfaces: single-tenant vs. multi-tenant

2.2.5.1 GCap detection engine and rules

SIGFLOW is the name of the GCap detection engine configured:

- By a set of rules (RULESET) defined on the GCenter
- By locally defined rules, therefore not known to the GCenter

These rules must describe the characteristics of the attacks to be detected as well as being optimised to reduce false positives.
The set of rules is composed of signatures grouped by categories that were provided by sources.
This compilation is done by the administrator on the GCenter. Therefore, it can be configured differently depending on the number of GCap and their specifications.

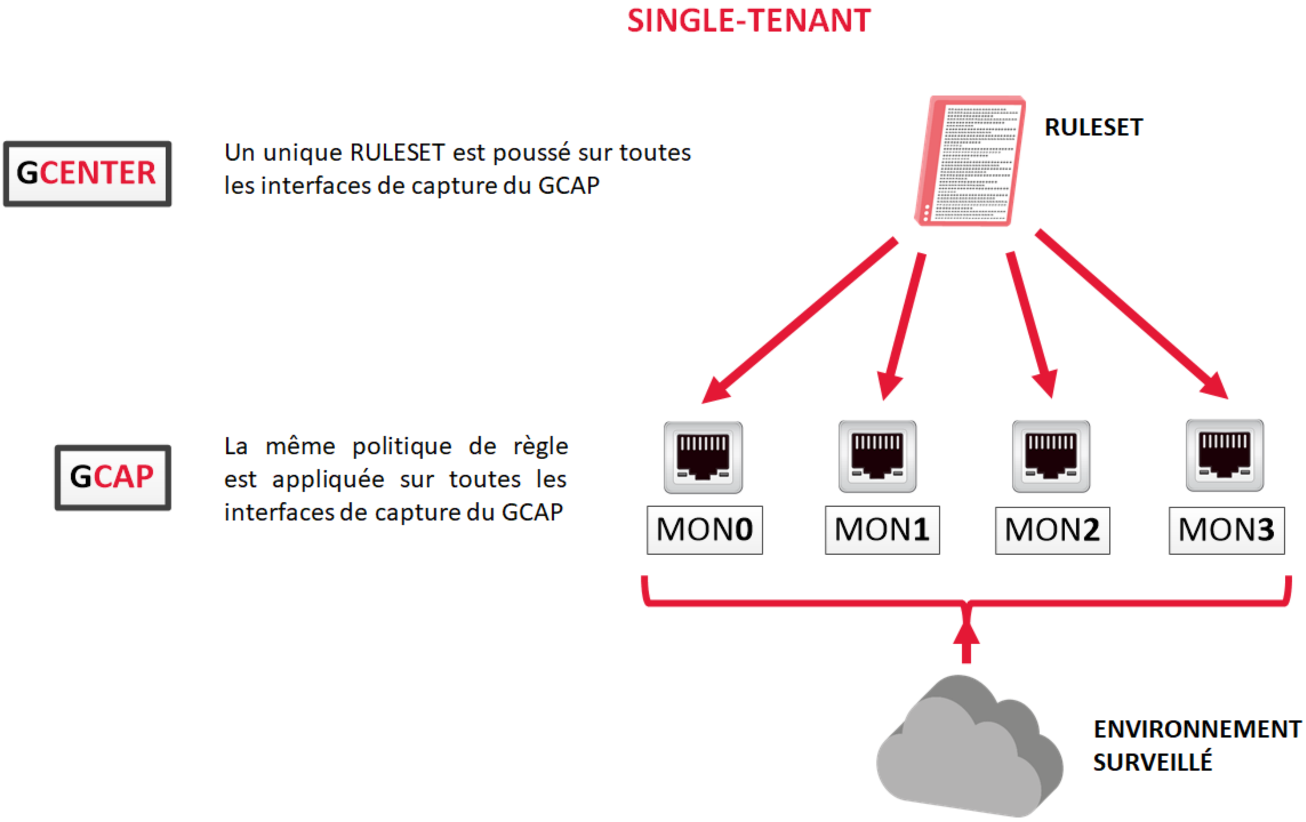
2.2.5.1.1 CLI commands

The ability to view and create local rules is handled differently depending on the configuration.
For more information on rules, see the table *Managing the detection engine (advanced functions)*.

2.2.5.2 Transferring the rule set in single-tenant mode

2.2.5.2.1 Single-tenant principle

Once configured on GCenter, a single set of rules (RULESET) is sent to the GCap detection engine.
The GCap detection engine applies this ruleset to all capture interfaces: this is the single-tenant configuration.


SINGLE-TENANT
GCENTER
Un unique RULESET est poussé sur toutes les interfaces de capture du GCAP
RULESET
GCAP
La même politique de règle est appliquée sur toutes les interfaces de capture du GCAP
MON0
MON1
MON2
MON3
ENVIRONNEMENT SURVEILLÉ

Sigflow rules in single-tenant

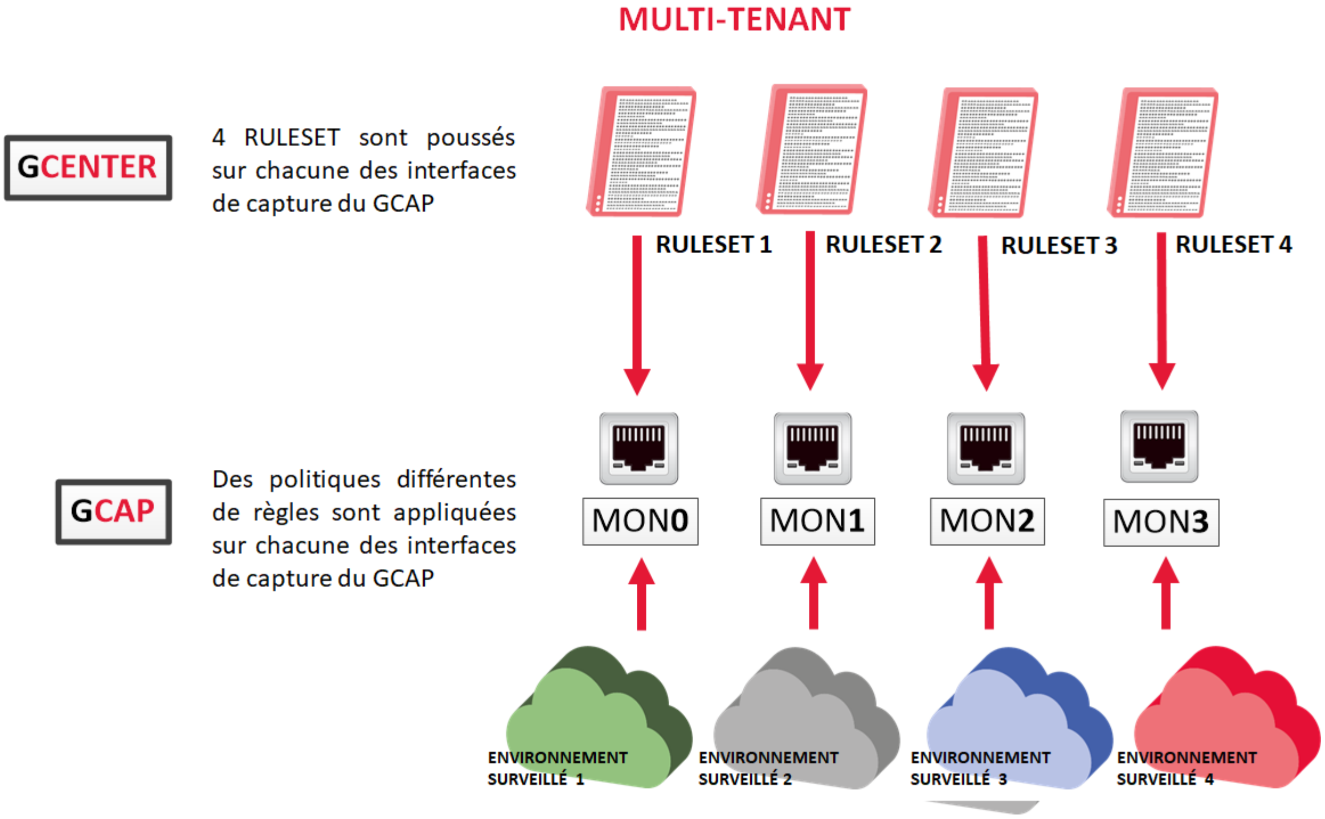
2.2.5.2.2 Configuring the single-tenant mode

In the GCenter web interface, in the `SIGFLOW - GCaps Profiles > Detection rulesets` part, the default option is single-tenant.

2.2.5.3 Transferring the SIGFLOW rule set in multi-tenant mode

2.2.5.3.1 Multi-tenant principle

Once configured on GCenter, it is possible to define a different set of SIGFLOW rules for each of the capture interfaces.
Then each of these rulesets will be applied to its capture interface: this is the **multi-tenant** configuration.


MULTI-TENANT
GCENTER
4 RULESET sont poussés sur chacune des interfaces de capture du GCAP
RULESET 1
RULESET 2
RULESET 3
RULESET 4
GCAP
Des politiques différentes de règles sont appliquées sur chacune des interfaces de capture du GCAP
MON0
MON1
MON2
MON3
ENVIRONNEMENT SURVEILLÉ 1
ENVIRONNEMENT SURVEILLÉ 2
ENVIRONNEMENT SURVEILLÉ 3
ENVIRONNEMENT SURVEILLÉ 4

Sigflow rules in multi-tenant

In contrast to single-tenant, multi-tenant will enable optimising resources and costs while simplifying the process of managing detection rules per environment.
The flexibility of the architecture enables efficient refinement of detection rules, easier isolation of threats, and customisation of capture.

2.2.5.3.2 Configuring the multi-tenant mode

In the GCenter web interface, in the `SIGFLOW - GCaps Profiles > Detection rulesets` part, the default option is single-tenant.
It is also possible to choose two other options:

- 'Multi-tenant by interface' or
- 'Multi-tenant by vlan'

In the event one of these options is selected, it offers the possibility to assign different SIGFLOW rulesets for:

- Each of the GCap interfaces or
- For the various vlan's...

...and thus have a different supervision on various networks.

It is strongly advisable to optimise the SIGFLOW ruleset in advance before choosing this configuration option.
The rules must be adapted to the monitored environment.
This version of GCap enables compatibility with GCenter.

2.2.6 Capture and monitoring interfaces: aggregation

2.2.6.1 Aggregation principle ("cluster")

For more information, refer to *Capture and monitoring interfaces between TAP and GCap*.

2.2.6.2 CLI commands

Displaying the current aggregation is achieved with the *show clusters* command.
Configuring the aggregation is done with the *set clusters* command.

2.2.6.3 Use case procedures

For implementation, refer to the *Procedure for managing capture interface aggregation*.

2.2.6.4 Aggregation configuration

Aggregation creation is done via the GCap Command Prompt (CLI). An aggregation, once created, must be activated.

2.2.6.5 Impact on other functionalities

The aggregation functionality of the capture interfaces on the GCap leads to a degradation of some related functions:

- Maximum Transmission Unit (MTU): the maximum size of a packet that can be transmitted at one time without fragmentation.
  The *MTU* uses the largest value of the interfaces making up the aggregation.
- Static rules for filtering flows captured by monitoring interface: XDP (eXpress Data Path) filter function.
  XDP Filter: XDP filtering is not applied by default to the aggregation created but rather to the interfaces that comprise it.
- File rebuilding rules.
  Rebuild rule: When enabling interface aggregation and multi-tenant detection, file rebuild rules are not generated.

2.2.7 Sigflow detection engine

To analyse the captured flow, the following steps must be taken:

- Activate one or more capture interfaces on the GCap
- Pairing the GCap with GCenter
- Activation of the Sigflow detection engine, by default it is deactivated

2.2.7.1 Activate one or more capture and monitoring interfaces on the GCap

2.2.7.1.1 CLI commands

Managing the capture interfaces is done using the CLI commands listed in the *Manage the network* table.

2.2.7.1.2 Use case procedures

To view or configure the capture interfaces, refer to the *Procedure for managing capture interface settings* `monx`.

2.2.7.2 Aggregating capture and monitoring interfaces `monx`

For more information on this aggregation, see the paragraph *Capture and monitoring interfaces `monx` between TAP and GCap: aggregation capability*.
For more information on how to configure this aggregation, refer to the paragraph Configuring the capture and monitoring interfaces: *aggregation*.

2.2.7.3 Pairing the GCap with GCenter

Once the network configuration is done, it is necessary to pair the GCap with GCenter.
For more information on pairing, refer to the procedure *Pairing between a GCap and GCenter*.

2.2.7.4 Activating the Sigflow monitor engine

By default the GCap monitor engine is disabled.

2.2.7.4.1 Checking the status of the Sigflow monitor engine (activating procedure)

The status of the monitor engine can be checked with the command `show status`.

2.2.7.4.2 Starting the Sigflow analysis engine

It is essential to start the Sigflow monitor engine (detection engine).
The flow capture only takes place after this start.
To do this:

- Enter the *monitoring-engine start* command
- Validate

```
(gcap-cli) monitoring-engine start
```

The system displays the following message indicating that the engine started.

```
Starting Detection Engine...
This operation may take a while... Please wait.
Detection Engine has been successfully started.
```

Once the monitor engine is activated, the configuration possibilities of the GCap probe change. Some of them cannot be configured while the engine is running.

> **Note:**
>
> The *eve-stats* command in the *show* subgroup enables displaying the Sigflow (*monitoring-engine*) statistics.

2.2.7.4.3 Grace period

The grace period is the sum of:

- The maximum starting time
- The maximum stopping time

In order to be able to load the rules of the detection engine before starting the engine, the engine cannot start until a certain time called maximum start time or start-up grace period (start-timeout).

- The current value is displayed using the *show monitoring-engine start-timeout* command.
- If the number of rules loaded by the analysis engine is large then the maximum start time must be changed via the *set monitoring-engine start-timeout* command.

Similarly, there is the maximum stopping time or grace period when the engine shuts down (stop-timeout).

- The current value is displayed via the *show monitoring-engine stop-timeout* command.
- The modification of the current value is done via the *set monitoring-engine stop-timeout* command.

If the number of rules loaded by the analysis engine is significant then the maximum start-up time must be modified via the *CLI*.

2.2.7.5 Deactivating the Sigflow monitor engine

2.2.7.5.1 Checking the status of the Sigflow monitor engine (deactivating procedure)

The status of the engine can be checked with the `show status` command.

2.2.7.5.2 Stopping the Sigflow monitor engine

In the same way, stopping is carried out with the *monitoring-engine stop* command:

```
(gcap-cli) monitoring-engine stop
```

The system displays the following message indicating that the engine started.

```
Stopping Detection Engine...
This operation may take a while... Please wait.
Detection Engine has been successfully stopped.
```

2.2.7.6 Compatibility mode

The compatibility mode between the GCap and GCenter must be specified via the *CLI*.

2.2.7.7 MTU

The Maximum Transfer Unit (MTU) of each GCap capture interface can be adjusted via the CLI.
Indeed, the maximum packet size to be captured at one time on an interface is configurable.

2.2.7.7.1 Display of the current MTU value

The MTU value can be displayed using the *show advanced-configuration mtu* command:

```
(gcap-cli) show advanced-configuration mtu

Current Monitoring Network MTU configuration:
        - mon0: 1500
        - monvirt: 1500
```

The administrator can change the MTU's value in bytes of the GCap capture interfaces.
This setting must be between 1280 and 9000 bytes.

Note:

Note that Load Balancing and XDP Filtering features are not supported if the MTU > 3000.

2.2.7.7.2 Changing the current MTU value

Regarding the modification of the MTU, this is done with the *set advanced-configuration mtu* command followed by the parameters:

- name of the interface, for example mon0
- value, for example 1300

> **Note:**
>
> To change the MTU of the `mon0` interface to 1300 :
> - Enter the *set advanced-configuration mtu mon0 1300* command
> - Validate

```
(gcap-cli) set advanced-configuration mtu mon0 2500
```

The system displays the parameter update information.

```
Updating Monitoring Network MTU configuration to:
        - mon0: 2500
```

2.2.7.8 Rebuilding files

Rebuilding files occurs on the GCap thanks to its monitor engine (Sigflow).
These files are rebuilt under certain conditions that can be set from GCenter.
These conditions include the following:

- The size of the observed file
- The type of file observed, based either on the extension or on the filemagic

In addition, file reconstruction is only possible on certain protocols, the list of which varies according to the different GCap versions.
Here is the list of protocols supported by the GCap:

- HTTP
- SMTP

Other protocols are available from GCenter. Please refer to the GCenter documentation for more information.

> **Note:**
>
> Namely, the protocols on which it is possible to rebuild depends on the GCap and not the GCenter.
> If the GCenter configuration instructs the GCap to rebuild a certain file type but the GCap is not capable of doing so, the rebuild will not take place.

The administrator can add a local rule from the CLI with the *local-rules* command if necessary.
An example of rule syntax for these protocols is as follows:

```
alert ftp-data any any -> any any (msg:"[ Message regle FTP ] FTP filestore all"; filestore;␣
↪ftpdata_command:retr; sid:13371340; rev:1;)

alert smb any any -> any any (msg:"[ Message regle SMB ] SMB filestore all"; filestore;␣
↪ftpdata_command:retr; sid:13371341; rev:1;)
```

2.3 Redundant GCaps: high availability

2.3.1 Introduction and operation

For more information, refer to the paragraph *GCaps in redundancy: high availability*.

2.3.2 Commands in the CLI

High availability is managed by using the CLI commands that are listed in the *Configure GCap* table.

2.3.3 Use case procedures

For implementation, refer to the *High availability management procedure*.

Chapter 3

Characteristics

3.1 Mechanical characteristics of GCap

REFERENCE	DIMENSIONS (H x L x P)	RACKAGE	WEIGHT (KG)
GCAP1010HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP1020HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP1050HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP1100HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP1200HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP1400HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP2200HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP2600HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP2800HWr2	42.8 x 482 x 808.5 mm	1 U	21.9
GCAP5400HWr2	86.8 x 434 x 836 mm	2 U	36.6
GCAP5600HWr2	86.8 x 434 x 836 mm	2 U	36.6
GCAP5800HWr2	86.8 x 434 x 836 mm	2 U	36.6

3.2 Electrical characteristics of GCap

REFERENCE	LOCAL STOCKAGE	CAPTURE PORTS	EXTENSION CAPTURE PORTS	ELECTRIC POWER
GCAP1010HWr2	256GB	4 x RJ45	N/A	2 x 750W
GCAP1020HWr2	256GB	4 x RJ45	N/A	2 x 750W
GCAP1050HWr2	256GB	4 x RJ45	N/A	2 x 750W
GCAP1100HWr2	2 x 600GB RAID1	1 x SFP	N/A	2 x 750W
GCAP1200HWr2	2 x 600GB RAID1	2 x SFP	N/A	2 x 750W
GCAP1400HWr2	2 x 600GB RAID1	4 x SFP	N/A	2 x 750W
GCAP2200HWr2	4 x 600GB RAID5	4 x SFP	4 x SFP	2 x 750W
GCAP2600HWr2	4 x 600GB RAID5	4 x SFP	4 x SFP	2 x 750W
GCAP2800HWr2	4 x 600GB RAID5	4 x SFP	4 x SFP	2 x 750W
GCAP5400HWr2	8 x 600GB RAID5	4 x SFP+	4 x SFP+	2 x 1100W
GCAP5600HWr2	8 x 600GB RAID5	4 x SFP+	4 x SFP+	2 x 1100W
GCAP5800HWr2	8 x 600GB RAID5	4 x SFP+	4 x SFP+	2 x 1100W

3.3 Functional characteristics of the GCaps

3.3.1 Functional characteristics

REFERENCE	MAX THROUGHPUT	NUMBER OF FILES RECONSTRUCTED MAX PER S	NUMBER OF SESSIONS MAX	NUMBER OF MAX SESSIONS PER	EPS MAX
GCAP1010HWr2	10 MBPS	1	1000	20	100
GCAP1020HWr2	20 MBPS	2	2000	50	100
GCAP1050HWr2	50 MBPS	2	5000	100	100
GCAP1100HWr2	100 MBPS	5	20000	1000	200
GCAP1200HWr2	200 MBPS	10	40000	2000	300
GCAP1400HWr2	400 MBPS	10	40000	2000	400
GCAP2200HWr2	1 GBPS	20	150 000	5 000	2000
GCAP2600HWr2	2 GBPS	30	200 000	10 000	3000
GCAP2800HWr2	4 GBPS	30	250 000	20 000	4000
GCAP5400HWr2	10 GBPS	50	500 000	50 000	8000
GCAP5600HWr2	20 GBPS	50	750 000	75 000	8000
GCAP5800HWr2	40 GBPS	50	1 000 000	100 000	8000

3.3.2 List of protocols that can be selected for analysis

Protocol detection consists of two parts:

- **parsing**:
  - It enables SIGFLOW signature detection for a given protocol
  - If parsing is enabled for a protocol then the flow identified by a signature raises an alert
  - If parsing is disabled for a protocol then no alert is raised
- **logging**:
  - It enables generating metadata for a given protocol
  - If logging is enabled for a protocol then the observed flow will generate metadata
  - If logging is disabled for a protocol then no metadata is generated

For each interface, it is possible to:

- Enable parsing and logging
- Enable parsing only
- Disable parsing and logging

PROTOCOLE	PARSING	LOGGING
DCE-RPC	supported	supported
DHCP	supported	supported
DNP3	supported	supported
DNS_udp	supported	supported
DNS_tcp	supported	supported
ENIP	supported	not supported
FTP	supported	supported
HTTP	supported	supported
HTTP2	supported	supported
IKEv2	supported	supported
IMAP	parsing detection only	not supported
Kerberos (KRB5)	supported	supported
MODBUS	supported	not supported
MQTT	supported	supported
NETFLOW	not supported	supported
NFS	supported	supported
NTP	supported	not supported
RDP	supported	supported
RFB	supported	supported
SIP	supported	supported
SMB	supported	supported
SMTP	supported	supported
SNMP	supported	supported
SHH	supported	supported
TFTP	supported	supported
TLS	supported	supported

These options depend on the Gcenter version, thus on the selected compatibility.
For more information, please refer to the GCenter documentation.

3.3.3 List of selectable protocols for file reconstruction

PROTOCOLE	SUPPORTED
FTP	supported
HTTP	supported
HTTP2	supported
NFS	supported
SMB	supported
SMTP	supported

These options depend on the Gcenter version, thus on the selected compatibility.
For more information, please refer to the GCenter documentation.

Chapter 4

The accounts

4.1 List of accounts

Remote or local access to the GCap administration interface is protected by a login password.
Three generic accounts are defined with different rights levels:

Account...	account for a...
gview	operator
gviewadm	manager
setup	system administrator

4.2 Related principles

4.2.1 Authentication mode

A user can be authenticated in two different ways:

- Username/password
- SSH key

Important:

Simultaneously connecting several accounts is not possible.

4.2.2 Password management

The current account manages its own password and potentially other accounts as well.

Details are provided in the table below:

User	can change the password		
	setup	gviewadm	gview
setup	X	X	X
gviewadm		X	X
gview			X

The *show passwords* command enables displaying the list of users managed by the current level.
The *set passwords* command enables changing the password managed by the current level.

4.2.3 Password management policy

The passwords entered must comply with the password management policy.
The default policy is as follows:

Criteria	Default value
Number of different characters for a password to be considered as different	2
Minimum password length	12 characters
Presence of at least one lower case letter	yes
Presence of at least one lower case letter	yes
Presence of at least one capital letter	yes
Presence of at least one digit (0 to 9)	yes
Presence of at least one symbol (i.e. neither a number nor a letter)	yes

This policy is:

- Viewable via the *show password-policy* command
- Modifiable via the *set password-policy* command

4.2.4 SSH key

Authenticating SSH connections to administer GCap can be done via an SSH key. All SSH keys authorised for an account and the list of different types of encryption are defined via the *set ssh-keys* command. This mode is to be preferred to the user name/password pair.
Indeed, it enables defining a key per employee, thus ensuring traceability of connections and accountability of actions.

4.2.5 Rights associated with each account

The rights assigned to each account are listed in the presentation of each account.

4.3 gview profile

To log in to the **gview** account, the default password is: default

Note:

It is necessary to change the password the first time you log in. It should be kept in a safe place, for example, with the **GCap** encryption keys.

From the **gview** account, it will be possible:

- To access the commands of all `show` for:
  - Viewing alert logs (`show alerts`)
  - Monitoring CPU usage (`show cpus`)
  - Displaying the keyboard layout (`show keymap`)
  - Displaying the list of users managed by the current level (`show passwords`)
  - Displaying the current GCap status (`show status`)
  - Displaying the Sigflow configuration as well as the rules transmitted by the GCenter (`show config-files`)
  - Displaying the statistics of the Sigflow detection engine (`show eve-stats`)
  - Displaying the various log files of the GCap (`show logs`)
  - Displaying the connection mode - graphical GUI or command line CLI (`show setup-mode`)
  - Password policy for the accounts (`show password-policy`)
- To access the commands of all `set` for:
  - Changing the password of the current user and the lower level (`set passwords`)
  - Changing the keyboard configuration (`set keymap`)
  - Changing the login mode for the current user and the lower level - graphical GUI or command line CLI (`set setup-mode`)
  - Changing SSH keys for the current user and the lower level (`set ssh-keys`)

This account corresponds to an operator profile, member of a detection service in charge of operating the service.

> **Note:**
>
> Commands in the **gview** account are also found in the other **gviewadm** and **setup** accounts.

4.4 gviewadm profile

To log in to the **gviewadm** account, the default password is: default

> **Note:**
>
> It is necessary to change the password the first time you log in. It should be kept in a safe place, for example, with the **GCap** encryption keys.

In addition to the common functions of **gview**, the **gviewadm** account has the following supplementary functions:

- To access the commands of all `show` for:
  - Displaying the statistics and health information of the GCap (`show health`)
- Accessing the commands of all `services` to manage the services:
  - Viewing the status of a service or services (`services status`)
  - Starting a service (`services start`)
  - Stopping a service (`services stop`)
  - Displaying GCap file retention periods (`services show`)
- Start, stop and view the status of the detection engine (`monitoring-engine`)

This account represents an administrator profile, a member of the Detection Service with privileged rights enabling them to ensure the correct operation of the Detection Service devices.

> **Note:**
>
> Commands present in the **gviewadm** account are also found in the **setup** account.

4.5 Setup profile

To log in to the **setup** account, the default password is: default

> **Note:**
>
> It is necessary to change the password the first time you log in. It should be kept in a safe place, for example, with the **GCap** encryption keys.

In addition to the common functions of **gviewadm**, the **setup** account has the following supplementary functions:

- Access the commands of set `show` to display:
  - Information about the available capture interfaces (`show interfaces`)
  - The aggregations of capture and monitoring interfaces `mon` and their configurations (`show clusters`)
  - The compatibility mode used to interact with the GCenter (`show compatibility-mode`)
  - The date and time of the GCap (`show datetime`)
  - The protection system policy (`show bruteforce-protection`)
  - The inactivity time before logging out of a user session (`show session-timeout`)
  - The IP address of the GCenter with which the GCap is paired (`show gcenter-ip`)
  - The advanced options of the detection engine configuration (`show monitoring-engine`)
  - The GCap information requested by technical support (`show tech-support`)
- Access the advanced commands of the `show advanced-configuration` set to display:
  - The number of CPUs dedicated to the Sigflow detection engine (`show advanced-configuration cpu-config`)
  - The static filtering rules of the flow (`show advanced-configuration packet-filtering`)
  - The high availability configuration (`show advanced-configuration high-availability`)
  - The MTU value of the enabled capture interfaces (`show advanced-configuration mtu`)
  - The load balancing configuration coming from the `monx` capture interface listed to the CPUs (`show advanced-configuration load-balancing`)
  - The local Sigflow rules according to the configured tenant (`show advanced-configuration local-rules`)
  - The replacement name of the interfaces (`show advanced-configuration interface-names`)
- Access the commands of the `set` set to:
  - Manage the protection system against brute force attacks (`set bruteforce-protection`)
  - Configure the aggregation on the GCap capture interfaces (`set clusters`)
  - Change the compatibility mode used to interact with the GCenter (`set compatibility-mode`)
  - Adjust the date and time (`set datetime`)
  - Specify the IP address of the GCenter to which the GCap will be paired (`set gcenter-ip`)
  - Administer network capture interfaces (`set interfaces`)
  - Change the keyboard configuration (`set keymap`)
  - Apply advanced configuration for the GCap sensor detection engine (`set monitoring-engine`)
  - Change the network configuration (`set network-config`)
  - Set password policy for accounts (`set password-policy`)
  - Configure inactivity time before logging out (`set session-timeout`)
- Access the advanced commands of the `set advanced-configuration` set to:
  - Modify the number of CPUs dedicated to the Sigflow detection engine (`set advanced-configuration cpu-config`)
  - Modify the high availability configuration (`set advanced-configuration high-availability`)
  - Define an advanced load balancing configuration of the captured flows (`set advanced-configuration load-balancing`)
  - Modify the local Sigflow rules according to the configured tenant (`set advanced-configuration local-rules`)
  - Modify the MTU value of enabled capture interfaces (`set advanced-configuration mtu`)
  - Specify the static filtering rules for the flow (`set advanced-configuration packet-filtering`)
  - Detect/name the GCap interfaces (`set advanced-configuration rescan-interfaces`)
- Access the `system` set commands to manage the server:
  - Restart the GCap (`system restart`)

- Shut down the GCap (`system shutdown`)
- Stop a service (`system reload-drivers`)
- Reload network card drivers (`services show`)
- Reset `gview`, `gviewadm` and `setup` account lockout after unsuccessful authentication attempts (`system unlock`)

This account represents an administrator profile, a member of the detection service with privileged rights enabling them to ensure the correct operation of the detection service devices.

4.6 List of functions by level and by theme

4.6.1 Configuring the GCap

Table1: Configuring the GCap

Function by level	setup	gviewadm	gview
Keyboard configuration : display	show keymap	show keymap	show keymap
Keyboard configuration : modify	set keymap	set keymap	set keymap
Interface (GUI or CLI) for the next connection : Display	show setup-mode	show setup-mode	show setup-mode
Interface (GUI or CLI) for the next connection : change the mode	set setup-mode	set setup-mode	set setup-mode
Date and time : display	show datetime	N/A	N/A
Date and time : modify	set datetime	N/A	N/A
Colours: enable or disable for the current session	colour	colour	colour
Compatibility mode with the GCenter : display	show compatibility-mode	N/A	N/A
Compatibility mode with the GCenter : modify	set compatibility-mode	N/A	N/A
Services : display the file retention periods	services show retention-periods	services show retention-periods	N/A
Services : start a service (to be defined)	services start +service to be defined	services start +service to be defined	N/A
Services : stop a service (to be defined)	services stop +service to be defined	services stop +service to be defined	N/A
Services : view the status	services status +service to be defined	services status +service to be defined	N/A
high availability : view the status	show advanced-configuration high-availability	N/A	N/A
high availability : configure	set advanced-configuration high-availability	N/A	N/A
Pairing with the GCenter	pairing	N/A	N/A

4.6.2 Managing accounts

Table2: Managing accounts

Function per level	setup	gviewadm	gview
Authentication: display the list of users	show passwords	show passwords	show passwords
Authentication: change passwords	set passwords	set passwords	set passwords
Authentication: change SSH keys	set ssh-keys	set ssh-keys	set ssh-keys
Authentication: display the password policy	show password-policy	show password-policy	N/A
Authentication: unlocking blocked accounts	system unlock	N/A	N/A
Authentication: define a password policy	set password-policy	N/A	N/A
Authentication: display policy for protecting against brute force attacks	show bruteforce-protection	N/A	N/A
Authentication: modify the policy for protecting against brute force attacks	set bruteforce-protection	N/A	N/A
Session: display the duration of inactivity before disconnection	show session-timeout	N/A	N/A
Session: change the duration of inactivity before disconnection	set session-timeout	N/A	N/A

4.6.3 Managing the detection engine

Table3: Managing the detection engine

Function per level	setup	gviewadm	gview
Sigflow configuration: display the configuration and the rules	show config-files	show config-files	show config-files
Sigflow configuration: display advanced options	show monitoring-engine	N/A	N/A
Sigflow configuration: apply an advanced configuration	set monitoring-engine	N/A	N/A
Sigflow configuration: start the detection engine	monitoring-engine start	monitoring-engine start	N/A
Sigflow configuration: stop the detection engine	monitoring-engine stop	monitoring-engine stop	N/A
Sigflow configuration: display status	monitoring-engine status	monitoring-engine status	N/A
Traffic generation: replaying a pcap file	replay	replay	N/A

4.6.3.1 Managing the detection engine (advanced functions)

The advanced functions include:

- Resource allocation: modification of the distribution of CPUs reserved for the detection engine
- Capture interface load balancing: load balancing of captured flows per capture interface using load balancing methods (algorithm)
- Flow filtering: specification of static rules for filtering flows captured by capture interfaces
- Sigflow local rules: local modification in the GCap of the traffic monitoring rules performed by the Sigflow detection engine using the detection rules (local_all.rules file)

Table4: Managing the detection engine (advanced functions)

Function per level	setup	gviewadm	gview
Resource allocation: display the number of dedicated CPUs	show advanced-configuration cpu-config	N/A	N/A
Resource allocation: modify the number of dedicated CPUs	set advanced-configuration cpu-config	N/A	N/A
Load balancing monx capture interface - CPU: show the configuration	show advanced-configuration load-balancing	N/A	N/A
Load balancing monx capture interface - CPU: modify the configuration	set advanced-configuration load-balancing	N/A	N/A
Flow filtering: display static rules	show advanced-configuration packet-filtering	N/A	N/A
Flow filtering: specify the static rules	set advanced-configuration packet-filtering	N/A	N/A
Sigflow local rules: display	show advanced-configuration local-rules	N/A	N/A
Sigflow local rules: modify	set advanced-configuration local-rules	N/A	N/A

4.6.4 Managing network

Table5: Managing network

Function per level	setup	gviewadm	gview
Network configuration: consult the network configuration (IP addresses, name, domain...)	show network-config	N/A	N/A
Network configuration: change the configuration	set network-config	N/A	N/A
GCenter IP address: display the IP address of the GCenter with which the GCap is paired	show gcenter-ip	N/A	N/A
GCenter IP address: specify the IP address of the GCenter to which the GCap will be paired	set gcenter-ip	N/A	N/A
Detection interfaces: display the information	show interfaces	N/A	N/A
Detection interfaces: configure	set interfaces	N/A	N/A
Detection interfaces: display the MTU value	show advanced-configuration mtu	N/A	N/A
Detection interfaces: change the MTU value	set advanced-configuration mtu	N/A	N/A
Detection interfaces: display the replacement name of the interfaces	show advanced-configuration interface-names	N/A	N/A
Detection interfaces: detect/name the interfaces	set advanced-configuration rescan-interfaces	N/A	N/A
Aggregation of detection interfaces: display the information	show clusters	N/A	N/A
Aggregation of detection interfaces: configure	set clusters	N/A	N/A

4.6.5 Managing server

Table6: Managing server

Function per level	setup	gviewadm	gview
Display help on the commands	help	help	help
Launch the GCap configuration GUI	gui	gui	gui
Exit the current session or leave the SSH session	exit	system restart	system restart
System: restart the GCap	system restart	N/A	N/A
System: shut down the GCap	system shutdown	N/A	N/A
System: reloading network card drivers	system reload-drivers	N/A	N/A

4.6.6 Monitoring the GCAP

Table7: Monitoring the GCAP

Function per level	setup	gviewadm	gview
Monitoring: consult the alert logs	show alerts	show alerts	show alerts
Monitoring: CPU usage	show cpus	show cpus	show cpus
Monitoring: display the current status of the GCap	show status	show status	show status
Monitoring: display the statistics of the Sigflow detection engine	show eve-stats	show eve-stats	show eve-stats
Monitoring: display the different event logs	show logs	show logs	show logs
Monitoring: display statistics and health information	show health	show health	N/A
Monitoring: extract the information from the GCap as requested by technical support	show tech-support	N/A	N/A

Chapter 5

Use cases

5.1 Introduction

For the initial GCap configuration and to do advanced configurations or checks, it is necessary to use the CLI.
For most functions, the use of this interface is adequate.
The tables listed in the *Procedure list* section enable a general overview of the most common actions.

5.2 How to connect to Gcap?

Access to the GCap can be made:

- Either by a direct connection (connect directly to the server)
- Or by a HTTP remote connection (iDRAC function for a Dell server)
- Or by a remote connection to the CLI in SSH via the iDRAC interface in serial port redirection mode
- Or by a remote connection to the CLI in SSH via the network interfaces `gcp0` or `gcp1`

Access to the operating system and CLI for managing the GCap can be done remotely via an SSH or HTTP connection.

Note:

The list of physical connectors to use was described in the PRESENTATION - General section.

5.2.1 Direct connection and configuration

There is no specific configuration required other than knowing the iDRAC login name and password.
This access can be done to configure the network connection of the iDRAC among other things.
For implementation, refer to the *Procedure for direct connection to the GCap*.

Note:

The default login and password are provided in the server manufacturer's documentation.

5.2.2 Remote connection to the iDRAC in HTTP (DELL server)

The remote access can be made:

- Via the network connection to the iDRAC port of the GCap
- Using a WEB browser

This access requires:

- Knowledge of the iDRAC login name and password (iDRAC access)
- The network configuration is complete (IP address of the iDRAC is known)

This connection is not the normal way to access the GCap but enables access to the GCap in the event of problems. For its implementation, refer to the *Procedure for remote HTTP connection to iDRAC*.

5.2.3 Remote connection to the CLI using SSH via the iDRAC interface in serial port forwarding mode

The remote access can be made:

- Via the network connection to the iDRAC port of the GCap
- By using a connection tool via SSH

This access requires:

- Knowledge of the iDRAC login name and password (iDRAC access)
- The network configuration is complete (IP address of the iDRAC is known)

This connection is not the normal way to access the GCap but enables access to the GCap in the event of problems. For the implementation, refer to the *Procedure for remote connection to the CLI by SSH via the iDRAC interface in serial port forwarding mode*.

5.2.4 Remote connection to the CLI in SSH via the network interfaces `gcp0` or `gcp1`

Remote access to the GCap CLI is achieved via the network connection to the port:

- `gcp1` (dual-interface configuration) or
- `gcp0` (single interface configuration)

This connection is the nominal way to access the GCap.
For more information, see *Procedure for connecting to the GCap via SSH*.

5.3 Remote connection to the GCenter

Remote access to the GCenter is done either:

- By SSH to configure the GCenter.
  For more information, please refer to the GCenter documentation.
- Or via a web browser in order to pair the GCap.
  For more information, see *Procedure for connecting to the GCenter via a web browser*.

5.4 How to use the procedures

5.4.1 Accessing the GCap and GCenter

To perform the following task	#	Carry out the following procedures in succession
First connection to the GCap by a direct connection	1	*Direct connection to GCap with keyboard and monitor*
Remote connection to iDRAC via HTTP	1	*Remote connection to iDRAC via HTTP*
Remote SSH connection in serial port forwarding mode	1	*Remote connection to the CLI using SSH via the iDRAC interface in serial port forwarding mode*
Connection to the GCenter via a web browser	1	*Connection to the GCenter via a web browser*

5.4.2 Configuring the GCap

To perform the following task	Perform the following procedures in sequence	
The first installation to GCap	1	*Configuring the GCap on first login*
	2	*Putting a GCap into operation*
Keyboard configuration	1	Display: use the command *show keymap*
	2	Modify: use the command *set keymap*
Configuring the Gcap interface: (GUI or CLI)	1	Display: use the command *show setup-mode*
	2	Modify: use the command *set setup-mode*
Date and time	1	Display: use the command *show datetime*
	2	Modify: use the procedure *Change GCap date and time*
Colours in the display	1	Enable or disable: use the command *colour*
Compatibility mode with the GCenter	1	Show: use the command *show compatibility-mode*
	2	Modify: use the command *set compatibility-mode*
Services: start a service (to be defined)	1	View the status of services: use the command *services status + service to be defined*
	2	Starting a service: use the command *services start +service to be defined*
Services: stop a service (to be defined)	1	View the status of services: use the command *services status + service to be defined*
	2	Stopping a service: use the command *services stop +service to be defined*
Services: view status services	1	Viewing the status of services: use the command *services status +service to be defined*
Services: display the periods for file retention	1	Stop a service: use the command *services show retention-periods*
High availability	1	Show: use the command *show advanced-configuration high-availability*
	2	Management: use the procedure *Managing GCaps High Availability*
Pairing with GCenter	1	Use the procedure *Pairing between a GCap and a GCenter*

5.4.3 Managing accounts

To perform the following task	Perform the following procedures in sequence	
Authentication: the list of users	1	Display the list: use the command *show passwords*
	2	Change passwords: use the command *set passwords*
Authentication: modify the SSH keys	1	Use the command *set ssh-keys*
Authentication: display the password policy	1	Use the command *show password-policy*
Authentication: unlock blocked accounts	1	Use the command *system unlock*
Authentication: define a password policy	1	Use the command *set password-policy*
Authentication: display the protection policy against brute force attacks	1	Use the command *show bruteforce-protection*
Authentication: modify the protection policy against brute force attacks	1	Use the command *set bruteforce-protection*
Session: display the duration of inactivity before disconnection	1	Use the command *show session-timeout*
Session: modify the duration of inactivity before disconnection	1	Use the command *set session-timeout*

5.4.4 Managing networks

To perform the following task	Perform the following procedures in sequence	
Managing gcp0 and gcp1 interfaces	1	Use the procedure *Managing network settings for gcp0 and gcp1 interfaces*
IP address of the GCenter: display the GCenter IP address	1	Use the command *show gcenter-ip*
IP address of the GCenter: modify the GCenter IP address	1	Use the command *set gcenter-ip*
Manage the capture interfaces monx	1	Use the procedure *Manage monx capture interface settings*
Detection interfaces: display the replacement name of the monx capture interfaces	1	Use the command *show advanced-configuration interface-names*
Authentication: detect / name the capture interfaces monx	1	Use the command *set advanced-configuration rescan-interfaces*
Manage interface aggregation of capture	1	Use the procedure *Manage capture interface aggregation*
Switch to the configuration single-interface for connection SSH managed by the gcp0 interface	1	Use the procedure *Flip to single-interface configuration*
Switching to the configuration dual-interface for connection SSH managed by the gcp1 interface	1	Use the procedure *Flip to dual-interface configuration*

5.4.5 Managing the detection engine

Table1: Basic functions

To perform the following task	#	Carry out the following procedures in succession
Display the detection engine configuration as well as the rules	1	Use the command *show config-file*
Display advanced options	1	Use the command *show monitoring-engine*
Apply an advanced configuration	1	Use the command *set monitoring-engine*
Start the detection engine	1	Use the command *monitoring-engine start*
Stop the detection engine	1	Use the command *monitoring-engine stop*
Display the detection engine status	1	Use the command *monitoring-engine status*
Traffic generation: replaying a pcap file	1	Use the command *replay*

Table2: Advanced functions

To perform the following task	#	Carry out the following procedures in succession
Resource allocation: display the number of dedicated CPUs	1	Use the command *show advanced-configuration cpu-config*
Resource allocation: modify the number of dedicated CPUs	1	Use the command *set advanced-configuration cpu-config*
Load balancing monx capture interface - CPU: show the configuration	1	Use the command *show advanced-configuration load-balancing*
Load balancing monx capture interface - CPU: modify the configuration	1	Use the command *set advanced-configuration load-balancing*
Flow filtering: display static rules	1	Use the command *show advanced-configuration packet-filtering*
Flow filtering: specify the static rules	1	Use the command *set advanced-configuration packet-filtering*
Sigflow local rules: display	1	Use the command *show advanced-configuration local-rules*
Sigflow local rules: modify	1	Use the command *set advanced-configuration local-rules*
Optimise the performance of the GCap	1	Use the procedure *Optimize GCap performance*

5.4.6 Managing servers

To perform the following task	#	Carry out the following procedures in succession
Display help on the commands	1	Use the command *help*
Launch the GCap configuration GUI	1	Use the command *gui*
Exit the current session or leave the SSH session	1	Use the command *exit*
System: restart the GCap	1	Use the command *system restart*
System: shut down the GCap	1	Use the command *system shutdown*
System: reloading network card drivers	1	Use the command *system reload-drivers*

5.4.7 Monitoring the GCAP

To perform the following task	#	Carry out the following procedures in succession
Monitoring: consult the alert logs	1	Use the command *show alerts*
Monitoring: CPU usage	1	Use the command *show cpus*
Monitoring: display the current status of the GCap	1	Use the command *show status*
Monitoring: display the statistics of the Sigflow detection engine	1	Use the command *show eve-stats*
Monitoring: display the different event logs	1	Use the command *show logs*
Monitoring: display statistics and health information	1	Use the command *show health*
Monitoring: extract the information from the GCap as requested by technical support	1	Use the command *show tech-support*

5.5 List of procedures

5.5.1 Configuring the GCap for the first connection

5.5.1.1 Introduction

The procedure described here explains how to set up the GCap when it is first installed.

5.5.1.2 Prerequisites

- **User:** setup

5.5.1.3 Preliminary operations

- Check that the LUKS key is connected to the GCap.

Note:

If there is no LUKS key or if it is the wrong one, the operating system will not be able to access the contents on the hard drives.
In case of problems, check:

- Whether the correct key is used and not one from another GCap...
- The USB port is working properly: change the USB port

- Connect to the GCap.
- Depending on the situation:
  - Either connect directly to the GCap via keyboard and screen (see *Procedure for connecting directly to the GCap*)
  - Or connect to the GCap via the iDRAC (see *Procedure for connecting to the GCap via the iDRAC*)
- Connect as a **setup**.

Note:

The first time you log in to the GCap, a prompt to change your password will be displayed.
Make sure the keyboard configuration is correct (fr or en version).

5.5.1.4 Procedure

- Manage passwords (passwords, SSH keys, and the like): see the *Manage accounts* table.
- Manage network interfaces `gcp0` and `gcp1`: see the *Manage network* table.
  - Configure the IP addressing
  - Enter the GCap name and the domain name
  - Configure the MTU value if necessary

  To do this, see the *Procedure for managing the network parameters of the `gcp0` and `gcp1` interfaces*.
- Connect to the GCap via a remote connection through an SSH tunnel (see *Procedure for remote connection to GCap via an SSH tunnel*).
- Set the operating mode for the SSH link to single-interface or dual-interface

  To do this, see the *Procedure for switching to single-interface configuration* or the *Procedure for switching to dual-interface configuration*.
- Manage the GCap date and time, refer to the *Procedure for changing the GCap date and time*.
- Manage the capture interfaces: see the *Manage network* table.
  - Activate the desired interfaces
  - Configure the MTU value

  To do this, see the *Procedure for managing capture interface settings `monx`*.
- If needed, manage the aggregation of detection interfaces: see the *Procedure for managing the aggregation of capture interfaces*.
- If needed, manage the high availability of GCaps: see *Procedure for managing the high availability of GCaps*.
- Pairing the GCap with the GCenter: see the *Procedure for pairing a GCap with a GCenter*.
  - On the GCenter,
    - Connect via an SSH
    - Know the GCenter IP address
  - On the GCap, enter the IP address of the GCenter
  - On the GCenter, declare the GCap and generate the One Time Password (OTP)
  - On the GCap, pair the GCap and the GCenter
- Put the GCap into operation: see the *Procedure for putting a GCap into operation*.

5.5.2 Starting up a GCap

5.5.2.1 Introduction

After configuring the GCap, this procedure describes how to start operating the GCap.

5.5.2.2 Prerequisites

- **User:** setup

5.5.2.3 Preliminary operations

- Perform the *Procedure for connecting to the GCap for the first time*.
- Activate the required `monx` capture interfaces: see *Procedure for managing `monx` capture interface settings*.

5.5.2.4 Procedure to be followed on the GCap

- Starting the detection engine: see the *Managing the detection engine* table.
  The system displays the following command prompt:

  ```
  Monitoring DOWN gcap-name (gcap-cli)
  ```

  The command prompt indicates the status of the detection engine: here it is stopped.
- Enter the following command.

  ```
  (gcap-cli) monitoring-engine start
  ```

- Validate.
- Wait for the engine to be up and running.
- Check the status of the detection engine.
  The system displays the following command prompt:

  ```
  [Monitoring UP] gcap-name (gcap-cli)
  ```

  The command prompt indicates the status of the detection engine: here it is running.

5.5.2.5 Procedure to be carried out on the GCenter

- Apply a ruleset to the GCap.
- Enable or disable the shellcode detection.
- Enable or disable the powershell detection.
- Enable or disable powershell detection.
- Configure the Sigflow specific parameters, namely Base variables, Net variables and File rules management.

5.5.3 Direct connection to the GCap with keyboard and monitor

5.5.3.1 Introduction

The first connection to the GCap can be done by a direct connection with a keyboard and monitor.
This is necessary if the network configuration is not yet completed on the GCap or if the network address is not known.

5.5.3.2 Preliminary operations

- Connect the GCap power cables.
- Connect the network cables of the GCap (*see section Description / The GCap*).

5.5.3.3 Procedure for connecting the screen and keyboard

- Connect the screen to the VGA connector of the GCap.
- Connect the keyboard to the USB connector of the GCap.
- Switch on the server.

5.5.3.4 Procedure for obtaining the network settings via the BIOS

- Press **F2** during the boot up self-test (POST).
- On the `System Setup Main Menu` page (System Setup main menu), click on `iDRAC Settings` (iDRAC Settings).
  The `Paramètres iDRAC` page appears.
- Click on `Réseau`.
  The Network page appears.
- Note the network settings in the `Network Settings` settings.
- After noting down the network settings, exit the BIOS.
- Click successively on `Retour`, `Terminer` and `Non`.

5.5.3.5 Procedure for accessing the CLI

The command prompt is displayed:

```
gcap-protor login:
```

- Enter the login and the corresponding password.
  The following command prompt is displayed:

  ```
  gcap-protor (gcap-cli)
  ```

Note:

The first time you log in to the GCap, a prompt to change your password will be displayed.

Note:

Press **Tab** to display all available commands.
Press **Enter** to display all available commands along with a short explanation.

Astuce:

If a password error occurs, the protection system will be activated.
To view the policy setting on the Gcap, use the `show bruteforce-protection` command.
After a certain number of failures, the account will be locked.
To unlock it: either wait, or use the `system unlock` command with a higher privilege level account.

5.5.4 Remote connection to the iDRAC in HTTP (DELL server)

5.5.4.1 Introduction

This procedure describes the remote connection from a distant PC using:

- The network connection to the iDRAC port of the GCap
- A WEB browser

This connection is not the normal way to access the GCap but enables access to the GCap in the event of problems.

To carry out this procedure, it is necessary:

- That the iDRAC has an accessible IP in order to be able to connect to it
- To know the login name and password of iDRAC

From the iDRAC web page, it is possible to:

- View the material resources, their status and the BIOS configurations
- Interact with the server to turn it on, off and restart it
- Connect to the GCap via the CLI console

> **Astuce:**
>
> If a password error occurs, the protection system will be activated.
> To view the policy setting on the Gcap, use the `show bruteforce-protection` command.
> After a certain number of failures, the account will be locked.
> To unlock it: either wait, or use the `system unlock` command with a higher privilege level account.

5.5.4.2 Preliminary operations

- Perform the network configuration (IP address of the iDRAC): otherwise, use the *Procedure for direct connection to the GCap* to connect to the GCap.

5.5.4.3 Procedure

- On the remote PC, open a web browser.
- Enter the IP address of the GCap iDRAC interface and confirm.

  The `Login` window is displayed.

- Enter the requested parameters:
  - `Username`: login name
  - `Password`: password of the entered login
  - `Domain`: select `This IDRA`
- Click on the `Submit` button.
- Launch the virtual console (`Virtual console Preview` zone, `Lanch` button).
  Following this action, a new page will open. It will be possible to interact with the GCap.
- Connect to the CLI (`gcap-cli` command).
  After connection, the following message is displayed:

```
(gcap-cli)
```

Note:

Press **Tab** to display all available commands.
Press **Enter** to display all available commands along with a short explanation.

5.5.4.4 Special cases

It is possible to open an SSH connection, run a CLI command line and then close the connection.
To do this:

- Enter the command

```
~$ ssh -t setup@x.x.xx.x show status
```

- Validate
  The system:
  - Opens the SSH connection
  - Executes the command (here `show status`)
  - Closes the SSH connection

```
GCAP Name          :
Version            : z.z.z
Paired on GCenter  : Not paired
Tunnel status      : Down
Detection Engine   : Up and running
© Copyright GATEWATCHER 202
Connection to x.x.x.x closed
```

5.5.5 Remote connection to the CLI using SSH via the iDRAC interface in serial port forwarding mode

5.5.5.1 Introduction

This procedure describes the remote connection from a distant PC using:

- The network connection to the iDRAC port of the GCap
- A connection tool via SSH

This connection is not the normal way to access the GCap but enables access to the GCap in the event of problems.
To carry out this procedure, it is necessary:

- That the iDRAC has an accessible IP in order to be able to connect to it
- To know the login name and password of iDRAC

From the interface, it is possible to:

- View the operating system messages
- Connect to the GCap via the CLI console

Astuce:

If a password error occurs, the protection system will be activated.
To view the policy setting on the Gcap, use the `show bruteforce-protection` command.
After a certain number of failures, the account will be locked.
To unlock it: either wait, or use the `system unlock` command with a higher privilege level account.

5.5.5.2 Preliminary operations

- Perform the network configuration (IP address of the iDRAC). Otherwise, use the *Procedure for direct connection to the GCap* to connect to the GCap.

5.5.5.3 Procedure

- On the remote PC running Linux:
  - Open a command prompt
  - Enter the `ssh identifiant@adresse_ip` command
    For example, `ssh setup@x.x.x.x` where - `setup` is the ID and - x.x.x.x is the IP address of the GCap's iDRAC port.
  - Validate the command
  - Enter password of the entered login
  - Press **Enter** to display all available commands and a short explanation
- On a Windows PC:
  - Open an SSH client software, such as Putty
  - Enter the IP address of the iDRAC interface of the GCap then validate
- Enter the following command `racadm>>console com2`
- Validate
  The system now displays the graphical interface of the device.
  Following this action, a new page will open. It will be possible to interact with the GCap.
- Connect to the CLI (gcap-cli command)

After connection, the following message is displayed:

```
(gcap-cli)
```

Note:

Press **Tab** to display all available commands.

Press **Enter** to display all available commands along with a short explanation.

5.5.5.4 Special cases

It is possible to open an SSH connection, run a CLI command line and then close the connection.
To do this:

- Enter the command

```
~$ ssh -t setup@x.x.xx.x show status
```

- Validate
  The system:
  - Opens the SSH connection
  - Executes the command (here `show status`)
  - Closes the SSH connection

```
GCAP Name          :
Version            : z.z.z
Paired on GCenter : Not paired
Tunnel status      : Down
Detection Engine  : Up and running
© Copyright GATEWATCHER 202
Connection to x.x.x.x closed.
```

5.5.6 Remote connection to GCap via an SSH tunnel

5.5.6.1 Introduction

This procedure describes how to connect from a remote PC securely using an SSH tunnel.

Astuce:

If a password error occurs, the protection system will be activated.
To view the policy setting on the Gcap, use the `show bruteforce-protection` command.
After a certain number of failures, the account will be locked.
To unlock it: either wait, or use the `system unlock` command with a higher privilege level account.

5.5.6.2 Preliminary operations

- Make an initial connection to the GCap (see *Procedure for direct connection to GCap*).
- Learn the name of the GCap or its IP address (see *Procedure for viewing network interface settings gcp0and gcp1*).

5.5.6.3 Procedure

- On the remote PC running Linux:
  - Open a command prompt
  - Enter the `ssh identifiant@adresse_ip_GCap` or `ssh identifiant@FQDN_GCap` command
    For example, `ssh setup@gcap` where:
  - The identifier is `setup` and
  - The FQDN is `gcap`.
  - Validate the command
  - Enter password of the entered login
- On a Windows PC:
  - Open an SSH client software, such as Putty
  - Enter the IP address of the interface GCap then validate

The command prompt is displayed.

```
[Monitoring DOWN] GCap name (gcap-cli)
```

Note:

Press **Tab** to display all available commands.
Press **Enter** to display all available commands along with a short explanation.

5.5.7 Connection to the GCenter via a web browser

5.5.7.1 Introduction

This procedure describes how to connect from a remote PC to the GCenter via a web browser.

5.5.7.2 Preliminary operations

- Know the name of the GCenter or its IP address.
- Connect to a PC linked to the GCap and GCenter network.

5.5.7.3 Procedure

On the remote PC:

- Open a web browser.
- Enter the following URL:
  - `ssh identifiant@adresse_ip`
  - or `ssh identifiant@FQDN`

  For example: *`ssh setup@gcenter.domain.com`* *with:*
  - *the identifier is* *`setup`*
  - *the FQDN is* *`gcenter.domain.com`*
- Validate.

  The GCenter login window is displayed.
  - Enter the login name.
  - Enter the password.
  - Validate.

The GCenter graphical interface is displayed.

Note:

Refer to the GCenter documentation for its use.

5.5.8 Changing the GCap date and time

5.5.8.1 Introduction

Before pairing the GCap and GCenter, it is important to ensure that both systems are in sync in terms of time. Once the pairing is complete, the GCenter acts as an NTP server for the GCap to ensure that the equipment clocks are synchronised.
When connecting for the first time, these items must be set via the *datetime* command in the CLI.
The adjustment is necessary for establishing the IPsec tunnel.
The datetime of the GCap and the GCenter must be the same to within 1 minute.

Important:

If there is a discrepancy, it is the time of the GCap that must be changed.

5.5.8.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show datetime*
  - *set datetime*

5.5.8.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Connect as a **setup**.

5.5.8.4 Procedure for viewing the date and time on the GCap and GCenter

- Enter the `show datetime` command then validate.
  The `datetime` command of the `show` subgroup enables displaying the date and time of the GCap in the format YYYY-MM-DD HH:MM:SS.

```
(gcap-cli) show datetime
Current datetime is 2022-01-26 16:10:44
```

- Log in to the GCenter.
- Display the GCenter date and time and note them down.
  If there is a discrepancy between the GCap and the GCenter, the GCap time is the one to be changed.
- To correct this, perform the following procedure.

5.5.8.5 Procedure for changing the date and time of the GCap

- Enter the command *`set datetime`* followed by the parameters in the following order {YYYY-MM-DDThh:mm:ssZ}.
  Example: set datetime 2022-01-26T16:00:00Z
  - YYYY indicates a four-digit year from 0000 to 9999.
  - MM indicates a two-digit year from 01 to 12.
  - DD indicates a two-digit year from 01 to 31.
  - T indicates the beginning of the field defining the time format
  - hh indicates the two-digit hour from 00 to 23.
  - mm indicates the two-digit minutes from 00 to 59.
  - ss indicates the two-digit seconds from 00 to 59.
  - Z indicates CUT (Coordinated Universal Time)

```
(gcap-cli) set datetime 2022-01-26T16:00:00Z
```

- Validate.
  A confirmation window is displayed.

```
Date successfully changed to Wed Jan 26 2022 16:00:00
```

5.5.9 Managing the network parameters of the gcp0 and gcp1 interfaces

5.5.9.1 Introduction

This procedure describes:

- Viewing the network settings
- Modifying these parameters.

To...	Use the command	described in the procedure
obtain an overview of the information on all network interfaces	show network-config configuration	Procedure A
display for each interface: MAC address, carrier presence, speed, and type of connection	show network-config status	Procedure B
display or change the domain name	show network-config domain set network-config domain	Procedure C
display or change the system name	show network-config hostname set network-config hostname	Procedure D
display or modify the interface used in SSH for administering the GCap and the GCap GCenter link	show network-config ssh and set network-config ssh	Procedure E
display or modify the MTU value of the interfaces	show advanced-configuration mtu set advanced-configuration mtu	Procedure F
display or modify the TCP/IP settings of the GCPx interfaces	show network-config gcpx	Procedure G

5.5.9.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show network-config configuration*
  - *show network-config status*
  - *show network-config domain*
  - *set network-config domain*
  - *show network-config hostname*
  - *set network-config hostname*
  - *show network-config ssh*
  - *set network-config ssh*
  - *show advanced-configuration mtu*
  - *set advanced-configuration mtu*
  - *show network-config gcp0*
  - *set network-config gcp0*

5.5.9.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Stop the detection engine (see *monitoring-engine*).

5.5.9.4 Procedure A: Display the network configuration

- Enter the `show network-config configuration` command then validate.
  The system displays the information of all network interfaces.
  In this procedure, only the information on the gcpx network interfaces is detailed.
  For information on the `monx` capture interfaces, refer to the *Procedure for managing monx capture interface settings*.
  The system displays the following information:
  - System name (**hostname**)
  - Domain name (**domain_name**)
  - Details of the TCP/IP settings for each network interface (`gcp0` and `gcp1`)
  - Whether or not the interface is enabled

```
(gcap-cli) show  network-config configuration

{
    "hostname": "GCap",
    "domain_name": "domain.local",
    "gcp0": {
        "description": "VPN / SSH",
        "ip_address": "192.168.1.1",
        "mask": "255.255.255.0",
        "default_gateway": "192.168.1.254",
        "enabled": true,
        "mtu": 1500
    },
    "gcp1": {
        "description": "SSH",
        "ip_address": "None",
        "mask": "255.255.255.0",
        "default_gateway": "",
        "enabled": false,
        "mtu": 1500
    },
```

Note:

The configuration in the above example is single interface, i.e. `gcp0` used and `gcp1` not used.

For gcp0:

- The **description** field indicates that VPN and SSH connections are on this interface
- The TCP/IP parameters are listed
- Since the interface is enabled, the **enabled** parameter is **true**

For gcp1:

- The **description** field indicates that the SSH connection is on this interface
- The TCP/IP parameters are set to **None**
- Since the interface is disabled, the **enabled** parameter is **false**

5.5.9.5 Procedure B: display the status of the GCap network interfaces gcp0 and gcp1

- Enter the following command.

```
(gcap-cli) show network-config status
```

- Validate.
  The system displays the status of the GCap network interfaces.

```
   Name         Address        Carrier       Speed           Type
   gcp0     xx:xx:xx:xx:xx:xx        UP    1000Mb/s           RJ45
   gcp1     xx:xx:xx:xx:xx:xx        UP    1000Mb/s           RJ45
```

  For each interface, the following information is displayed:
  - Address: the MAC address of the interface
  - Carrier: status of the current transmission:
    * value UP: physical interface is connected
    * value DOWN: physical interface is not connected
  - Speed: the interface speed in Mb/s
  - Type: the type of cable/sfp connected to the physical port

5.5.9.6 Procedure C : display/change the GCap domain name

- To display the current name:
  - Enter the following command.

```
(gcap-cli) show network-config domain
```

  - Validate.
    The system displays the domain name.

```
Current domain name: gatewatcher.com
```

- To change the current name:
  - Enter the following command.

```
(gcap-cli) set network-config domain-name gatewatcher.com
```

  - Validate.

```
 Setting hostname/domain name to:
    - Hostname: gcap-int-129-dag
    - Domain name: gatewatcher.com
 Do you want to apply this new configuration? (y/N)
```

  - Press **y** and then confirm.

```
Applying configuration...

00% Generating interfaces configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting gcenter-xfer-daemon service  [OK]
81% Restarting netdata service              [OK]
```

(suite sur la page suivante)

(suite de la page précédente)

```
90% Restarting rsyslog service              [OK]
Procedure completed with success
```

- To check the value modification:
  - Enter the following command.

```
(gcap-cli) show network-config domain
```

  - Validate.
    The system displays the domain name.

```
Current domain name: gatewatcher.com
```

5.5.9.7 Procedure D: display or change the GCap name

- To display the current name:
  - Enter the following command.

```
(gcap-cli) show network-config hostname
```

  - Validate.
    The system displays the interface the host name of the GCap.

```
Current hostname: GCap-name
```

- To change the current name:
  - Enter the following command.

```
(gcap-cli) set network-config hostname gcap-name
```

  - Validate.

```
Setting hostname/domain name to:
 - Hostname: gcap-name
 - Domain name: gatewatcher.com
Do you want to apply this new configuration? (y/N)
```

  - Press **y** and then confirm

```
Applying configuration...

00% Generating interfaces     configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting gcenter-xfer-daemon service  [OK]
81% Restarting netdata service              [OK]
90% Restarting rsyslog service              [OK]
Procedure completed with success
```

- To check the value modification:
  - Enter the following command.

```
(gcap-cli) show network-config hostname
```

  - Validate.
    The system displays the host name of the GCap.

```
Current hostname: GCap-name
```

5.5.9.8 Procedure E: display or modify the interface used to manage the GCap in SSH

- To display the current configuration:
  - Enter the following command.

```
(gcap-cli) show network-config ssh
```

  - Validate.
    The system displays the SSH interface used to manage the GCap.
    * In the case of the single-interface configuration, the system displays:

```
SSH is using interface gcp0
```

    * In the case of dual-interface configuration, the system displays:

```
SSH is using interface gcp1
```

- To configure the gcp1 interface in SSH:
  - Enter the following command.

```
(gcap-cli) set network-config ssh gcp1
```

  - Validate.
  - Enter the following command.

```
(gcap-cli) set network-config gcp0 ip-address X.X.X.X gateway X.X.X.X mask X.X.X.X
```

  - Validate.
  - Enter the following command.

```
(gcap-cli) set network-config gcp1 ip-address Y.Y.Y.Y gateway Y.Y.Y.Y mask Y.Y.Y.Y␣
↪confirm
```

  - Validate.
- To configure the gcp0 interface in SSH:

Note:

The gcp1 interface is not used.

  - Enter the following command.

```
(gcap-cli) set network-config ssh gcp0
```

  - Validate.
  - Enter the following command.

```
(gcap-cli) set network-config gcp0 ip-address X.X.X.X gateway X.X.X.X mask X.X.X.X␣
↪confirm
```

  - Validate.

5.5.9.9 Procedure F: display or change the MTU value

- To display the current configuration of enabled interfaces:
  - Enter the following command.

```
(gcap-cli) show advanced-configuration mtu
```

  - Validate.
    The system displays the result.

```
 Current Network MTU configuration:
   - mon1: 1500
   - mon2: 1500
   - mon3: 1500
   - cluster0: 1500
   - gcp0: 1500
```

    The values are displayed for all enabled network interfaces.

- To change the current configuration of enabled interfaces: e.g. to change the MTU value of the gcp0 interface:
  - Enter the following command.

```
(gcap-cli) set advanced-configuration mtu gcp0 2000
```

  - Validate.
    The system displays the result.

```
Updating Network MTU configuration to:
   - gcp0: 2000
```

5.5.9.10 Procedure G: display or modify the TCP/IP settings of a gcpx interface

- To display the gcp0 interface configuration:
  - Enter the following command.

```
(gcap-cli) show network-config gcp0
```

  - Validate.
    The system displaying the gcp0 interface configuration.
    Depending on the single or dual interface configuration, the information is different.
    The two cases are listed below.
  - **Single-interface configuration**
    SSH and VPN connections are handled by the gcp0 interface.
    In this case, the system displays:

```
Interface gcp0 configuration (VPN / SSH):
  - IP Address: X.X.X.X
  - Mask: 255.255.255.0
  - Gateway: X.X.X.X
```

  - **Dual-interface configuration**
    The VPN communication is controlled by the gcp0 interface.
    The SSH connection for GCap management is handled by the gcp1 interface.
    In this case, the system displays:

```
Interface gcp0 configuration (VPN):
  - IP Address: X.X.X.X
  - Mask: 255.255.255.0
  - Gateway: X.X.X.X
```

- To change the configuration of the gcp0 interface address:

- Enter the following command.

```
(gcap-cli) set network-config gcp0 ip-address x.x.x.x gateway y.y.y.y mask z.z.z.z
```

- Validate.
  The system displaying the gcp0 interface configuration.

```
Setting interface gcp0 (VPN / SSH) to configuration :
   - IP Address: 10.2.19.129
   - Mask: 255.255.255.0
   - Gateway: 10.2.19.254
Do you want to apply this new configuration? (y/N)
```

- Press **y** and then confirm.

```
Applying configuration...
00% Generating interfaces configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting gcenter-xfer-daemon service  [OK]
81% Restarting netdata service              [OK]
90% Restarting rsyslog service              [OK]
Procedure completed with success
```

5.5.10 Managing capture interface settings monx

5.5.10.1 Introduction

This procedure describes:

- Viewing the network settings
- Modifying these parameters

To...	Use the command	described in the procedure
obtain an overview of the information on all network interfaces	show network-config configuration	Procedure A
display or modify the MTU value of the interfaces	show advanced-configuration mtu set advanced-configuration mtu	Procedure B
modify the MTU value of the interfaces	set advanced-configuration mtu	Procedure B
display or administer the available detection interfaces	show interfaces set interfaces	Procedure C
administer the available detection interfaces	set interfaces	Procedure C

5.5.10.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show network-config configuration*
  - *show advanced-configuration mtu*
  - *set advanced-configuration mtu*
  - *show interfaces*
  - *set interfaces*

5.5.10.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Stop the detection engine (see *monitoring-engine*).

5.5.10.4 Procedure A: Display the network configuration

In this procedure, only the information on the capture interfaces is detailed.
For information on GPCx network interfaces, refer to the *Procedure for managing the network settings of `gcp0` and `gcp1` interfaces*.

- Enter the `show network-config configuration` command then validate.
  The system displays the information of all network interfaces.

```
(gcap-cli) show network-config configuration
{
    ...
    },
    "mon0": {
        "description": "default",
        "enabled": true,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon1": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon2": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon3": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    }
}
```

Note:

The mon0 interface is enabled (**enabled** field: **true**).
The mon1 to mon3 interfaces are disabled (**enabled** field: **false**).

5.5.10.5 Procedure B: display or change the MTU value

- To display the current configuration of enabled interfaces:
  - Enter the following command.

```
(gcap-cli) show advanced-configuration mtu
```

  - Validate.
    The system displays the result.

```
Current Network MTU configuration:
     - mon1: 1500
     - mon2: 1500
     - mon3: 1500
     - cluster0: 1500
     - gcp0: 1500
```

The values are displayed for all enabled network interfaces.

- To change the current configuration of enabled interfaces: e.g. to change the MTU value of the mon1 interface
  - Enter the following command.

```
(gcap-cli) set advanced-configuration mtu mon1 2000
```

  - Validate.
    The system displays the result.

```
Updating Network MTU configuration to:
        - mon1: 2000
```

5.5.10.6 Procedure C: display or change the available detection interfaces

- To display the information on the detection interfaces:
  - Enter the following command.

```
(gcap-cli) show interfaces
```

  - Validate.
    The system displays the available detection interfaces.

```
Waiting 10s for interfaces to be up

Name     State     Physical Address   Status   Speed    Type
mon0     Enabled   xx:xx:xx:xx:xx:xx    UP     10Gb     10G Base-SR
mon1     Disabled  xx:xx:xx:xx:xx:xx    UP     1Gb      1G Base-SR
mon2     Disabled  xx:xx:xx:xx:xx:xx    UP     1Gb      1G Base-SR
mon3     Disabled  xx:xx:xx:xx:xx:xx    UP     1Gb      1G Base-SR
monvirt  Enabled   N/A                  UP     N/A      Virtual
```

The information displayed is:

* **Status:** the configured status of the interface among {Enabled|Disabled}

    - **Physical Address:** the MAC address of the interface
    - **Speed:** the speed of the interface
    - **Type:**
      - If it concerns a virtual interface: Virtual
      - If it is a physical interface: the type of cable/sfp connected to the physical port

- To activate an interface (here mon0 for example):
  - Enter the following command.

    ```
    (gcap-cli) set interfaces enable mon0
    ```

  - Validate.
- To deactivate an interface (here mon0 for example):
  - Enter the following command.

    ```
    (gcap-cli) set interfaces disable mon1
    ```

  - Validate.
- To change the interface start-up delay by five seconds for example
  - Enter the following command.

    ```
    (gcap-cli) set interfaces delay 5
    ```

  - Validate.

5.5.11 Switching to a single-interface configuration

5.5.11.1 Introduction

In single-interface configuration, the SSH connection for managing the GCap and the VPN communication are handled by the `gcp0` interface.
In dual-interface configuration:

- The VPN communication is controlled by the `gcp0` interface
- The SSH connection for GCap management is handled by the `gcp1`

This procedure outlines the switchover from a dual-interface configuration to a single-interface configuration.

Important:

The user will lose the session if the connection between the GCap and the user's PC is made remotely via SSH.
This is because in dual-interface, the link via SSH is on the \``gcp1`\` interface.
However, after running this command, this link will be disabled and the interface to be used will be \``gcp0`\`.
In order to avoid this disconnection, connect to the GCap:

- Either by a direct connection (connect directly to the server)
- Or by a HTTP remote connection (iDRAC function for a Dell server)
- Or by a remote connection to the CLI in SSH via the iDRAC interface in serial port redirection mode

5.5.11.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show network-config*
  - *set network-config ssh*

5.5.11.3 Preliminary operations

- As appropriate, refer to:
  - The *Procedure for direct connection to the GCap*
  - The *Procedure for remote HTTP connection to iDRAC*
  - The *Procedure for remote connection to the CLI using SSH via the iDRAC interface in serial port forwarding mode*
- Stop the detection engine (see *monitoring-engine*)

5.5.11.4 Procedure for displaying the current configuration

- To display the gcp0 interface configuration:
  - Enter the following command.

    ```
    (gcap-cli) show network-config gcp0
    ```

  - Validate.
    The system displaying the gcp0 interface configuration.
    Depending on the single or dual interface configuration, the information is different.
    The two cases are listed below.
    - **Single-interface configuration**
      SSH and VPN connections are handled by the gcp0 interface.
      In this case, the system displays:

      ```
      Interface gcp0 configuration (VPN / SSH):
         - IP Address: X.X.X.X
         - Mask: 255.255.255.0
         - Gateway: X.X.X.X
      ```

      The VPN/ SSH field indicates that the current configuration is single-interface.
      In this case, there is nothing to do.
    - **Dual-interface configuration**
      The VPN communication is controlled by the gcp0 interface.
      The SSH connection for GCap management is handled by the gcp1 interface.
      In this case, the system displays:

      ```
      Interface gcp0 configuration (VPN):
         - IP Address: X.X.X.X
         - Mask: 255.255.255.0
         - Gateway: X.X.X.X
      ```

      The VPN field indicates that the current configuration is dual-interface.
      The absence of the SSH parameter on the gcp0 interface indicates that the gcp1 interface is managing the SSH.
      conclusion: the current configuration is dual-interface.
      In this case, continue with this procedure.

5.5.11.5 Procedure for switching from dual to single interface configuration

- Enter the *set network-config ssh gcp0* command followed by the network parameters of the `gcp0` interface.
  Example:
  - `set network-config gcp0 ip-address 192.168.1.1 gateway 192.168.1.254 mask 255.255.255.0`

```
(gcap-cli) set network-config gcp0 ip-address 192.168.1.1 gateway 192.168.1.254 mask 255.
↪255.255.0
```

- Validate.

```
Setting interface gcp0 (VPN / SSH) to configuration:
   - IP Address: 192.168.1.1
   - Mask: 255.255.255.0
   - Gateway: 192.168.1.254
Do you want to apply this new configuration? (y/N)
```

- Press **y** and then confirm.

```
Applying configuration...
00% Generating interfaces configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting GCenter-xfer-daemon service  [OK]
81% Restarting heartbeat service            [OK]
90% Restarting netdata service              [OK]
Procedure completed with success
```

  The system shows the progress and displays the message - **Procedure completed with success**, to indicate that the switch to single interface was successful.
- Rewire the GCap network cables if necessary.

> **Note:**
>
> It is necessary to add the command attribute 'confirm' at the end of the command if the pairing with the GCenter is active.

5.5.12 Switching to a dual-interface configuration

5.5.12.1 Introduction

In single-interface configuration, the SSH connection for managing the GCap and the VPN communication are handled by the `gcp0` interface.
In dual-interface configuration:

- The VPN communication is controlled by the `gcp0` interface
- The SSH connection for GCap management is handled by the `gcp1`

This procedure outlines the switch-over from a single-interface configuration to a dual-interface configuration.

Important:

The user will lose the session if the connection between the GCap and the user's PC is made remotely via SSH.
This is because in dual-interface, the link via SSH is on the `` `gcp1` `` interface.
However, after running this command, this link will be disabled and the interface to be used will be `` `gcp0` ``.
In order to avoid this disconnection, connect to the GCap:

- Either by a direct connection (connect directly to the server)
- Or by a HTTP remote connection (iDRAC function for a Dell server)
- Or by a remote connection to the CLI in SSH via the iDRAC interface in serial port redirection mode

5.5.12.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show network-config*
  - *set network-config ssh*

5.5.12.3 Preliminary operations

- As appropriate, refer to:
  - The *Procedure for direct connection to the GCap*
  - The *Procedure for remote HTTP connection to iDRAC*
  - The *Procedure for remote connection to the CLI using SSH via the iDRAC interface in serial port forwarding mode*
- Stop the detection engine (see *monitoring-engine*).

5.5.12.4 Procedure for displaying the gcp0 interface configuration

- Enter the following command.

```
(gcap-cli) show network-config gcp0
```

- Validate.
  The system displaying the gcp0 interface configuration.
  Depending on the single or dual interface configuration, the information is different.
  The two cases are listed below.
  - **Dual-interface configuration**
    The VPN communication is controlled by the gcp0 interface.
    The SSH connection for GCap management is handled by the gcp1 interface.
    In this case, the system displays:

```
Interface gcp0 configuration (VPN):
  - IP Address: X.X.X.X
  - Mask: 255.255.255.0
  - Gateway: X.X.X.X
```

The VPN field indicates that the current configuration is dual-interface.
The absence of the SSH parameter on the `gcp0` **interface indicates that the** `gcp1` **interface is managing the SSH.**
conclusion: the current configuration is dual-interface.
In this case, there is nothing to do.

- **Single-interface configuration**
  SSH and VPN connections are handled by the `gcp0` interface.
  In this case, the system displays:

  ```
  Interface gcp0 configuration (VPN / SSH):
    - IP Address: X.X.X.X
    - Mask: 255.255.255.0
    - Gateway: X.X.X.X
  ```

  The VPN/ SSH field indicates that the current configuration is single-interface.
  In this case, continue with the following procedure.

5.5.12.5 Procedure for switching from single to dual interface configuration

- Enter the *set network-config ssh gcp1* command followed by the network parameters of the `gcp1` interface.

  Note:

  Example
  `set network-config gcp1 ip-address 192.168.1.2 gateway 192.168.1.254 mask 255.255.255.0`

- Validate.

```
(gcap-cli) set network-config ssh gcp1
SSH has been set to interface gcp1
Do you want to apply this new configuration? (y/N)
```

- Press **y** and then confirm.

```
Applying configuration...
00% Generating interfaces configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting GCenter-xfer-daemon service  [OK]
81% Restarting heartbeat service            [OK]
90% Restarting netdata service              [OK]
Procedure completed with success
```

  The system shows the progress and displays the message - **Procedure completed with success**, to indicate that the switch to dual interface was successful.

- Rewire the GCap network cables if necessary.

5.5.13 Managing capture interface aggregation

5.5.13.1 Introduction

This procedure describes the aggregation of `monx` capture interfaces.
For more information on aggregation, see the paragraph *Capture and monitoring interfaces monx between TAP and GCap: aggregation possibility*.
The aggregation functionality of the capture interfaces on the GCap leads to impacting some related functions:

- Maximum Transmission Unit (MTU): the maximum size of a packet that can be transmitted at one time without fragmentation.
  MTU: uses the largest value of the interfaces making up the aggregation.
- Static rules for filtering flows captured by capture interface: XDP (eXpress Data Path) filter function.
  XDP filter. XDP filtering is not applied by default to the aggregation created but rather to the interfaces that comprise it. It must therefore be applied individually to each aggregated interface.
- File rebuilding rules.
  Rebuild rule: When enabling interface aggregation and multi-tenant detection, file rebuild rules are not generated.

To create an aggregation of `mon0` and `mon1` interfaces, use the *set clusters add interfaces mon0 mon1* command.

5.5.13.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show clusters*
  - *set clusters*

5.5.13.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Stop the detection engine (see *monitoring-engine*).

5.5.13.4 Procedure for displaying the aggregation of capture interfaces

- Enter the following command.

```
(gcap-cli) show clusters
```

- Validate.
  The system displays the aggregation if it exists.
  If none exists, then the following message is displayed:

```
No network cluster defined.
```

5.5.13.5 Procedure for displaying the available capture interfaces and activating the 2 interfaces to be aggregated

- Use Procedure C of the *Procedure for managing capture interface settings monx*.
- Note the interfaces to be used (e.g. mon0 and mon1).

5.5.13.6 Procedure to create an interface aggregation

- Enter the following command.

  > **Note:**
  >
  > The description of an aggregation of interfaces is optional (part *description test*).

```
(gcap-cli) set clusters add interfaces mon0 mon1 description `test`
```

- Validate.
  The system displays the result.

```
Creating cluster test with interfaces mon0, mon1
Successfully created cluster `test`
```

5.5.13.7 Procedure for displaying the status of the created aggregation

- Enter the following command.

```
(gcap-cli) show clusters
```

- Validate.
  The system displays the created aggregation.

```
Name       State      Description    Interfaces
cluster0   Disabled   test           mon0, mon1
```

The aggregation, once created with the Name **cluster0**, must be activated.

5.5.13.8 Procedure for activating the created aggregation

- Enter the following command.

```
(gcap-cli) set clusters enable cluster0
```

- Validate.
  The system displays the following message.

```
Enabling cluster cluster0
```

5.5.14 Pairing between a GCap and a GCenter

5.5.14.1 Introduction

This procedure describes the pairing between a GCap and a GCenter.
The following operations must be performed:

- On the GCenter, get the IP address of the GCenter
- On the GCap, enter the IP address of the GCenter
- On the GCenter, declare the GCap and generate the One Time Password (OTP)
- On the GCap, pair the GCap and the GCenter

5.5.14.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show compatibility-mode*
  - *set compatibility-mode*
  - *show gcenter-ip*
  - *set gcenter-ip*
  - *show status*
  - *pairing otp*

5.5.14.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Know the FQDN of the GCap and its IP address.
- Know the FQDN of the GCenter and its IP address.
- Check that the date and time of the GCenter and the GCap match: refer to the *Procedure for modifying the GCap date and time*.

5.5.14.4 Procedure for displaying the IP address of the GCenter

- Connect to the GCenter and display the GCenter network settings.
  For more information, please refer to the GCenter documentation.

5.5.14.5 Procedure for setting the compatibility mode on the GCap

- To view the software version of the GCenter:
  - Log into the GCenter and view the GCenter version number.
    The information is located at the bottom left of the GCenter page (GCenter v2.5.3.101-7173-HF3).
- To display the current compatibility mode between the GCap and the GCenter:
  - Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
  - Enter the following command.

```
(gcap-cli) show compatibility-mode
```

  - Validate. The system displays the current compatibility mode.

```
Current compatibility mode: 2.5.3.101
```

  - Compare the version between the one displayed on the GCap and the one on the GCenter.
    In this example:
    * On the GCenter, the version is: v2.5.3.101
    * On the GCap, the mode is: 2.5.3.101

    Thus, the GCap is well configured.
    In this example, it is not necessary to modify the compatibility mode.
    However, if it is necessary to change the mode, use the following procedure.
- To change the GCap compatibility mode:
  - Enter the following command (for example for version 2.5.3.102).

```
(gcap-cli) set compatibility-mode 2.5.3.102
```

  - Validate.

5.5.14.6 Procedure for setting the GCenter IP on the GCap

- To display the current version of the GCenter IP:
  - Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
  - Enter the following command.

```
(gcap-cli) show gcenter-ip
```

  - Validate.
    The system displays the IP address of the current GCenter: make sure it is the IP address of the GCenter to be paired.

```
Current GCenter IP: X.X.X.X
```

    If there is no paired Gcenter then the following message is displayed:

```
Current GCenter IP: None
```

  - Check that the IP address displayed is that of the GCenter to be paired. If there is a change, continue this procedure.
- To change the current version of the GCenter IP:
  - Enter the *set gcenter-ip* command followed by the GCenter IP setting.
    Example: set gcenter-ip 10.2.10.234
  - Validate.

  The system displays the new IP address of the GCenter.

~~~
```
Setting GCenter IP to 10.2.19.218
```
~~~

5.5.14.7 Procedure for declaring the GCap in the GCenter

- Obtain the FQDN (hostname.domain) of the GCap via the `show status` command.
- Connect to the GCenter via a web browser.
- Enter the FQDN (refer to the GCenter documentation).
- Click on the `Start Pairing` button.
  The One Time Password (OTP) is displayed at the top left of the web page.
  For example: pcmqsnf7iyo34ianzzi7gbgrr
- Copy the OTP.

5.5.14.8 Procedure for pairing the GCap and the GCenter

- Log on to the GCap CLI.
- Enter the following command.

```
(gcap-cli) pairing otp
```

- Insert the OTP previously generated by the GCenter after positioning the cursor after the text.

```
(gcap-cli) pairing otp pcmqsnf7iyo34ianzzi7gbgrr
```

- Validate.
  The GCap connects to the GCenter via the IP address of the GCenter set on the GCap earlier.
  The GCap then calculates the fingerprint using the FQDN of the GCap.
  It asks the user to compare it with the fingerprint calculated by the GCenter, which was itself calculated using the FQDN entered.
  The system displays the following message:

```
  Resetting any previous GCenter pairing...
  Generating IPSec certificates for the GCenter pairing...
  Probing for GCenter SSH fingerprint...

  Fingerprint for GCenter x is
  e655bc02553e2291a486a32bdce3943a315f830de70b2c627c39884e80
  0f08b2. Is it correct? (y/N)
```

- Compare the GCenter fingerprint retrieved by the GCap in the CLI with the one present in the `GCaps pairing..` part under the `GcenterSSH fingerprint` text in the GCenter web interface on the web browser.
  - If the fingerprints are not identical:
    - ∗ Check the GCenter IP address and the value entered in the GCap,
    - ∗ Check the GCap FQDN and the name entered in the GCenter.
  - If they are identical, answer **Y** and validate.

```
Sending OTP to GCenter...
Pairing up with the GCenter (IPSec certificates exchange)...
Pairing up with the GCenter (restarting IPSec tunnel)...
Pairing successful
```

- On the GCenter Web UI, check that the GCap is now Online in the `GCaps pairing and status` menu page.
  For more information please refer to the GCenter documentation.
  On the GCap, this information is visible with the `show status` command.

```
(gcap-cli) show status

GCAP Name         : host.domain
Version           : 2.5.3.105-xxx
Paired on GCenter : 10.2.19.128
Tunnel status     : Up
Detection Engine  : Container down
```

  The `Paired on GCenter` field takes:
  - The value `Not paired` when the GCap is not paired with the GCenter
  - The IP value of the GCenter when the GCap is paired with the GCenter

5.5.15 Managing the high availability of GCaps

5.5.15.1 Introduction

This procedure describes the high availability between 2 GCaps.
For more information, please refer to the paragraph on *high-availability*.

5.5.15.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show advanced-configuration high-availability status*
  - *set advanced-config high-availability*

5.5.15.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*).
- Stop the detection engine (see *monitoring-engine*).

5.5.15.4 Procedure for displaying the high availability status (GCap redundancy)

- Enter the following command.

```
(gcap-cli) show advanced-configuration high-availability status
```

- Validate.
  The system displays the high availability status with the following counters:
  - **status:** status of the GCap:
    - unhealthy: the GCap is not connected to the neighbouring GCap
    - Not configured: there is no high availability configured on this system
  - **paired GCap:** IPv6 address of the neighbouring GCap.
  - **leader:** election status among Leader/Follower.
  - **time since last status:** time since the last healthcheck of the neighbouring GCap.
  - **Leader since:** date when the GCap became the Leader.

  Situation where there is no high availability (redundancy of GCaps)
  Current high-availability status:
  - status: Not configured
  - paired gcap: Unknown
  - leader: Follower
  - time since last status: Unknown
  - Follower since: Unknown

  Situation of high availability (redundancy of GCaps) with loss of connection between GCaps
  Current high-availability status:
  - status: Operational [unhealthy]
  - paired gcap: fe80::233
  - leader: Leader
  - time since last status: Unknown
  - Leader since: 2022-01-21T15:35:09Z

5.5.15.5 Procedure of configuring high availability on the first GCap

- Enter the following command.

```
(gcap-cli) set advanced-configuration high-availability peer-ip fe80::XXX
public-ip fe80::YYY multicast-group ff02::200
peer-pubkey 2wtmY/oCaoUGreyr2CROnKAIoEgTXkSOedXlXDvUfBU=
shared-secret Xxf4fknh4KoOH2zgrI4Wyw==
```

Note:

Explanation of parameters:

- *set advanced-configuration high-availability* : order to configure high availability
- ***peer-ip fe80::XXX***
  [IPv6 address of neighbouring GCap among:]
  - **Link-local :** if the GCap are in the same subnet. Plage FE80::/10. Ex : FE80::100/64.
  - **ULA (Unique Local Address) :** if GCap are in different subnets. Plage FD00::/7. Ex : FD00::100/64.
  - **Global Unicast :** if GCap should communicate via the internet. Plage 2001::/3. Ex : 2001::1/64.
- ***public-ip fe80::YYY***
  [IPv6 address of neighbouring GCap among:]
  - **Link-local :** If the GCap are in the same subnet. Plage FE80::/10. Ex : FE80::100/64.
  - **ULA (Unique Local Address) :** if GCap are in different subnets. Plage FD00::/7. Ex : FD00::100/64.
  - **Global Unicast :** if GCap should communicate via the internet. Plage 2001::/3. Ex : 2001::1/64
- *multicast-group ff02::200* : multicast IPv6 address for communication between GCaps. Plage FF00::/8. Ex : FF02::200.
- *peer-pubkey* 2wtmYCaoUGreyr2CROnKAIoEgTXkSOedXlXDvUfBU= : Neighboring GCap public key visible via *show advanced-configuration high-availability pubkey* command
- *shared-secret* Xxf4fknh4KoOH2zgrI4Wyw== : 16 byte secret encoded in base64 which must be identical between the 2 GCaps.

- Validate.
  The system displays the result.

```
Updating HA configuration
High availability configuration successfully updated
```

5.5.15.6 Example of configuring high availability on the second GCap

- Enter the following command.

```
(gcap-cli) set advanced-configuration high-availability peer-ip fe80::YYY public-ip␣
↪fe80::XXX multicast-group ff02::200 peer-pubkey␣
↪xehXnrigZ0IZZEvWbWri8XegNh0KaAQk8vC6mKj27Ug= shared-secret Xxf4fknh4KoOH2zgrI4Wyw==
```

The system displays the result.

```
Updating HA configuration
High availability configuration successfully updated
```

5.5.15.7 Example of configuring high availability on each GCap

- Enter the following command.

  ```
  (gcap-cli) set advanced-configuration high-availability enable confirm
  ```

  The system displays the result.

  ```
  Interfaces naming rules updated, reloading configuration
  Operation successful.
  High availability configuration successfully updated
  ```

5.5.16 Optimising performance

5.5.16.1 Introduction

Performance optimisation can be achieved in the following ways:

- **Subject 1: adapting the GCap to the network characteristics**
  - Inconsistency between the MTU defined on the GCap and that of the captured frames.
    To modify the MTU see the Procedure for adjusting the size of the captured packet.
  - Check that the characteristics of the GCap, such as maximum throughput, number of sessions, etc., match those of the network to be monitored.
    For this purpose, consult the GCap datasheets.
- **Subject 2: optimising GCap resources**
  - The number of CPUs allocated to the detection engine is too low. The CPUs may be overloaded and potentially packets may go unanalysed and therefore dropped.
    To change this value, see the Procedure for assigning the number of CPUs to the detection engine.
  - Prefer using a TAP aggregator as opposed to the GCap "cluster" function.
    The solution using a TAP aggregator is preferable because it requires the least amount of GCap resources for the same flow.
- **Subject 3: optimising the network flow to be analysed**
  - One or more CPUs are being overloaded because there are too many packets being analysed.
    - To reduce the size of the captured network, it is possible to suppress the unnecessarily analysed flow.
    - To manage this packet filtering, see the procedure for defining flow filtering rules.
  - Only one CPU is being overloaded. In this case, the flow load is poorly distributed between the CPUs.
    - To change this, another rule can be defined or, more likely, an existing rule can be modified. A flow was defined but it was too large. It must therefore be subdivided so that each part is analysed by several CPUs.
    - To modify the rules, see the procedure for defining static packet filtering rules.
  - Change the analysed protocols.
    - To modify this list, this action must be performed on the paired GCenter.
      See the GCenter documentation for more information.
- **Subject 4: optimising the detection engine rules**
  The rules define:
  - Detection rules
  - File rebuilding rules
  - Rules defining thresholds or limits under the threshold heading
    See the GCenter documentation for more information.
- **Subject 5: monitoring the solution**
  A monitoring service known as Netdata, embedded in the GCenter, enables real-time information to be collected on the status of CPUs, load, disks, detection engines, and filtering.
  This feature is available at https://Nom_du_GCenter/gstats.
  On the GCap, Netdata enables more information on protocol counters, number of sessions, flows, and hash table status from 'Stats.log'.

5.5.16.2 Prerequisites

- **User:** setup
- **Commands used in this procedure:**
  - *show advanced-configuration mtu*
  - *set advanced-configuration mtu*
  - *show advanced-configuration cpu-config*
  - *set advanced-configuration cpu-config*
  - *show advanced-configuration packet-filtering*
  - *set advanced-configuration packet-filtering*
  - *show advanced-configuration load-balancing*
  - *set advanced-configuration load-balancing*

5.5.16.3 Preliminary operations

- Connect to the GCap (see *Procedure for connecting to the GCap via SSH*)
- Stop the detection engine (see *monitoring-engine*)

5.5.16.4 Procedure for adjusting the captured packet size

This setting enables adjusting the size of the captured packet to match the size of those packets circulating on the network.

> **Danger:**
>
> Load Balancing and XDP Filtering features are not supported if the MTU > 3000.

- Use the *show advanced-configuration mtu* command to display the MTU value in bytes of all enabled network interfaces
- Use the *set advanced-configuration mtu* command to change the number of dedicated CPUs

---

5.5.16.5 Procedure for assigning the number of CPUs to the detection engine

> **Astuce:**
>
> Dedicate the maximum number of CPUs present to the detection engine without exceeding 80% of the CPUs.
> This is done when the CPUs dedicated to the detection engine are overloaded (use the *show cpus* command).

- Use the x command to display the number of CPUs dedicated to the Sigflow detection engine
- Use the *set advanced-configuration cpu-config* command to change the number of dedicated CPUs

---

5.5.16.6 Procedure for defining flow filtering rules

Astuce:

The CPU(s) present are overloaded and part of the flow cannot be analysed, a number of packets are dropped:

- To view the CPU overload, use the *show cpus* command
- To view the number of dropped packets per cpux core, use the *show health* command, sofnet counter details- Statistics on received packets by CPU core.

Certain parts of the captured flow cannot be detected or reconstructed: for example, encrypted flows.

If nothing is done, the system will monopolise resources to achieve a result known in advance.

To avoid this, it is possible to create rules to filter the flow to be captured.

- Use the *`show advanced-configuration packet-filtering`* command to display static packet filter rules.
- Use the *`set advanced-configuration packet-filtering`* command to specify static rules for filtering flows captured by the capture interfaces.

5.5.16.7 Procedure for load balancing configuration from the monx capture interface

Astuce:

In this case where there is an incorrect distribution of the flow load between the CPUs, it is possible to define a rule or most certainly modify an existing rule.

A flow was defined but it was too large. It must therefore be subdivided so that each part is analysed by several CPUs using load balancing methods (algorithm)..

- Use the *`show advanced-configuration load-balancing`* command to display the load balancing configuration from the monx capture interface listed to the GCap CPUs.
- Use the *`set advanced-configuration load-balancing`* command to change the load of the capture interfaces.

5.5.16.8 Procedure for optimising the detection engine rules

Astuce:

The rules for the detection engine can be defined:

- Locally on the GCap,
- On the GCenter.

It is on these 2 applications that they must be modified to optimise them.

Moreover, if the current configuration is multi-tenant then the same rules are applied on the interfaces. This may not be optimised!

- Use the `show advanced-configuration local-rules` command to display:

  - Under Rules heading: the local Sigflow rules, i.e.:
    - Detection rules
    - File rebuilding rules
  - Under the Threshold heading:
    - Thresholds or limits defined by the keyword "threshold"
    - Deletion rules defined by the keyword "suppress"
- Use the `set advanced-configuration local-rules` command to modify the local rules of the GCap probe.
- Optimise the ruleset sent from the GCenter. To do this, use the GCenter.

Chapter 6

CLI

6.1 Overview of the CLI

6.1.1 Introduction to the CLI

The Command Line Interface (CLI) is the means used to administer and configure the GCap.
It is therefore necessary to enter commands in text mode following the command prompt.

6.1.2 Overview of the command prompt

```
[Monitoring DOWN] gcap-name (gcap-cli)
```

It includes:

- The status of the Sigflow detection engine (here `Monitoring down`)
- The name of the GCap (here `gcap-name`)
- The level information in the tree:
  - Here (`gcap-cli`): means the command prompt is at the root of the commands
  - For example (`gcap-cli show`): means the command prompt is in the `show` set

6.1.3 Accessible commands grouped by set

The commands are grouped by set (show, set, etc.).
The detailed list of commands is provided in the CLI section.

The set...	is used to...
show	display the system configuration
set	modify the system configuration
services	manage GCap services
system	manage system operations

These sets are accessible from the root.

Note:

The set of commands in the GCap CLI is calculated dynamically. The list of commands depends on:

- The current user type
- The status of the GCap

This information can be found in the documentation.

Note:

- If a command is entered in the wrong set, or
- If the access level is not the correct one

... then the command is not recognised and the message `Command `X` is not recognised` is displayed.

Note:

User type or context elements are specified where necessary.

6.1.4 Directly accessible commands

The commands below are directly accessible:

Use the command...	to...
monitoring engine	manage the detection engine
pairing	pairing the GCap and GCenter
help	to obtain help with the available commands
colour	enable or disable colours for the current CLI session
exit	return to the root of the CLI or exit the CLI

6.1.5 Completion

To complete the name of a command or an argument, it is possible to use the completion, i.e.:

- Start by entering a command, then
- Use the tab key on the keyboard

The system proposes the possible values.
Example: by asking for a completion on the command below, the system displays the supported values of `set keymap`:

```
(gcap-cli) set keymap

fr us
```

6.1.6 Navigating in the command tree

6.1.6.1 To go from the root to a set

To access the commands of a set from the root, enter the name of the set.
Example:

```
(gcap-cli)
```

- Enter the `show` command.

```
(gcap-cli show)
```

The prompt changes to inform the user that the set has changed.
Now the commands of set `show` are accessible.
Commands can also be accessed directly from the prompt **(gcap-cli)** by issuing the complete command: for example `show alerts` for the `alerts` command of set `show`.

6.1.6.2 To return to the root

To exit the current set and return to the **root**, enter the `exit` command.
Example:

```
(gcap-cli show)
```

Only the commands in the show set are accessible.

- Enter the `exit` command.

  ```
  (gcap-cli)
  ```

  The prompt changes to inform the user that the command prompt is at the root.
  At this level, all command sets are accessible.
  The **CTRL** + **D** shortcut enables calling the `exit`command.

6.1.7 Launching a command

A command can be launched in two different ways:

- Either with only the command name but the command prompt must be at the set level
- Or from the root but the name of the set must be entered followed by the name of the command

6.1.7.1 Example of launching from the root for the `show alerts` command

```
(gcap-cli)
```

- Enter the `show alerts` command then validate.

6.1.7.2 Example of launching the `show alerts` command from the `show` set

```
(gcap-cli show)
```

- Enter the `alerts` command then validate.

6.1.8 Obtaining information on commands via Help

To receive help on the available commands, it is possible to use the `?` or `help` command.
To obtain help with a specific command, it is possible to:

- Prefix it with `help` (example `help show config-files`)
- Suffix the command with `?` (example `show config-files ?`)

For more information on assistance, see the paragraph on *help*.

6.1.9 Exit

If the GCap interactive CLI is used, the `exit` command must be used to return to the root of the command tree. For more information on the command, see the paragraph on *exit*.

6.2 cli

6.2.1 show

6.2.1.1 alerts

6.2.1.1.1 Introduction

The `alerts` command of the `show` subgroup enables monitoring of the alerts issued by Sigflow.

6.2.1.1.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.1.3 Command

```
show alerts
```

6.2.1.1.4 Example

- Enter the following command.

```
(gcap-cli) show alerts
```

- Validate.

6.2.1.2 bruteforce-protection

6.2.1.2.1 Introduction

The `bruteforce-protection` command in subgroup `show` enables displaying the system policy for protecting against brute force attacks.

6.2.1.2.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.1.2.3 Command

`show bruteforce-protection`

6.2.1.2.4 Example for showing the current system policy for protecting against brute force attacks

- Enter the following command.

```
(gcap-cli) show bruteforce-protection
```

- Validate.
  The system displays the following information.

```
Current bruteforce protection rules:
    - Max tries: 3
    - Lock duration: 120s
```

6.2.1.3 bypassed-flows

6.2.1.3.1 Introduction

This command has been removed since version 2.5.3.105.

6.2.1.4 clusters

6.2.1.4.1 Introduction

The `clusters` command in subgroup `show` enables the aggregation of the capture and monitoring interfaces `mon` and their configurations to be displayed.
For more information on aggregation, see the paragraph *Capture and Monitoring Interfaces `monx` between TAP and GCap: Aggregation Capability*.

> **Note:**
>
> This functionality is necessary if the qualified Test Access Port (TAP) present in the architecture does not provide the interface aggregation functionality.

6.2.1.4.2 Prerequisites

- **User:** setup
- **Dependencies:** activate at least two capture interfaces

6.2.1.4.3 Command

```
show clusters
```

6.2.1.4.4 Example for implementing interface aggregation

Refer to the *Procedure for Managing Capture Interface Aggregation*.

6.2.1.4.5 Example for displaying the interface cluster

- Enter the following command.

```
(gcap-cli) show clusters
```

- Validate.
  The system displays the result.

```
Name      State     Description       Interfaces
cluster0  Disabled  test              mon0, mon1
```

  The system displays the information of the existing aggregations and for each:
  - name
  - status
  - description
  - interfaces component

Note:

If the message *No network cluster defined* is displayed, check the prerequisites before entering the command.

6.2.1.5 compatibility-mode

6.2.1.5.1 Introduction

The `compatibility-mode` command of the `show` subgroup enables displaying the current compatibility mode to interact with GCenter.
The compatibility mode will affect the available functionality of GCap.
Several compatibility modes are available:

- 2.5.3.100: GCenter 2.5.3.100 and below
- 2.5.3.101: GCenter 2.5.3.101
- 2.5.3.102+: GCenter 2.5.3.102 and above

The current mode must be selected based on the current GCap and GCenter versions.
For more information, refer to the *compatibility table*.

Note:

The compatibility mode for GCenter version 2.5.3.100 and below is deprecated.

6.2.1.5.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.1.5.3 Command

```
show compatibility-mode
```

6.2.1.5.4 Example for displaying the current compatibility mode

- Enter the following command.

```
(gcap-cli) show compatibility-mode
```

- Validate.
  The system displays the current compatibility mode.

```
Current compatibility mode: 2.5.3.102+
 For a GCap V106 or V107 version, this is the recommended mode.

---
```

6.2.1.6 config-files

6.2.1.6.1 Introduction

The `config-files` command of the `show` subgroup enables displaying:

- The detailed configuration of the Sigflow detection engine using the `config-files suricata-config` command
- The rules transmitted by GCenter to the Sigflow engine:
  - rules-scirius: the scirius detection rules
  - rules-files: the file rebuilding rules
  - threshold.
    In this category, the following are defined:
    - alert threshold rules (detection rules)
      For example:
      No more alerts are sent beyond a certain value (notion of limit)
      Or conversely, validate alerts above a certain value (notion of threshold)
    - Limiting detection rules, for example not applying a rule to a specific IP address

It is only possible to display the rules of the configured tenant.

6.2.1.6.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:**
  - Pair the GCap and GCenter
  - Send rulesets from GCenter to the GCap

6.2.1.6.3 Command

`show config-files {suricata-config|rules-scirius|rules-files|threshold} [TENANT]`

The `show config-files` command must be followed by:

- The name of the configuration file:
  - `suricata-config` for the Sigflow configuration
  - `rules-scirius` for scirius rules for detection used by Sigflow
  - `rules-files` for file reconstruction rules used by Sigflow
  - `threshold` for threshold rules, limits, and deletion rules
- The TENANT parameter which can take the following values:
  - Multi-tenant by int: {mon0|mon1|mon2|mon3|monvirt}
  - Multi-tenant by vlan:
    - default
    - VLAN X
    - VLAN X Y

6.2.1.6.4 Example to display the scirius rules for detection, in single tenant mode

- Enter the following command.

```
(gcap-cli) show config-files rules-scirius
```

- Validate.
  The system displays the result.

```
# Rules file for ** generated by Scirius at 2022-05-30 12:41:33.634390+00:00

  alert dns any any -> any any (msg:"[ TEST AUTO ] ALERT DNS UDP";sid:12345600;
↪priority:2;)
```

The file displays:

  - First the generation date
  - Then, in each paragraph, a rule is defined

For more information on the syntax of the rules, please refer to the GCenter documentation.

6.2.1.6.5 Example to display the scirius rules for detection, in multi-tenant mode for the `mon0` interface

- Enter the following command.

```
(gcap-cli) show config-files rules-scirius mon0
```

- Validate.
  The system displays the result (see example above).

Note:

If the following message is displayed "Command *show config-files rules-scirius mon0* is not recognized", check the configuration (multi-tenant with `mon0` interface).

6.2.1.6.6 Example to display the scirius rules in multi-tenant mode for vlan 10

- Enter the following command.

```
(gcap-cli) show config-files rules-scirius VLAN 10
```

- Validate.
  The system displays the result (see example above).

Note:

If the following message is displayed "Command *show config-files rules-scirius VLAN 10* is not recognised", check the configuration (multi tenant with VLAN 10).

6.2.1.6.7 Example of displaying thresholds, limits, and deletion rules

- Enter the following command.

```
(gcap-cli) show config-files threshold
```

- Validate.
  The system displays the result.

```
  suppress gen_id 1, sig_id 2435, track by_src, ip 10.10.10.10
  threshold gen_id 1, sig_id 2435, type limit, track by_src, count 1, seconds 60)
```

The file displays:
  - thresholds or limits defined by the keyword "threshold"
  - deletion rules defined by the keyword "suppress"

6.2.1.7 cpus

6.2.1.7.1 Introduction

The `cpus` command of the `show` subgroup enables listing all the CPUs available on the GCap and their usage percentage.

6.2.1.7.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.7.3 Command

```
show cpus
```

6.2.1.7.4 Example for displaying the CPUs and their usage percentage

- Enter the following command.

```
(gcap-cli) show cpus
```

- Validate.
  The system displays the current information.

```
Linux 5.10.36-grsec (GCap)          19/01/22          _x86_64_        (12 CPU)

08:54:36     CPU     %usr    %nice     %sys %iowait     %irq    %soft   %steal   %guest   %gnice␣
↪   %idle
08:54:37     all     5.21     0.00     1.43     0.00     0.00     0.00     0.00     0.00     0.00␣
↪   93.36
08:54:37       0     2.00     0.00     1.00     0.00     0.00     0.00     0.00     0.00     0.00␣
```

(suite sur la page suivante)

(suite de la page précédente)

```
↪  97.00
08:54:37       1    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00␣
↪ 100.00
08:54:37       2    1.01    0.00    1.01    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  97.98
08:54:37       3    1.01    0.00    1.01    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  97.98
08:54:37       4    1.01    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  98.99
08:54:37       5    0.00    0.00    1.01    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  98.99
08:54:37       6    1.01    0.00    1.01    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  97.98
08:54:37       7   52.00    0.00    7.00    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  41.00
08:54:37       8    1.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  98.00
08:54:37       9    1.01    0.00    1.01    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  97.98
08:54:37      10    1.01    0.00    2.02    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  96.97
08:54:37      11    1.02    0.00    1.02    0.00    0.00    0.00    0.00    0.00    0.00␣
↪  97.9
```

- Press **CTRL** + **C** to stop.
  The system calculates the averages and displays them.

6.2.1.8 datetime

6.2.1.8.1 Introduction

The `datetime` command of the `show` subgroup enables the date and time of the GCap to be displayed in `YYYY-MM-DD HH:MM:SS` format.

6.2.1.8.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.1.8.3 Command

```
show datetime
```

6.2.1.8.4 Example of displaying the date and time of the GCap

- Enter the following command.

```
(gcap-cli) show datetime
```

- Validate.
  The system displays the current information.

```
Current datetime is 2022-01-26 16:10:44
```

6.2.1.9 eve-stats

6.2.1.9.1 Introduction

The `eve-stats` command of the `show` subgroup enables displaying the Sigflow (*monitoring-engine*) statistics.

6.2.1.9.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.9.3 Command

```
show eve-stats
```

6.2.1.9.4 Example

- Enter the following command.

```
(gcap-cli) show eve-stats
```

- Validate.
  The system displays the following information:
  - counter `Alerts` - Number of Sigflow alerts found
  - the counters `Files` - Files extracted by Sigflow
  - the counters `Codebreaker samples` - Files analysed by Codebreaker
  - counters `Protocols` - List of protocols seen by Sigflow
  - counters `Detection Engine Stats` - Sigflow statistics (*monitoring-engine*)

6.2.1.9.4.1 Counter `Alerts` details - Number of Sigflow alerts found

Example:

... Alerts: 0 ...

6.2.1.9.4.2 Detail of counters `Files` - Files extracted by Sigflow

- `Observed` - Number of files observed by Sigflow.
- `Extracted` - Number of files extracted by Sigflow.
- `Uploaded` - Data sent to GCenter.
  - `Metadata` - Number of metadata sent to GCenter.
  - `File` - Number of files sent to GCenter.

Example:

... Files: Observed: 6011816 Extracted: 0 Uploaded: Metadata: 0 File: 0 ...

6.2.1.9.4.3 Counter `Codebreaker samples` details - Files analysed by Codebreaker

- `Extracted` - Number of extracted files received by Codebreaker.
- `Uploaded` - Data on files received by Codebreaker on GCenter.
  - `Shellcodes` - Data on *shellcodes*.
    - `Plain` - *Shellcodes* detected without encoding.
    - `Encoded` - *Shellcodes* detected with encoding.
  - `Powershell` - Number of malicious Powershell scripts detected.

Example:

... Codebreaker samples: Extracted: 0 Uploaded: Shellcodes: Plain: 0 Encoded: 0 Powershell: 0 ...

6.2.1.9.4.4 Detail of counters `Protocols` - List of protocols seen by Sigflow

`<protocole>` Number of events observed by Sigflow concerning protocol e.g *HTTP*, *SMB*, and others.

Example:

```
    Protocols:
      DHCP:     0
      DNP3:     0
      DNS:      0
      FTP:      0
      HTTP:     6537929
      HTTP2:    0
      IKEv2:    0
      KRB5:     0
      MQTT:     0
      NETFLOW:  0
      NFS:      0
      RDP:      0
      RFB:      0
      SIP:      0
```

(suite sur la page suivante)

(suite de la page précédente)

```
    SMB:       0
    SMTP:      0
    SNMP:      0
    SSH:       0
    TFTP:      0
    TLS:       0
    Tunnels:   0
```

6.2.1.9.4.5 Detail of counters `Detection Engine Stats` - Sigflow statistics (*monitoring-engine*)

- `Events` - Data on events observed by Sigflow
  - `Total` - Total number of events
  - `Stats` - Number of statistics generated
- `Capture`
  - `Received` - Number of packets captured
  - `Dropped` - Number of packets ignored
- `Rules` - Sigflow rules data
  - `Loaded` - Number of rules loaded and validated
  - `Invalid` - Number of rules that could not be loaded
- `TCP`
  - `SYN` - Number of *SYN* observed by Sigflow.
  - `SYN/ACK` - Number of *SYN/ACK* observed by Sigflow.
  - `Sessions` - Number of *TCP* sessions observed by Sigflow.
- `Flow`
  - `TCP` - Number of *TCP* sessions observed
  - `UDP` - Number of *UDP* sessions observed
  - `SCTP` - Number of *SCTP* sessions observed
  - `ICMPv4` - Number of *ICMPv4* messages observed
  - `ICMPv6` - Number of *ICMPv6* messages observed
  - `Timeouts` - Statistics on *TCP* session expirations
    - `New` - Number of new windows *TCP*
    - `Established` - Number of windows established
    - `Closed` - Number of windows closed
    - `Bypassed` - Number of windows ignored

  Example :

```

  Detection Engine Stats:
    Events:
      Total:      12551855
      Stats:      2110

    Capture:
      Received:   153439718
      Dropped:    60964966

    Rules:
      Loaded:     78
      Invalid:    28

    TCP:
      SYN:        10274277
      SYN/ACK:    10274629
```

(suite sur la page suivante)

(suite de la page précédente)

```
        Sessions:  10273062

      Flows:
        TCP:       12067611
        UDP:       0
        SCTP:      0
        ICMPv4:    0
        ICMPv6:    0

        Timeouts:
            New:          0
            Established:  0
            Closed:       0
            Bypassed:     0
```

6.2.1.10 gcenter-ip

6.2.1.10.1 Introduction

The `gcenter-ip` command of the `show` subgroup enables displaying the IP address of the GCenter with which the GCap is paired.

6.2.1.10.2 Prerequisites

- **User:** setup
- **Dependencies:**
  - the detection engine must be switched off
  - a GCenter must be paired

6.2.1.10.3 Command

```
show gcenter-ip
```

6.2.1.10.4 Example

- Enter the following command.

  ```
  (gcap-cli) show gcenter-ip
  ```

- Validate.
  The system displays the IP address of the paired GCenter.

  ```
  Current GCenter IP: X.X.X.X
  ```

  If there is no paired Gcenter then the following message is displayed:

```
Current GCenter IP: None
```

6.2.1.11 health

6.2.1.11.1 Introduction

The `health` command of the `show` subgroup enables displaying statistics and the health information of the GCap.

6.2.1.11.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:** N/A

6.2.1.11.3 Command

`show health`

6.2.1.11.4 Example

- Enter the following command.

```
(gcap-cli) show health
```

- Validate.
  The system displays the following information:
  - `block` counters - Mass storage statistics
  - `cpu_stats` counters - Processor statistics
  - `disks` counters - Mount point occupancy statistics
  - `emergency` counters - GCap emergency mode information
  - `gcenter` counters - Paired GCenter information
  - `high_availability` counters - High Availability (*HA*) information
  - `interfaces` counters - Statistics on network interfaces
  - `loadavg` counters - Statistics on the average load of the GCap
  - `meminfo` counters - Statistics on the RAM
  - `numastat` counters - Non Uniform Memory Access (NUMA) node statistics
  - `quotas` counters - Quota Information
  - `sofnet` counters - Statistics on received packets according to processor cores
  - `suricata` counters - Sigflow (*monitoring-engine*) information
  - `systemd` counters - System initialisation information
  - `uptime` counters - Uptime
  - `virtualmemory` counters - Swap space information (*swap*)

6.2.1.11.4.1 `block` counters details - Mass storage statistics

- `sdN` - Disk statistics N where N is a letter of the alphabet
  - `read_bytes` - Bytes read since startup
  - `written_bytes` - Bytes written since startup

  Example:

```
{
 "block": {
    "sda": {
        "read_bytes": 302867968,
        "written_bytes": 4837645312
    },
    "sdb": {
        "read_bytes": 3894272,
        "written_bytes": 4096
    }
},
```

6.2.1.11.4.2 `cpu_stats` counter details - CPU statistics

- `cpus` - CPU usage statistics
  - `cpu` - Overall core usage statistics
  - `cpuX` - CPU X core statistics
    - `idle` - Elapsed time doing nothing in milliseconds
    - `iowait` - Elapsed time waiting for disk operations in milliseconds
    - `irq` - Elapsed time on material IRQs
    - `nice` - Time elapsed in user space on low priority processes in milliseconds
    - `softirq` - Elapsed time on hardware IRQs in milliseconds
    - `system` - Elapsed time in kernel space in milliseconds
    - `user` - Elapsed time in user space in milliseconds
  - `interrupts` - Number of interrupts since startup
  - `processes_blocked` - Number of blocked or *dead* processes
  - `processes_running` - Number of running processes

  Example:

```
"cpu_stats": {
    "cpus": {
        "cpu": {
            "idle": 961816208,
            "iowait": 11419,
            "irq": 0,
            "nice": 0,
            "softirq": 397899,
            "system": 21788203,
            "user": 50806194
        },
        "cpu0": {
            "idle": 79960857,
            "iowait": 985,
            "irq": 0,
            "nice": 0,
            "softirq": 234748,
```

(suite sur la page suivante)

(suite de la page précédente)

```
            "system": 1795880,
            "user": 4357374
        },
        "cpu1": {
            "idle": 80166571,
            "iowait": 951,
            "irq": 0,
            "nice": 0,
            "softirq": 88078,
            "system": 1830370,
            "user": 4138182
        }
    },
    "interrupts": 12942835029,
    "processes_blocked": 0,
    "processes_running": 1
},
```

6.2.1.11.4.3 `disks` counters details - Mount point occupancy statistics

- `/mountpoint/path` - Mount point path
  - `block_free` - Number of free *blocks*
  - `block_total` - Total number of blocks
  - `inode_free` - Number of remaining inodes
  - `inode_total` - Total number of *inodes*

  Example:

```
"disks": {
    "/": {
        "block_free": 247909,
        "block_total": 249830,
        "inode_free": 64258,
        "inode_total": 65536
    },
    "/data": {
        "block_free": 7150076,
        "block_total": 7161801,
        "inode_free": 1827417,
        "inode_total": 1827840
    },
},
```

6.2.1.11.4.4 `emergency` Counters details - GCap emergency mode information

- `emergency_active` - Active or inactive status of the *emergency mode*

Example:

```
 "emergency": {
     "emergency_active": false
 },
```

6.2.1.11.4.5 `gcenter` Counters details - Paired GCenter information

- `chronyc_sync` - Status of the NTP synchronisation with the GCenter
- `reachable` - GCenter reachable or not (false)

Example:

```
"gcenter": {
    "chronyc_sync": false,
    "reachable": false
},
```

6.2.1.11.4.6 `high_availability` counters details - High Availability (*HA*) information

- `healthy` - *HA* health status
- `last_status` - Last known *HA* status
- `last_transition` - Date of last known *HA* status change in *ISO8601* format
- `leader` - True for a GCap *leader*, false for a GCap *follower*
- `status` - Active or inactive (false) status of the *HA*

Example:

```
   "high_availability": {
       "healthy": false,
       "last_status": -1,
       "last_transition": "0001-01-01T00:00:00Z",
       "leader": false,
       "status": false
   },
```

6.2.1.11.4.7 `interfaces` counter details - Statistics on network interfaces

- `bond0` - Name of the network interface
  - `rx_bytes` - Number of bytes received
  - `rx_drop` - Number of bytes lost in reception
  - `rx_errs` - Number of invalid bytes received
  - `rx_packets` - Total number of packets received from this interface
  - `tx_bytes` - Number of bytes sent
  - `tx_drop` - Number of bytes lost while sending
  - `tx_errs` - Number of invalid bytes sent

  - `tx_packets` - Total number of packets sent from this interface

Example:

```
    "interfaces": {
        "bond0": {
            "rx_bytes": 0,
            "rx_drops": 0,
            "rx_errs": 0,
            "rx_packets": 0,
            "tx_bytes": 0,
            "tx_drops": 0,
            "tx_errs": 0,
            "tx_packets": 0
        },
        "gcp0": {
            "rx_bytes": 138433006,
            "rx_drops": 82901,
            "rx_errs": 0,
            "rx_packets": 2143236,
            "tx_bytes": 796294,
            "tx_drops": 0,
            "tx_errs": 0,
            "tx_packets": 3635
        },
        "gcp1": {
            "rx_bytes": 137642525,
            "rx_drops": 82902,
            "rx_errs": 0,
            "rx_packets": 2135060,
            "tx_bytes": 0,
            "tx_drops": 0,
            "tx_errs": 0,
            "tx_packets": 0
        }
    },
```

6.2.1.11.4.8 `loadavg` counter details - Statistics on the average load of the GCap

- `active_processes` - Number of processes started
- `load_average_15_mins` - Average load over the last fifteen minutes
- `load_average_1_min` - Average load over the last minute
- `load_average_5_mins` - Average load over the last five minutes
- `running_processes` - Number of running processes

Example:

```
    "loadavg": {
        "active_processes": 561,
        "load_average_15_mins": 0.99,
        "load_average_1_min": 0.67,
        "load_average_5_mins": 1,
        "running_processes": 2
    },
```

6.2.1.11.4.9 `meminfo` counter details - Statistics on the RAM

- `available` - Total physical memory in kilobytes
- `buffers` - Memory used by disk operations in kilobytes
- `cached` - Memory used by the cache in kilobytes
- `dirty` - Memory used by pending write operations in kilobytes
- `free` - Unused memory in kilobytes
- `hugepages_anonymous` - Number of anonymous transparent *huge pages* used
- `hugepages_free` - Number of available transparent *huge pages*
- `hugepages_reserved` - Number of reserved transparent *huge pages*
- `hugepages_shmem` - Number of shared transparent *huge pages*
- `hugepages_surplus` - Number of extra transparent *huge pages*
- `hugepages_total` - Total number of *huge pages*
- `kernel_stack` - Memory used by kernel stack allocations in kilobytes
- `page_tables` - Memory used for page management in kilobytes
- `s_reclaimable` - Cache memory that can be reallocated in case of memory shortage in kilobytes
- `shmem` - Memory used by shared pages in kilobytes
- `slab` - Memory used by kernel data structures in kilobytes
- `swap_cached` - Memory used by the swap cache in kilobytes
- `swap_free` - Available memory in swap in kilobytes
- `swap_total` - Total swap memory in kilobytes
- `total` - Total memory in kilobytes
- `v_malloc_used` - Memory used by large memory areas allocated by the kernel

For more information, please refer to this documentation `meminfo`.

Example:

```
    "meminfo": {
        "available": 13608896,
        "buffers": 380932,
        "cached": 1155824,
        "dirty": 28,
        "free": 13128080,
        "hugepages_anonymous": 423936,
        "hugepages_free": 0,
        "hugepages_reserved": 0,
        "hugepages_shmem": 0,
        "hugepages_surplus": 0,
        "hugepages_total": 0,
        "kernel_stack": 9152,
        "page_tables": 8400,
        "s_reclaimable": 43168,
        "shmem": 794564,
        "slab": 210008,
        "swap_cached": 0,
        "swap_free": 16777212,
        "swap_total": 16777212,
        "total": 15977468,
        "v_malloc_used": 66592
    },
```

6.2.1.11.4.10 `numastat` counter details- Non Uniform Memory Access (NUMA) node statistics

- `nodes` - List of NUMA nodes
  - `nodeX` - NUMA X node statistics
    - `interleave_hit` - Interleaved memory successfully allocated in this node
    - `local_node` - Memory allocated in this node while a process was running on it
    - `numa_foreign` - Memory planned for this node, but currently allocated in a different node
    - `numa_hit` - Memory successfully allocated in this node as expected
    - `numa_miss` - Memory allocated in this node despite process preferences. Each numa_miss has a numa_foreign in another node
    - `other_node` - Memory allocated in this node while a process was running in another node

Example:

```
    "numastat": {
        "nodes": {
            "node0": {
                "interleave_hit": 3871,
                "local_node": 4410557829,
                "numa_foreign": 0,
                "numa_hit": 4410454203,
                "numa_miss": 0,
                "other_node": 14170
            },
            "node1": {
                "interleave_hit": 3869,
                "local_node": 4224990850,
                "numa_foreign": 0,
                "numa_hit": 4224964539,
                "numa_miss": 0,
                "other_node": 21531
            }
        }
    },
```

`quotas` counter details - Quota statistics by category

- `quotas` - Quota list
  - `by_gid` - Statistics sorted by group (gid identifier)
  - `by_prj` - Statistics sorted by project (prj identifier)
  - `by_uid` - Statistics sorted by user (uid identifier)

In each category, the following counters are displayed:

- `block_grace` - Grace time for blocks
- `block_hard_limit` - Hardware limit of blocks.
  Sets an absolute limit for the use of space.
  The user cannot exceed this limit.
  Beyond this limit, writing to this file system is forbidden.
- `block_soft_limit` - Software block limit
  Specifies the maximum amount of space a user can occupy on the file system.
  If this limit is reached, the user receives warning messages that the quota assigned to them has been exceeded.
  If its use is combined with the timeframes (or grace period), when the user continues to exceed the software limit after the grace period has elapsed, then he finds himself in the same situation as in the reaching of a hard limit.
- `block_used` - Number of blocks used
- 'file_grace` - Grace time for files
- `file_hard_limit` - Hardware file limit
  Sets an absolute limit for the use of space.

The user cannot exceed this limit.
Beyond this limit, writing to this file system is forbidden.

- `file_soft_limit` - Software file limit
  Specifies the maximum amount of space a user can occupy on the file system.
  If this limit is reached, the user receives warning messages that the quota assigned to them has been exceeded.
  If its use is combined with the timeframes (or grace period), when the user continues to exceed the software limit after the grace period has elapsed, then he finds himself in the same situation as in the reaching of a hard limit.
- `file_used`` - Number of files used

Exemple :

```
    "quotas": {
        "by_gid": {
            "0": {
                "block_grace": "0",
                "block_hard_limit": "0",
                "block_soft_limit": "0",
                "block_used": "2148952",
                "file_grace": "0",
                "file_hard_limit": "0",
                "file_soft_limit": "0",
                "file_used": "177"
            },
            "10012": {
                "block_grace": "0",
                "block_hard_limit": "0",
                "block_soft_limit": "0",
                "block_used": "5216",
                "file_grace": "0",
                "file_hard_limit": "0",
                "file_soft_limit": "0",
                "file_used": "295"
            },
            }
        },
        "by_prj": {
            "0": {
                "block_grace": "0",
                "block_hard_limit": "0",
                "block_soft_limit": "0",
                "block_used": "51600",
                "file_grace": "0",
                "file_hard_limit": "0",
                "file_soft_limit": "0",
                "file_used": "225"
            },
            "1": {
                "block_grace": "0",
                "block_hard_limit": "7980499",
                "block_soft_limit": "7980499",
                "block_used": "2101904",
                "file_grace": "0",
                "file_hard_limit": "1000",
                "file_soft_limit": "1000",
                "file_used": "43"
            },
            }
```

(suite sur la page suivante)

(suite de la page précédente)

```
        },
        "by_uid": {
            "0": {
                "block_grace": "0",
                "block_hard_limit": "0",
                "block_soft_limit": "0",
                "block_used": "2153356",
                "file_grace": "0",
                "file_hard_limit": "0",
                "file_soft_limit": "0",
                "file_used": "269"
            },
            "10012": {
                "block_grace": "0",
                "block_hard_limit": "0",
                "block_soft_limit": "0",
                "block_used": "1032",
                "file_grace": "0",
                "file_hard_limit": "0",
                "file_soft_limit": "0",
                "file_used": "258"
            },
        }
```

Example below is without defined limit: the value "0" indicates that there is no defined value for limits and grace times.

```
"10012": {
     "block_grace": "0",
     "block_hard_limit": "0",
     "block_soft_limit": "0",
     "block_used": "1032",
     "file_grace": "0",
     "file_hard_limit": "0",
     "file_soft_limit": "0",
     "file_used": "258"
},
```

6.2.1.11.4.11 `sofnet` counter details - Statistics on received packets according to processor cores

- `cpus` - Usage statistics per CPU
  - `cpuX` - CPU X core statistics
    - `backlog_len` -
    - `dropped` - Number of packets dropped
    - `flow_limit_count` - Number of times the throughput limit was reached
    - `processed` - Number of packets processed
    - `received_rps` - Number of times the CPU was woken up
    - `time_squeeze` - Number of times the thread could not process all the packets in its backlog within the budget
  - `summed` - Overall core usage statistics
    - `backlog_len` -
    - `dropped` - Number of packets dropped
    - `flow_limit_count` - Number of times the throughput limit was reached
    - `processed` - Number of packets processed

    - `received_rps` - Number of times the CPU was woken up
    - `time_squeeze` - Number of times the thread could not process all the packets in its backlog within the budget

Example:

```
    "softnet": {
        "cpus": {
            "cpu0": {
                "backlog_len": 0,
                "dropped": 0,
                "flow_limit_count": 0,
                "processed": 448550,
                "received_rps": 0,
                "time_squeeze": 2
            },
            "cpu1": {
                "backlog_len": 0,
                "dropped": 0,
                "flow_limit_count": 0,
                "processed": 36250,
                "received_rps": 0,
                "time_squeeze": 0
            }
        },
        "summed": {
            "backlog_len": 0,
            "dropped": 0,
            "flow_limit_count": 0,
            "processed": 5239450,
            "received_rps": 0,
            "time_squeeze": 27
        }
    },
```

6.2.1.11.4.12 `Sigflow` counter details - Sigflow (*monitoring-engine*) information

`detailed_status` - Sigflow container status

- `up` - Status of Sigflow and the detection engine

detailed_status + status "up"	signification
status "Container down" + "up" false	status engine off
status "Container down" + "up" true	impossible status: device cannot be rotated in a disabled container
status "Container UP" + "up" false	unstable status: call GATEWATCHER support
status "Container UP" + "up" true	status engine on

Example:

```
    "suricata": {
        "detailed_status": "Container down",
        "up": false
    },
```

6.2.1.11.4.13 `systemd` counter details - System initialisation information

- `failed_services` - List of failed services reported by `systemctl --failed`.

Example:

```
    "systemd": {
        "failed_services": [ "netdata.service" ]
    },
```

6.2.1.11.4.14 `uptime` counter details - Uptime

- `up_seconds` - Number of seconds since start-up.

Example:

```
    "uptime": {
        "up_seconds": 874179.8
    },
```

6.2.1.11.4.15 `virtualmemory` counter details - Swap space information (*swap*)

- `disk_in`: Number of pages saved to disk since start-up.
- `disk_out` - Number of pages out of disk since start-up.
- `pagefaults_major` - Number of *page faults* per second.
- `pagefaults_minor` - Number of *page faults* per second to load a memory page from disk to RAM.
- `swap_in` - Number of kilobytes the system swapped from disk to RAM per second.
- `swap_out` - Number of kilobytes the system swapped from RAM to disk per second.

Example:

```
    "virtualmemory": {
        "disk_in": 307828,
        "disk_out": 4724267,
        "pagefaults_major": 1210,
        "pagefaults_minor": 14233474300,
        "swap_in": 0,
        "swap_out": 0
    }
```

6.2.1.12 interfaces

6.2.1.12.1 Introduction

The `interfaces` command of the `show` subgroup enables displaying the GCap network interfaces:

- The management interfaces (`gcp0` and `gcp1`)
- The detection interfaces available physically `mon0` to `monx` or virtually `monvirt`

This command can take the keyword `delay` as a parameter to display the grace period granted to the interfaces. The following information is available with the `show interfaces` command:

- **Status:** the configured status of the interface among {**Enabled**|**Disabled**}
- **Physical Address:** the mac address of the interface
- **Speed:** the speed of the interface
- **Type:**
  - If it concerns a virtual interface: **Virtual**
  - If it is a physical interface: the type of cable/sfp connected to the physical port

6.2.1.12.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.1.12.3 Commands

```
show interfaces{ |delay|}
```

6.2.1.12.4 Example of displaying the available capture interfaces

- Enter the following command.

```
(gcap-cli) show interfaces
```

- Validate.
  The system displays the available capture interfaces.

```
 Waiting 10s for interfaces to be up

Name     State      Physical Address   Status   Speed    Type          Vendor ID   ␣
↪Device ID     PCI bus
gcp0     Enabled    00:50:56:01:29:01  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        0b:00.0
gcp1     Disabled   00:50:56:01:29:02  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        13:00.0
mon0     Enabled    00:50:56:01:29:03  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        1b:00.0
mon1     Disabled   00:50:56:01:29:04  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        04:00.0
mon2     Disabled   00:50:56:01:29:05  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        0c:00.0
mon3     Disabled   00:50:56:01:29:06  UP       1Gb      RJ45          0x8086      ␣
↪0x10d3        14:00.0
monvirt  Enabled    N/A                UP       N/A      Virtual       N/A         N/A   ␣
↪        N/A
```

> **Note:**
>
> All existing interfaces are displayed, even those making up an aggregation of interfaces.

If the interfaces are not recognised, the system displays irrelevant information as in the example below:

```
Waiting 10s for interfaces to be up

Name     State      Physical Address   Status   Speed    Type          Vendor ID  ␣
↪Device ID    PCI bus
eno12399 N/A        68:05:ca:dd:fe:fa  UP       1Gb      1000BASE-SX   0x8086     ␣
↪0x1572       31:00.0
eno12409 N/A        68:05:ca:dd:fe:fb  UP       1Gb      1000BASE-SX   0x8086     ␣
↪0x1572       31:00.1
eno12419 N/A        68:05:ca:dd:fe:fc  UP       1Gb      1000BASE-SX   0x8086     ␣
↪0x1572       31:00.2
eno12429 N/A        68:05:ca:dd:fe:fd  UP       1Gb      1000BASE-SX   0x8086     ␣
↪0x1572       31:00.3
eno8303  N/A        ec:2a:72:02:3a:1c  DOWN     N/A      RJ45          0x14e4     ␣
↪0x165f       04:00.0
eno8403  N/A        ec:2a:72:02:3a:1d  DOWN     N/A      RJ45          0x14e4     ␣
↪0x165f       04:00.1
monvirt  Disabled   N/A                UP       N/A      Virtual       N/A         N/A  ␣
↪        N/A
```

In this case, the system was unable to associate each of the network interfaces with its name.

Note:

Without the interfaces being assigned, access via the SSH connection on the *gcpx* port does not work. Connect to the GCap:

- Either by a direct connection (connect directly to the server)
- Or by a HTTP remote connection (iDRAC function for a Dell server)
- Or by a remote connection to the CLI in SSH via the iDRAC interface in serial port redirection mode

To correct this problem, two actions are possible:

- Manually restart an assignment using the *set advanced-configuration rescan-interfaces* command
- Manually assign the network interfaces using the *set advanced-configuration interface-names ...* command

6.2.1.12.5 Example of displaying the grace period given to start up the interfaces

- Enter the following command.

```
(gcap-cli) show interfaces delay
```

- Validate.
  The system displays the grace period for starting up the interfaces.

```
NIC startup delay: 10 seconds
```

6.2.1.13 keymap

6.2.1.13.1 Introduction

The `keymap` command of the `show` subgroup enables displaying the keyboard layout between azerty (choice fr) and qwerty (choice en) used on physical interfaces (KVM, iDRAC, physical).

6.2.1.13.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.13.3 Command

```
show keymap
```

6.2.1.13.4 Example of displaying the current keyboard language

```
(gcap-cli) show keymap
```

- Validate.
  The system displays the current information.

  ```
  Current keymap is fr
  ```

6.2.1.14 logs

6.2.1.14.1 Introduction

The `logs` command of the `show` subgroup enables displaying the various log files of the GCap:

Table1: Introduction

To display...	file name...
detection engine events	detection-engine-logs
kernel events	var-log-kernel
the aggregation of different logs	var-log-messages
GCap authentication information	var-log-auth
the launch information of scheduled tasks	var-log-cron
information about the activity of the various applications used	var-log-cron
information on the activity of the GCap users	var-log-user
debugging events	var-log-debug

A detailed explanation is given in the *Log files* section.

6.2.1.14.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.14.3 Command

```
show logs {detection-engine-logs|var-log-kernel|var-log-messages|var-log-auth|var-log-cron|var-log-daemo
```

6.2.1.14.4 Example of displaying the events of the detection engine

For this command, the detection engine must be started.

- Enter the following command.

```
(gcap-cli) show logs detection-engine-logs
```

- Validate.
  The system displays the detection engine events.
  A detailed explanation is given in the *Files of the logs-detection-engine-logs* section.

6.2.1.14.5 Example of displaying events related to the kernel

- Enter the following command.

```
(gcap-cli) show logs var-log-kernel
```

- Validate.
  The system displays the events related to the kernel.
  A detailed explanation is given in the *Files of the logs-var-log-kernel* section.

6.2.1.14.6 Example of displaying the aggregation of different logs

- Enter the following command.

```
(gcap-cli) show logs var-log-messages
```

- Validate.
  The system displays the connection information.
  A detailed explanation is given in the *Files of the logs-var-log-messages* section.

6.2.1.14.7 Example of displaying the GCap authentication information

- Enter the following command.

```
(gcap-cli) show logs var-log-auth
```

- Validate.
  The system displays the connection information.
  A detailed explanation is given in the *Files of the logs-var-log-auth* section.

6.2.1.14.8 Example of displaying the start information of scheduled tasks

- Enter the following command.

```
(gcap-cli) show logs var-log-cron
```

- Validate.
  The system displays the scheduled task start information.
  A detailed explanation is given in the *Files of the logs-var-log-cron* section.

6.2.1.14.9 Example of displaying information about the activity of different applications used

- Enter the following command.

```
(gcap-cli) show logs var-log-daemon
```

- Validate.
  The system displays information about the activity of the various applications used.
  A detailed explanation is given in the *Files of the logs-var-log-daemon* section.

6.2.1.14.10 Example of displaying GCap user activity information

- Enter the following command.

```
(gcap-cli) show logs var-log-user
```

- Validate.
  The system displays the information on the activity of the GCap users. A detailed explanation is given in the *Files of the logs-var-log-user* section.

6.2.1.14.11 Example of displaying the debug logs

- Enter the following command.

```
(gcap-cli) show logs var-log-debug
```

- Validate.
  The system displays the information on the activity of the GCap users.
  A detailed explanation is given in the *Files of the logs-var-log-debug* section.

6.2.1.15 monitoring-engine

6.2.1.15.1 Introduction

The `monitoring-engine` command of the `show` subgroup enables displaying the advanced options of the GCap detection engine configuration:

- The start-timeout grace period
- The grace period when the engine is stopped (stop-timeout)
- The status of the sanity checks

6.2.1.15.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.1.15.3 Command

```
show monitoring-engine {start-timeout|stop-timeout|sanity-checks}
```

6.2.1.15.4 Example of displaying the default value of the start-timeout

- Enter the following command.

```
(gcap-cli) show monitoring-engine start-timeout
```

- Validate. The system displays the current value.

```
Monitoring Engine Options:
Start timeout: 600s
```

6.2.1.15.5 Example of displaying the default value of the stop-timeout

- Enter the following command.

```
(gcap-cli) show monitoring-engine stop-timeout
```

- Validate.
  The system displays the current value.

```
Monitoring Engine Options:
Stop timeout: 300s
```

6.2.1.15.6 Example of displaying the status of the verification check

- Enter the following command.

```
(gcap-cli) show monitoring-engine sanity-checks
```

- Validate.
  The system displays the current value.

```
Monitoring Engine Options:
Sanity checks enabled
```

The system reports that the control system is active.
The detection engine will only start after it verifies that at least one `monx` capture interface is activated and a cable is connected.

6.2.1.16 network-config

6.2.1.16.1 Introduction

The GCap includes:

- capture and monitoring interfaces (`mon0` to `monx`)
- network interfaces (`gcp0`/`gcp1`) for managing the probe via SSH and for pairing with the GCenter
  Two cases are possible:
  - **Single-interface configuration**
    SSH connection for GCap management and VPN communication are managed through the `gcp0` interface.
  - **dual-interface configuration**.
    The VPN communication is controlled by the `gcp0` interface.
    The SSH connection for GCap management is handled by the `gcp1` interface.

For more information on network interfaces, refer to the *GCap input / output description* section.
The `network-config` command of the `show` subgroup enables displaying:

- The status of all GCap interfaces: `show network-config configuration` command
- The status for only the network interfaces: `show network-config status` command
  - The status for each interface: command `show network-config gcp0` or `show network-config gcp1`
  - The status for all network interfaces: `show network-config status` command
- The domain name: `show network-config domain` command
- The host name: `show network-config hostname` command
- The interface used to manage the probe in SSH: `show network-config ssh` command
- The speed of the VPN link between GCap and Gcenter: `show network-config vpn-link speed` command

---

6.2.1.16.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.1.16.3 Commands

```
show network-config {configuration|domain|gcp0|gcp1|hostname|ssh|status|vpn-link}
```

6.2.1.16.4 Example of displaying the GCap configuration

- Enter the following command.

```
(gcap-cli) show network-config configuration
```

- Validate.
  Depending on the single or dual interface configuration, the information is different.
  The two cases are listed below.

6.2.1.16.4.1 Single-interface configuration

In this case, the system displays the network configuration information including gcp1 (not used).

```
(gcap-cli) show network-config configuration
{
    "hostname": "GCap",
    "domain_name": "gatewatcher.com",
    "gcp0": {
        "description": "VPN / SSH",
        "ip_address": "X.X.X.X",
        "mask": "255.255.255.0",
        "default_gateway": "X.X.X.X",
        "enabled": true,
        "mtu": 1500
    },
    "gcp1": {
        "description": "SSH",
        "ip_address": "None",
        "mask": "255.255.255.0",
        "default_gateway": "",
        "enabled": false,
        "mtu": 1500
    },
    "mon0": {
        "description": "default",
        "enabled": true,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon1": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon2": {
        "description": "default",
```

(suite sur la page suivante)

(suite de la page précédente)

```
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon3": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    }
}
```

6.2.1.16.4.2 Dual-interface configuration

In this case, the system displays the configuration information.

```
(gcap-cli) show network-config configuration
{
    "hostname": "GCap",
    "domain_name": "gatewatcher.com",
    "gcp0": {
        "description": "VPN",
        "ip_address": "X.X.X.X",
        "mask": "255.255.255.0",
        "default_gateway": "X.X.X.X",
        "enabled": true,
        "mtu": 1500
    },
    "gcp1": {
        "description": "SSH",
        "ip_address": ""X.X.X.X"",
        "mask": "255.255.255.0",
        "default_gateway": "255.255.255.0",
        "enabled": true,
        "mtu": 1500
    },
    "mon0": {
        "description": "default",
        "enabled": true,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon1": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    },
    "mon2": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
```

(suite sur la page suivante)

(suite de la page précédente)

```
    },
    "mon3": {
        "description": "default",
        "enabled": false,
        "filtering_rules": {},
        "mtu": 1500
    }
}
```

6.2.1.16.5 Example of displaying the GCap domain

- Enter the following command.

```
(gcap-cli) show network-config domain
```

- Validate.
  The system displays the domain name.

```
Current domain name: gatewatcher.com
```

6.2.1.16.6 Example of displaying the gcp0 interface configuration

- Enter the following command.

```
(gcap-cli) show network-config gcp0
```

- Validate.
  The system displaying the gcp0 interface configuration.
  Depending on the single or dual interface configuration, the information is different.
  The two cases are listed below.

6.2.1.16.6.1 Single-interface configuration: gcp0 interface

SSH and VPN connections are handled by the gcp0 interface.
In this case, the system displays:

```
Interface ```gcp0``` configuration (VPN / SSH):
        - IP Address: X.X.X.X
        - Mask: 255.255.255.0
        - Gateway: X.X.X.X
```

6.2.1.16.6.2 Dual-interface configuration: `gcp0` interface

The VPN communication is controlled by the `gcp0` interface.
The SSH connection for GCap management is handled by the `gcp1` interface.
In this case, the system displays:

```
Interface gcp0 configuration (VPN):
        - IP Address: X.X.X.X
        - Mask: 255.255.255.0
        - Gateway: X.X.X.X
```

6.2.1.16.7 Example of displaying the `gcp1` interface configuration

- Enter the following command.

  ```
  (gcap-cli) show network-config gcp1
  ```

- Validate.
  The system displaying the `gcp1` interface configuration.
  Depending on the single or dual interface configuration, the information is different.
  The two cases are listed below.

6.2.1.16.7.1 Single-interface configuration: `gcp1` interface

In this case, the system displays the information of the `gcp1` not used:

```
Interface gcp1 configuration (SSH):
        - IP Address: None
        - Mask: 255.255.255.0
```

6.2.1.16.7.2 Dual-interface configuration: `gcp1` interface

In this case, the system displays the information of the `gcp1` used:

```
Interface gcp1 configuration (SSH):
        - IP Address: X.X.X.X
        - Mask: 255.255.255.0
        - Gateway: X.X.X.X
```

6.2.1.16.8 Example of displaying the host name of the GCap

- Enter the following command.

  ```
  (gcap-cli) show network-config hostname
  ```

- Validate.
  The system displays the interface the host name of the GCap.

```
Current hostname: GCap-name
```

6.2.1.16.9 Example of displaying the interface used to manage the probe in SSH

- Enter the following command.

```
(gcap-cli) show network-config ssh
```

- Validate.
  The system displays the SSH interface used to manage the GCap.
  In the case of the single-interface configuration, the system displays:

```
SSH is using interface gcp0
```

In the case of dual-interface configuration, the system displays:

```
SSH is using interface gcp1
```

6.2.1.16.10 Example of displaying the status of the GCap's gcp0 and gcp1 network interfaces

- Enter the following command.

```
(gcap-cli) show network-config status
```

- Validate.
  The system displays the status of the GCap network interfaces.

```
Name         Address        Carrier      Speed          Type
gcp0     xx:xx:xx:xx:xx:xx        UP     1000Mb/s          RJ45
gcp1     xx:xx:xx:xx:xx:xx        UP     1000Mb/s          RJ45
```

For each interface, the following information is displayed:

- `Address` : interface MAC address
- `Carrier` :
- valeur `UP`: the physical interface is connected
- valeur `DOWN`: the physical interface is not connected
- `Speed` : interface speed in Mb/s
- `Type` : the type of cable/sfp connected to the physical port

6.2.1.16.11 Example of displaying the speed of the VPN link between GCap and GCenter

- Enter the following command.

```
(gcap-cli) show network-config vpn-link speed
```

- Validate.
  The system displays the status of the GCap network interfaces.

```
Current VPN link speed: Fast
```

The system displays the current value: here `Fast`.

6.2.1.17 password-policy

6.2.1.17.1 Introduction

The `password-policy` command in subgroup `show` enables displaying the password policy for the accounts `setup`, `gviewadm` and `gview`.
The possibility of modifying this policy is enabled by the *`set password-policy`* command.

6.2.1.17.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.17.3 Command

```
show password-policy
```

6.2.1.17.4 Example of displaying the default password policy

- Enter the following command.

```
(gcap-cli) show password-policy
```

- Validate.
  The system displays the rules to be followed for defining a password.

```
Password complexity rules:
 Minimum different characters between old and new passwords: 2
 Minimum length: 12
 Lowercase character required: yes
 Uppercase character required: yes
 Digit required: yes
 Other character class required: yes
```

Parametre...	signification...
Minimum different characters between old and new passwords : x	At least x different characters are required for a password to be considered different
Minimum length	minimum password length: here 12 characters
Lowercase character required:	yes: means that the password must contain at least 1 lower case letter
Uppercase character required:	yes: means that the password must contain at least 1 capital letter
Digits required:	yes: means that the password must contain at least 1 digit 0 to 9
Symbols required:	yes: means that the password must contain at least 1 symbol, not a number or a letter

---

6.2.1.18 passwords

6.2.1.18.1 Introduction

The `passwords` command of the `show` subgroup enables:

- Displaying the list of users managed by the current level. This is accessible for setup, gviewadm, and gview users.
- Retrieving the root token as a text or QR code. This is available to setup users only.

> **Note:**
>
> The "retrieve root token" feature must be used in consultation with GATEWATCHER customer support.

6.2.1.18.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.18.3 Command

```
show passwords {list|text|qrcode}
```

6.2.1.18.4 Example of displaying the list of users managed by the current level

- Enter the following command.

```
(gcap-cli) show passwords list
```

- Validate.
  The system displays the list of users managed by the current level:
  Example for the gview level:

```
Allowed users: gview
```

  Example for the setup level:

```
Allowed users: gviewadm, gview, setup
```

6.2.1.18.5 Example of displaying the root token in text

- Enter the following command.

```
(gcap-cli) show passwords root text
```

- Validate.
  The system displays the root token in text.

```
Encrypted Root Token is:
↪"hzDpahGYq2i8aiSXwRfmhC7W3ZtSHteyJ22J2tL5OlI1Aq+nYsgJaGi7JyXVjGKyDs1TCBZqbXiobXe9y1o"
```

6.2.1.18.6 Example of displaying the root token as a QR code

- Enter the following command.

```
(gcap-cli) show passwords root qrcode
```

- Validate.
  The system displays the root token as a QR code.

6.2.1.19 protocols-selector

6.2.1.19.1 Introduction

This command has been removed since version 2.5.3.105.

6.2.1.20 session-timeout

6.2.1.20.1 Introduction

The `session-timeout` command of the `show` subgroup enables displaying the time of inactivity before a user session is disconnected.
This figure is expressed in minutes and the default value is 5 minutes.

6.2.1.20.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.1.20.3 Command

```
show session-timeout
```

6.2.1.20.4 Example of displaying the session-timeout value

- Enter the following command.

```
(gcap-cli) show session-timeout
```

- Validate.
  The system displays the current session-timeout value.

```
Current session timeout is 5 mins
```

6.2.1.21 setup-mode

6.2.1.21.1 Introduction

The `setup-mode` command of the `show` subgroup enables displaying:

- The current level,
- The interface of each accessible user profile:
  - The graphical interface or GUI mode
  - The command line interface or CLI mode

The default mode is CLI mode.
The GUI mode is deprecated.
The possibility of modifying these choices is provided by the *`set setup-mode`* command.

6.2.1.21.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.21.3 Command

```
show setup-mode
```

6.2.1.21.4 Example for displaying the user profile mode

- Enter the following command.

```
(gcap-cli) show setup-mode
```

- Validate.

```
Current default setup modes:
    - gview: cli mode
    - gviewadm: cli mode
    - setup: cli mode
```

This means that:
  – the current user is setup (highest level displayed)
  – upon the next access, each user will log on in CLI mode

6.2.1.22 status

6.2.1.22.1 Introduction

The `status` command of the `show` subgroup enables displaying the current GCap status.

6.2.1.22.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.1.22.3 Command

```
show status
```

6.2.1.22.4 Example of displaying the GCap information

- Enter the following command.

```
(gcap-cli) show status
```

- Validate.

```
status

GCap Name         : GCap
Version           : 2.5.3.105
Paired on GCenter : Not paired
Tunnel status     : Down
Detection Engine  : Container down

© Copyright GATEWATCHER 2021
```

The system displays the following information:

- **GCAP name**: name of the GCap (here GCap)
- **Version**: current software version: here 2.5.3.105
- **Tunnel status**: status of the tunnel between GCap and GCenter (here **Not paired**)
- **Detection Engine**: status of the detection engine container (here not started **Container down**)

6.2.1.23 tech-support

6.2.1.23.1 Introduction

The `tech-support` command of the `show` subgroup enables extracting the GCap information requested by technical support.

Note:

Tech-support is not encrypted and may contain confidential information.

6.2.1.23.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.1.23.3 Command

```
ssh -t setup@GCapX show tech-support {brief|large} > /tmp/tech-supp-brief-GCapX
```

Note:

GCapX should be replaced with the IP address of GCap..

6.2.1.23.3.1 Command for extracting light tech-support

```
ssh -t setup@GCapX show tech-support brief > /tmp/tech-supp-brief-GCapX
```

6.2.1.23.3.2 Command for extracting standard tech-support

```
ssh -t setup@GCapX show tech-support > /tmp/tech-supp-GCapX
```

6.2.1.23.3.3 Command for extracting heavy tech-support

```
ssh -t setup@GCapX show tech-support large > /tmp/tech-supp-large-GCapX
```

6.2.1.24 advanced-configuration

6.2.1.24.1 cpu-config

6.2.1.24.1.1 Introduction

The `cpu-config` command of the `show advanced-configuration` subgroup enables displaying the number of CPUs dedicated to the Sigflow detection engine.

6.2.1.24.1.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.1.24.1.3 Command

```
show advanced-configuration cpu-config
```

6.2.1.24.1.4 Example to display the CPUs assigned to Sigflow

- Enter the following command.

  ```
  (gcap-cli) show advanced-configuration cpu-config
  ```

- Validate.
  The system displays the number of CPUs dedicated to the Sigflow detection engine.

```
Current CPU profile is 1/2
```

In this example, there is 1 dedicated CPU out of 2 present.

6.2.1.24.2 high availability by redundancy of 2 GCaps

6.2.1.24.2.1 Introduction

The `status` command of the `show advanced-configuration high-availability` subgroup enables displaying the GCap status.
The `configuration` command of the `show advanced-configuration high-availability` subgroup enables displaying the high availability configuration of the GCap.
The `pubkey` command of the `show advanced-configuration high-availability` subgroup enables displaying the public key used by the high availability.

Operation:
Refer to the paragraph on *Operation of high availability*.

Type of network configuration:

- **link with 1 interface:** `mon0` is replaced by `ha0`, so capture interfaces can be used from `mon1`
- **link with 2 interfaces:** `mon0` and `mon1` are replaced by `ha0` and `ha1`, so the capture interfaces can be used from `mon2`

A GCap `leader` becomes a `follower` under the following conditions:

- Loss of connection to the GCenter for 1 min
- Loss of the detection engine for 5 min

6.2.1.24.2.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.1.24.2.3 Command

```
show advanced-configuration high-availability {status|configuration|pubkey}
```

6.2.1.24.2.4 Example for displaying the high availability status (GCap redundancy)

- Enter the following command.

```
(gcap-cli) show advanced-configuration high-availability status
```

- Validate.
  The system displays the result on the connected GCap.

```
Current high-availability status:
  - status: Operational [unhealthy]
  - paired GCap: fe80::233
  - leader: Leader
  - time since last status: Unknown
  - Leader since: 2022-01-21T15:35:09Z
```

The counters displayed are:

- **status:** status of the GCap:
- Operational: OK
- unhealthy: if the GCap is not connected to the neighbouring GCap.
- **paired GCap:** IPv6 address of the neighbouring GCap.
- leader: election status among:
- Leader
- Follower.
- **time since last status:** time since the last healthcheck of the neighbouring GCap.
- **Leader since:** date when the GCap became the Leader.

6.2.1.24.2.5 Example of displaying the public key used by the high availability

- Enter the following command.

```
(gcap-cli) show advanced-configuration high-availability pubkey
```

- Validate.
  The system displays the public key.

```
Wireguard public key: 'Fypsdign0R6aRP9j5pJkTcAJoi4eE/gTV9McCpBYjAk='
```

6.2.1.24.2.6 Example for displaying the configuration of the GCap high availability

- Enter the following command.

```
(gcap-cli) show advanced-configuration high-availability configuration
```

- Validate.
  The system displays the result.

```
Current high-availability configuration [enabled]:
  - bonding enabled: disabled
  - public ip: fe80::149/128
  - multicast group: ff02::200
  - peer public IP: fe80::233
  - peer public key: 2wtmY/oCaoUGreyr2CR0nKAIoEgTXkS0edXlXDvUfBU=
  - shared secret: Xxf4fknh4Ko0H2zgrI4Wyw==
```

- **bonding enabled:**
  - enabled: aggregation is activated
  - disabled: aggregation is desactivated.
- **public ip:** IPv6 address of the GCap among:
  - **Link-local:** If the GCaps are in the same subnet. Range FE80::/10. Ex: FE80::100/64.
  - **Unique Local Address (ULA):** If the GCaps are in different subnets. Range FD00::/7. Ex: FD00::100/64.
  - **Global Unicast:** If the GCap's need to communicate via the internet. Range 2001::/3. Ex: 2001::1/64.

  - **multicast group:** IPv6 multicast address for communicating between GCaps. Range FD00::/8. Ex: FF02::200.
  - **peer public IP:** IPv6 address of the neighbouring GCap among:
    - **Link-local:** If the GCaps are in the same subnet. Range FE80::/10. Ex: FE80::100/64.
    - **Unique Local Address (ULA):** If the GCaps are in different subnets. Range FD00::/7. Ex: FD00::100/64.
    - **Global Unicast:** If the GCap's need to communicate via the internet. Range 2001::/3. Ex: 2001::1/64.
  - **peer public key:** Public key of the neighbouring GCap via the `show advanced-configuration high-availability pubkey` command.
  - **shared secret:** Secret of 16 bytes encoded in base 64 that must be identical between the 2 GCaps.

6.2.1.24.3 interface-names

6.2.1.24.3.1 Introduction

The `interface-names` command of the `advanced-configuration` subgroup enables displaying the name of the interfaces.

6.2.1.24.3.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.1.24.3.3 Command

```
show advanced-configuration interface-names
```

6.2.1.24.3.4 Example for displaying the names of the interfaces

- Enter the following command.

```
(gcap-cli) show advanced-configuration interface-names
```

- Validate.
  The system displays the names of the interfaces.

6.2.1.24.4 load-balancing

6.2.1.24.4.1 Introduction

The `load-balancing` command of the `show advanced-configuration` subgroup enables displaying the load balancing configuration from the `monx` listed capture interface to the GCap CPUs.

Note:

The feature is compatible with some GCap models (see model datasheet).

6.2.1.24.4.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.1.24.4.3 Command

```
show advanced-configuration load-balancing
```

6.2.1.24.4.4 Example of displaying the load balancing configuration

- Enter the following command.

```
(gcap-cli) show advanced-configuration load-balancing
```

- Validate.

```
Showing current load balancing configuration

Interface mon2:
  - Load balancing method: XDP
  - Load balancing algorithm: 5-tuple
  - Seed: 1
```

6.2.1.24.5 local-rules

6.2.1.24.5.1 Introduction

The `local-rules` command of the `show advanced-configuration` subgroup enables displaying:

- Under the heading `Rules`: the local Sigflow rules, i.e.:
  - Detection rules and
  - File rebuilding rules
- Under the heading `threshold`:
  - Thresholds or limits defined by the keyword "threshold"
  - Deletion rules defined by the keyword "suppress"

It is only possible to display the rules of the configured tenant.
If the probe is configured in single-tenant mode then only the local_all.rules file can be displayed.
For more information, please refer to the paragraph *Capture and monitoring interfaces: single-tenant vs multi-tenant*.

6.2.1.24.5.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.1.24.5.3 Command

```
show advanced-configuration local-rules {TENANT|list}
```

The TENANT parameter can take the following values:

- Single-tenant: all
- Multi-tenant by int: {mon0|mon1|mon2|mon3|monvirt}
- Multi-tenant by vlan:
  - default
  - VLAN X
  - VLAN X Y

6.2.1.24.5.4 Example to list searchable rule files

- Enter the following command.

```
(gcap-cli) show advanced-configuration local-rules list
```

- Validate.
  The system displays the result.

```
 Available rule files:
   - mon0
  - monvirt
```

6.2.1.24.5.5 Example to list the searchable rule files display the rules in single tenant mode

- Enter the following command.

```
(gcap-cli) show advanced-configuration local-rules all
```

- Validate.
  The system displays the result.

```
Rules:
alert dns any any -> any any (msg:"[ TEST AUTO ] ALERT DNS UDP";sid:12345600;priority:2;)

Thresholds
```

The result is displayed in two categories:
  - **Rules:** in this category, the locally defined rules are listed
  - **Thresholds:** in this category, the locally defined thresholds and limits are listed

---

6.2.1.24.5.6 Example of displaying the rules in multi-tenant mode for the `mon0` interface

- Enter the following command.

```
(gcap-cli) show advanced-configuration local-rules mon0
```

- Validate.
  The system displays the result.

```
Displaying rules for mon0

Rules:
 alert dns any any -> any any (msg:"[ TEST AUTO ] ALERT DNS UDP";sid:12345600;priority:2;
↪)

Thresholds
```

6.2.1.24.5.7 Example of displaying the multi-tenant rules for vlan 10

- Enter the following command.

```
(gcap-cli) show advanced-configuration local-rules VLAN 10
```

- Validate.
  The system displays the result.

```
Displaying rules for vlan 10

Rules:
alert dns any any -> any any (msg:"[ TEST AUTO ] ALERT DNS UDP";sid:12345600;priority:2;)

Thresholds
```

6.2.1.24.6 MTU

6.2.1.24.6.1 Introduction

The `mtu` command in subgroup `show advanced-configuration` enables displaying the MTU value in bytes of all enabled network interfaces.

6.2.1.24.6.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.1.24.6.3 Command

```
show advanced-configuration mtu
```

6.2.1.24.6.4 Example of displaying the MTU value

- Enter the following command.

```
(gcap-cli) show advanced-configuration mtu
```

- Validate.
  The system displays the result.

```
 Current Network MTU configuration:
    - mon1: 1500
    - mon2: 1500
    - mon3: 1500
    - cluster0: 1500
    - gcp0: 1500
```

The values are displayed for all enabled network interfaces.

6.2.1.24.7 packet-filtering

6.2.1.24.7.1 Introduction

The `packet-filtering` command of the `show advanced-configuration` subgroup enables displaying the static packet filtering rules.

Note:

Packet filtering is not supported when the MTU > 3000.

6.2.1.24.7.2 Prerequisites

- **User:** setup
- **Dependencies:**
  - The detection engine must be switched off
  - A network capture interface must be enabled

6.2.1.24.7.3 Command

```
show advanced-configuration packet-filtering
```

6.2.1.24.7.4 Example of displaying the flow filtering rules

- Enter the following command.

```
(gcap-cli) show advanced-configuration packet-filtering
```

- Validate.
  The system displays the result.

```
Current XDP filters:
 - 0: iface mon1 native vlan 10
 - 1: iface mon2 native vlan 1
 - 2: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 22:22
 - 3: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 443:443
 - 4: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 465:465
 - 5: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 993:993
 - 6: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 995:995
 - 7: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 500:500
 - 8: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 4500:4500
 - 9: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto GRE
 - 10: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto ESP
 - 11: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto AH
 - 12: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto L2TP
```

6.2.2 set

6.2.2.1 bruteforce-protection

6.2.2.1.1 Introduction

The `bruteforce-protection` command of the `set` subgroup enables the system to protect against brute force attacks when a user logs in.
User accounts are automatically locked for a set period of time after several unsuccessful attempts.
The default value is 3.
To view the current values for the number of attempts and the account lockout duration, use the *show bruteforce-protection* command.

6.2.2.1.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.2.1.3 Commands

```
set bruteforce-protection{lock-duration|max-tries|restore-default}
```

6.2.2.1.3.1 Command used to set a maximum number of authentication attempts for an account (0 to deactivate)

```
set bruteforce-protection lock-duration {0|1-86400}
```

6.2.2.1.3.2 Command used to set an account lockout duration in seconds (0 to deactivate)

```
set bruteforce-protection max-tries {0|1-100}
```

6.2.2.1.3.3 Command to restore the default configuration

```
set bruteforce-protection restore-default
```

6.2.2.1.4 Example to change the lockout duration to 360 seconds

- Enter the following command.

```
(gcap-cli) set bruteforce-protection lock-duration 360
```

- Validate.
  The system indicates the setting has been changed.

```
Updating bruteforce protection configuration
Bruteforce protection configuration updated
```

6.2.2.2 Clusters

6.2.2.2.1 Introduction

The `clusters` command of the `set` subgroup enables the aggregation to be configured on the GCap capture interfaces.
For more information on aggregation, see the paragraph *Capture and Monitoring Interfaces `monx` between TAP and GCap: Aggregation Capability*.
This command enables defining which interfaces are connected to the same TAP in order to ensure a correct flow interpretation.

> **Note:**
>
> This functionality is necessary if the qualified Test Access Port (TAP) present in the architecture does not provide the interface aggregation functionality.
> The cluster automatically inherits the MTU of the interface with the highest MTU in the grouping.

Aggregation has an *Impact on the other functionalities*.

6.2.2.2.2 Prerequisites

- **User:** setup
- **Dependencies:** activate at least two capture interfaces

6.2.2.2.3 Commands

6.2.2.2.3.1 Command to add an interface aggregation

```
(gcap-cli) set clusters add interfaces {mon0|mon1|mon2|mon3} {mon0|mon1|mon2|mon3} description
DESCRIPTION
```

6.2.2.2.3.2 Command to activate or deactivate an interface aggregation

```
(gcap-cli) set clusters {enable|disable} NAME
```

The `NAME` field can be viewed by using the `show clusters` command.

6.2.2.2.3.3 Command to delete an interface aggregation

```
(gcap-cli) set clusters delete NAME
```

6.2.2.2.4 Example to create an aggregation of the interfaces mon0 and mon1 with the description test.

- Enter the following command.

```
(gcap-cli) set clusters add interfaces mon0 mon1 description `test`
```

- Validate.
  The system displays the result.

```
Creating cluster test with interfaces mon0, mon1
Successfully created cluster `test`
```

6.2.2.3 compatibility-mode

6.2.2.3.1 Introduction

The `compatibility-mode` command of the `set` subgroup enables modifying the compatibility mode used to interact with GCenter.
The compatibility mode will affect the available functionality of GCap.

Several compatibility modes are available:

- 2.5.3.100: GCenter 2.5.3.100 and below
- 2.5.3.101: GCenter 2.5.3.101
- 2.5.3.102+: GCenter 2.5.3.102 and above

For a GCap	GCenter version	supported	Action or Order to be executed
2.5.3.105	2.5.3.100 HF7	unsupported	GCenter to migrate to a newer version
2.5.3.105	2.5.3.101 HF3	supported	set compatibility-mode 2.5.3.101
2.5.3.105	2.5.3.102	supported	set compatibility-mode 2.5.3.102+
2.5.3.106	2.5.3.100 HF7	unsupported	GCenter to migrate to a newer version
2.5.3.106	2.5.3.101 HF3	supported	set compatibility-mode 2.5.3.101
2.5.3.106	2.5.3.102	supported	set compatibility-mode 2.5.3.102+
2.5.3.107	2.5.3.100 HF7	unsupported	GCenter to migrate to a newer version
2.5.3.107	2.5.3.101 HF3	supported	set compatibility-mode 2.5.3.101
2.5.3.107	2.5.3.102	supported	set compatibility-mode 2.5.3.102+

Important:

The above table is given as an example. Please refer to the GCap Release Note.

Note:

The compatibility mode for GCenter version 2.5.3.100 and below is deprecated.

6.2.2.3.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.2.3.3 Command

```
set compatibility-mode {2.5.3.100|2.5.3.101|2.5.3.102+}
```

6.2.2.3.4 Example of configuring compatibility between a GCap version V106 (or V107) with a GCenter 2.5.3.102

- Enter the following command.

  ```
  (gcap-cli) set compatibility-mode 2.5.3.102+
  ```

- Validate.

6.2.2.4 datetime

6.2.2.4.1 Introduction

The `datetime` command of the `set` subgroup enables the date and time of the GCap to be adjusted.
This enables avoiding clock problems that could lead to the impossibility of establishing an IPSec tunnel with GCenter, for example.

> **Note:**
>
> This clock must always be adjusted so that the GCap and the associated GCenter are on the same time (e.g. for the time-stamping of events).

6.2.2.4.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.2.4.3 Command

```
set datetime {YYYY-MM-DDThh:mm:ssZ}
```

6.2.2.4.4 Examples for changing the GCap time

- Enter the following command.

```
(gcap-cli) set datetime 2022-01-26T16:00:00Z
```

- Validate.
  The system displays the result.

```
Date successfully changed to Wed Jan 26 2022 16:00:00
```

6.2.2.5 gcenter-ip

6.2.2.5.1 Introduction

The `gcenter-ip` command of the `set` subgroup enables specifying the IP address of the GCenter to which the GCap will be paired.

Note:

The GCap uses this IP address during pairing to connect to the GCenter via SSH and retrieve the GCenter fingerprint.

6.2.2.5.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.2.5.3 Command

```
set gcenter-ip {GCenter-IP}
```

6.2.2.5.4 Example

- Enter the following command.

```
(gcap-cli) set gcenter-ip 192.168.1.1
```

- Validate.
  The system displays the result.

```
Setting GCenter IP to 192.168.1.1
```

6.2.2.6 interfaces

6.2.2.6.1 Introduction

The `interfaces` command of the `set` subgroup enables the administration of capture interfaces.
Interfaces can be physical or virtual.
Virtual interfaces enable replaying *.pcap* files directly on the GCap.

Important:

Interfaces composing an aggregation of interfaces ("cluster") can neither be enabled nor disabled.

6.2.2.6.2 Prerequisites

- **User:** setup
- **Dependencies:**
  - The detection engine must be switched off
  - To enable an interface, it must be in the disabled state
  - To disable an interface, it must be in the enabled state

6.2.2.6.3 Command

To change the delay before starting up the interfaces: `set interfaces delay SECOND`.
To enable or disable interfaces: `set interfaces {enable|disable} {mon0|mon1|mon2|mon3|monvirt}`

6.2.2.6.4 Example to change the interface start-up delay by five seconds

- Enter the following command.

```
(gcap-cli) set interfaces delay 5
```

- Validate.

6.2.2.6.5 Example of enabling the `mon0` capture interface

- Enter the following command.

```
(gcap-cli) set interfaces enable mon0
```

- Validate.

> **Note:**
>
> If the system displays the following message, *Command set interfaces enable monx is not recognised*, check whether the monx interface is an aggregation using the *show clusters* command.

6.2.2.6.6 Example of disabling the `mon1` capture interface

- Enter the following command.

```
(gcap-cli) set interfaces disable mon1
```

- Validate.

> **Note:**
>
> If the system displays the following message, *Command set interfaces disable monx is not recognised*, check whether the monx interface is an aggregation using the *show clusters* command.

6.2.2.7 keymap

6.2.2.7.1 Introduction

The `keymap` command of the `set` subgroup enables choosing the keyboard layout between azerty (choice fr) and qwerty (choice en) used on physical interfaces (KVM, iDRAC, physical).

6.2.2.7.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.2.7.3 Command

```
set keymap {fr|en}
```

6.2.2.7.4 Example of a French keyboard

- Enter the following command.

```
(gcap-cli) set keymap fr
```

- Validate.
  The system displays the result.

```
Setting keymap to fr
```

6.2.2.7.5 Example of an English keyboard

- Enter the following command.

```
(gcap-cli) set keymap en
```

- Validate.
  The system displays the result.

```
Setting keymap to en
```

6.2.2.8 monitoring-engine

6.2.2.8.1 Introduction

The `monitoring-engine` command of the `set` subgroup enables applying an advanced configuration for the GCap sensor detection engine.

> **Note:**
>
> If the number of signatures loaded by Sigflow is too large, the timeout value must be adjusted.

6.2.2.8.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine is switched off

6.2.2.8.3 Command

To change the grace period when starting the engine: `set monitoring-engine start-timeout SECOND`.
To change the grace period when the engine is stopped: `set monitoring-engine stop-timeout SECOND`.
To enable or disable the check of the controls: `set monitoring-engine {disable-sanity-checks|enable-sanity-checks}`.
If the `sanity-checks` option is set to `enable`, the detection engine starts only after verifying that at least one `monx` capture interface has been activated and that a cable is connected.

6.2.2.8.4 Example of changing the grace period to 600 seconds when starting the engine

- To change the grace period to 600 seconds when starting the engine:
  - Enter the following command.

    ```
    (gcap-cli) set monitoring-engine start-timeout 600
    ```

  - Validate.
- To check the value modification:
  - Enter the following command.

    ```
    (gcap-cli) show monitoring-engine start-timeout
    ```

  - Validate.
    The system displays the current value.

    ```
    Monitoring Engine Options:
    start timeout: 600s
    ```

6.2.2.8.5 Example of changing the grace period on engine shutdown to 600 seconds

- To change the grace period to 600 seconds when the engine is stopped:
  - Enter the following command.

    ```
    (gcap-cli) set monitoring-engine stop-timeout 600
    ```

  - Validate.
- To check the value modification:
  - Enter the following command.

    ```
    (gcap-cli) show monitoring-engine stop-timeout
    ```

- Validate.
  The system displays the current value.

  ```
  Monitoring Engine Options:
  Stop timeout: 600s
  ```

6.2.2.8.6 Example of disabling the capture interface verification

- To disable the capture interface verification:
  - Enter the following command.

    ```
    (gcap-cli) set monitoring-engine disable-sanity-checks
    ```

  - Validate.
- To check the value modification:
  - Enter the following command.

    ```
    (gcap-cli) show monitoring-engine sanity-checks
    ```

  - Validate.
    The system displays the current value.

    ```
    Monitoring Engine Options:
    Sanity checks disabled
    ```

6.2.2.8.7 Example of enabling the capture interface verification

- To disable the capture interface verification:
  - Enter the following command.

    ```
    (gcap-cli) set monitoring-engine enable-sanity-checks
    ```

  - Validate.
- To check the value modification:
  - Enter the following command.

    ```
    (gcap-cli) show monitoring-engine sanity-checks
    ```

  - Validate.
    The system displays the current value.

    ```
    Monitoring Engine Options:
    Sanity checks enabled
    ```

6.2.2.9 network-config

6.2.2.9.1 Introduction

For more information on the network interfaces (gcp0/gcp1) and the capture and monitoring interfaces (mon0 to monx), refer to the *show network-config* command.
The network-config command of the set subgroup enables modifying the network configuration of the GCap.
The network-config command of the set subgroup enables configuring:

- Each interface with the network parameters: `set network-config {gcp0|gcp1} [ip-address IP_value] [gateway GATEWAY_value] [mask MASK_value]` command
- The domain name: `set network-config domain NAME_value` command
- The host name: `set network-config hostname HOSTNAME_value` command
- The interface used to manage the probe in SSH: `set network-config ssh {gcp0|gcp1}` command
- The speed of the VPN link between GCap and GCenter: `set network-config vpn-link speed {slow|fast}` command
  - The slow parameter defines a link of less than 100Mbit/s.
  - The fast parameter defines a link higher than 100Mbit/s.

6.2.2.9.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.2.9.3 Command

```
set network-config {gcp0|gcp1} [ip-address IP_value] [gateway GATEWAY_value] [mask MASK_value]
[confirm] [no-reload]
```

```
set network-config ssh {gcp0|gcp1} [confirm] [no-reload]
```

```
set network-config [domain-name NAME_value|hostname HOSTNAME_value] [confirm]
```

```
set network-config vpn-link speed {slow|fast}
```

Note:

The *no-reload* option enables not reloading network services.

6.2.2.9.4 Example of configuring the `gcp0` interface for pairing and the `gcp1` interface for management

- Enter the following command.

```
(gcap-cli) set network-config ssh gcp1
```

- Validate.
- Enter the following command.

```
(gcap-cli) set network-config gcp0 ip-address X.X.X.X gateway Z.Z.Z.Z mask Z.Z.Z.Z
```

- Validate.
- Enter the following command.

```
(gcap-cli) set network-config gcp1 ip-address Y.Y.Y.Y gateway Z.Z.Z.Z mask Z.Z.Z.Z confirm
```

- Validate.

6.2.2.9.5 Example of configuring the `gcp0` interface for pairing and for management

Note:

L'interface `gcp1` n'est pas utilisée.

- Enter the following command.

```
(gcap-cli) set network-config ssh gcp0
```

- Validate.
- Enter the following command.

```
(gcap-cli) set network-config gcp0 ip-address X.X.X.X gateway X.X.X.X mask X.X.X.X confirm
```

- Validate.

6.2.2.9.6 Example of displaying the Gcap in gatewatcher.com

- To change the GCap domain in gatewatcher.com:
  - Enter the following command.

```
(gcap-cli) set network-config domain-name gatewatcher.com
```

  - Validate.

```
Setting hostname/domain name to:
  - Hostname: gcap-int-129-dag
  - Domain name: gatewatcher.com
Do you want to apply this new configuration? (y/N)
```

  - Press **y** and then confirm.

```
Applying configuration...

00% Generating interfaces configuration     [OK]
09% Generating network configuration        [OK]
18% Generating sshd configuration           [OK]
27% Reconfiguring network                   [OK]
36% Reconfiguring firewall                  [OK]
45% Notifying new network addresses         [OK]
54% Restarting sshd service                 [OK]
63% Restarting rsyslog service              [OK]
72% Restarting gcenter-xfer-daemon service  [OK]
81% Restarting netdata service              [OK]
90% Restarting rsyslog service              [OK]
Procedure completed with success
```

- To check the value modification:
  - Enter the following command

```
(gcap-cli) show network-config domain
```

  - Validate.
    The system displays the domain name.

```
Current domain name: gatewatcher.com
```

6.2.2.9.7 Example of modifying the speed of the VPN link between GCap and GCenter (for example from fast to slow)

- To view the current speed:
  - Enter the following command.

```
(gcap-cli) show network-config vpn-link speed
```

  - Validate.
    The system displays the status of the GCap network interfaces.

```
Current VPN link speed: Fast
```

    The system displays the current value: here `Fast`.
- To change the current speed (from fast to slow):
  - Enter the following command.

```
(gcap-cli) set network-config vpn-link speed slow
```

- Validate.
  The system displays the result of the configuration change.

```
New VPN link qualifiers configured successfully
```

6.2.2.10 password-policy

6.2.2.10.1 Introduction

The `password-policy` command in subgroup `set` enables defining a password policy for the `setup`, `gviewadm` and `gview` accounts.
This policy applies to all users.

6.2.2.10.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.2.10.3 Command

To set the password complexity options: `(gcap-cli) set password-policy {lowercase-optional|lowercase-required|uppercase-optional|uppercase-required|digits-optional|digits-requ`

To enable or disable the password control policy: `(gcap-cli) set password-policy {disable|enable}`

To restore the default password control policy: `(gcap-cli) set password-policy restore-default`

To specify the minimum password length: `(gcap-cli) set password-policy password-length {8-100}`

To set the length of time a password is valid: `(gcap-cli) set password-policy validity-duration {0|1-3650}`

To disallow previously used passwords: `(gcap-cli) set password-policy previous-check {0|1-1000}`

6.2.2.10.4 Example of removing the restriction on numbers

- Enter the following command.

```
(gcap-cli) set password-policy digits-optional
```

- Validate.
  The system displays the result.

```
Rules successfully updated
```

Note:

To avoid having an end of validity, put 0 in the `Validity duration` field.
To prevent verification of old passwords, put 0 in the `Verify last 0 passwords` field.

6.2.2.10.5 Example of disabling the default password control policy

- To disable the default password control policy:
  - Enter the following command.

```
(gcap-cli) set password-policy disable
```

  - Validate.
    The system displays the result.

```
Rules successfully updated
```

- To check the value modification:
  - Enter the following command.

```
(gcap-cli) show password-policy
```

  - Validate.
    The system displays the disabled status of the control.

```
No active password policy
```

6.2.2.11 passwords

6.2.2.11.1 Introduction

The `passwords` command of the `set` subgroup enables modifying the passwords of the setup, gviewadm, and gview users.

User	can change the password		
	setup	gviewadm	gview
setup	X	X	X
gviewadm		X	X
gview			X

Passwords must match predefined rules.

For more information on these rules, use the *show password-policy* command.

Important:

Check the keyboard configuration before changing the password (*show keymap* command).

6.2.2.11.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.2.11.3 Command

```
set passwords {setup|gviewadm|gview}
```

6.2.2.11.4 Example of changing the password of the current user (here setup)

- Enter the following command.

```
(gcap-cli) set passwords setup
```

- Validate.

```
(current) LDAP Password:
```

- Enter the LDAP password and confirm.
  The system asks for the new password of the account (here setup).

```
New password:
```

- Enter the new password and confirm.
  The system asks you to re-enter the new password.

```
Retype new password:
```

- Enter the new password again and confirm.
  The system announces that the password has been changed.

```
passwd: password updated successfully
Password changed for user setup
```

6.2.2.11.5 Example of changing the password of another user

- Enter the following command.

```
(gcap-cli) set passwords gviewadm
```

- Validate.

```
Password complexity rules:
   Minimum different characters between old and new passwords: 2
   Minimum length: 12
   Lowercase character required: yes
   Uppercase character required: yes
   Digit required: yes
   Other character class required: yes
New password:
```

- Enter the new password for the account (here gviewadm) then validate.
  The system asks you to re-enter the new password.

```
Retype new password:
```

- Enter the new password again and confirm.
  The system announces that the password has been changed.

```
passwd: password updated successfully
Password changed for user gviewadm
```

6.2.2.12 protocols-selector

6.2.2.12.1 Introduction

This command has been removed since version 2.5.3.105.

6.2.2.13 session-timeout

6.2.2.13.1 Introduction

The `session-timeout` command of the `set` subgroup enables configuring the time of inactivity before logging out of a user session.

Below are the configuration options:

- The default value is `5min`
- The value `0` enables deactivating the automatic disconnection
- The maximum value is `1440min`

Modifying this configuration is possible at any time. It has no impact on the overall operation of the GCap.

6.2.2.13.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.2.13.3 Command

```
set session-timeout MINUTES
```

6.2.2.13.4 Example of changing the default value for automatic logoff via the user setup

- To change the default value for automatic logoff via user setup:
  - Enter the following command.

    ```
    (gcap-cli) set session-timeout 1200
    ```

  - Validate.
    The system displays the result.

    ```
    Setting session timeout to 1200 mins
    Session timeout successfully changed.
    ```

- To check the value modification:
  - Enter the following command.

    ```
    (gcap-cli) show session-timeout
    ```

  - Validate.
    The system displays the current session-timeout value.

    ```
    Current session timeout is 1200 mins
    ```

6.2.2.14 setup-mode

6.2.2.14.1 Introduction

The `setup-mode` command of the `set` subgroup enables choosing, for each user profile:

- Either the graphical interface (GUI mode)
- Or the command line interface (CLI mode)

The CLI mode is activated by default on all user profiles (setup, gview, and gviewadm).
Modifying this configuration is possible at any time. It has no impact on the overall operation of the GCap.

> **Note:**
>
> The GUI mode is deprecated and will be removed in a future release.

The possibility of displaying the current configuration is provided by the *`show setup-mode`* command.

6.2.2.14.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.2.14.3 Command

```
set setup-mode {setup|gview|gviewadm} {gui|cli}
```

6.2.2.14.4 Example of changing the default mode from setup user profile to CLI mode

- Enter the following command.

```
(gcap-cli) set setup-mode setup cli
```

- Validate.
  The system displays change the default mode to CLI mode.

```
Setting setup to mode cli
Default setup mode successfully updated. Changes will be effective on next login
```

As indicated, the change will be made upon the next login.

Note:

If the following message is displayed:

```
User setup is already set to mode cli
Default setup mode successfully updated.
Changes will be effective on next login
```

this means that the current mode is in CLI mode but, for safety, it is reapplied.

6.2.2.15 ssh-keys

6.2.2.15.1 Introduction

The `ssh-keys` command of the `set` subgroup enables adding or changing the SSH keys.
Depending on the account, it is possible to change only the current level and the lower level. The addition or modification can be carried out either on the command line or via the Nano text editor.
Changing SSH keys overwrites the old keys. You must specify the old keys followed by the new ones in the command.

User	can change the password		
	setup	gviewadm	gview
setup	X	X	X
gviewadm		X	X
gview			X

The GCap enables up to 50 different users with different key sizes:

- RSA 2048 or 4096
- ssh-ed25519
- ecdsa-sha2-nistp256.

6.2.2.15.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.2.15.3 Command

```
set ssh-keys {setup|gviewadm|gview} "ssh-rsa ...\nssh-rsa
```

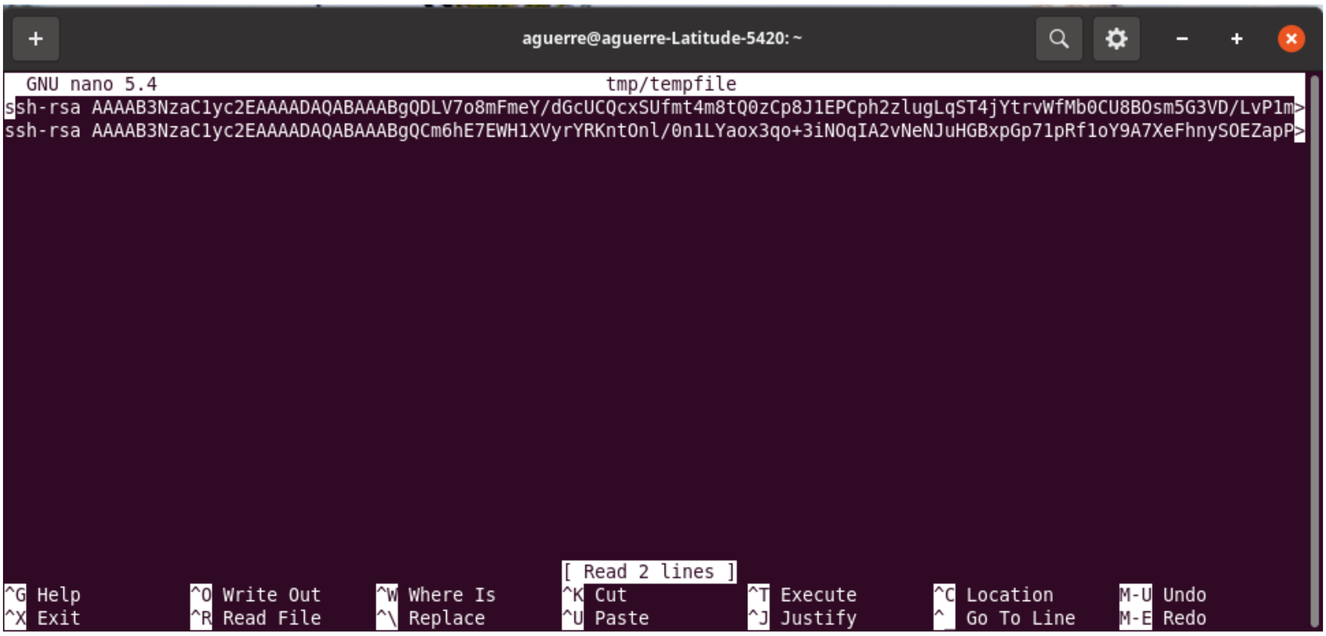
6.2.2.15.4 Example of using the text editor

- Enter the following command.

```
(gcap-cli) set ssh-keys gview
```

- Validate.

The text editor displays the SSH password file.



```
  GNU nano 5.4                                     tmp/tempfile
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDLV7o8mFmeY/dGcUCQcxSUfmt4m8tQ0zCp8J1EPCph2zlugLqST4jYtrvWfMb0CU8BOsm5G3VD/LvP1m>
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCm6hE7EWH1XVyrYRKntOnl/0n1LYaox3qo+3iNOqIA2vNeNJuHGBxpGp71pRf1oY9A7XeFhnySOEZapP>









                                              [ Read 2 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^_ Go To Line    M-E Redo
```

Each line in the file is an SSH key starting with ssh-rsa.

- To delete a key, delete the line.
  To change a key, edit a line.
  To add a key, add a line starting with ssh-rsa.
- To exit, press **CTRL** + **X**.
- Save the changes if necessary.

6.2.2.15.5 Example of adding an SSH key to the setup user from a connection with the setup user

- Enter the following command.

```
(gcap-cli) set ssh-keys setup "ssh-rsa ..."
```

- Validate.

6.2.2.16 advanced-configuration

6.2.2.16.1 cpu-config

6.2.2.16.1.1 Introduction

The `cpu-config` command of the `advanced-configuration` subgroup enables modifying the number of CPUs dedicated to the Sigflow detection engine.
This CPU allocation can be achieved by selecting:

- The distribution of the CPUs for Sigflow in relation to the total number of CPUs present: parameter (`1/2`, `2/3`, `3/4`, `4/5`)
- The numbers of CPUs reserved for Sigflow

Important:

Do not exceed 80% of the CPUs for Sigflow.

6.2.2.16.1.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.1.3 Command

```
set advanced-configuration cpu-config {1/2|2/3|3/4|4/5|custom} [CPU_LIST]
```

6.2.2.16.1.4 Example to allocate half of the CPUs to Sigflow

- Enter the following command.

```
(gcap-cli) set advanced-configuration cpu-config 1
```

- Validate.
  The system displays the requested action.

```
Updating CPU profile from 3/4 to 1/2
```

6.2.2.16.1.5 Example to allocate a list of CPUs to Sigflow

- Enter the following command.

```
(gcap-cli) set advanced-configuration custom 1,2,3,4,5
```

- Validate.

```
cpu-config custom 1,2,3,4,5
Building custom profile 1,2,3,4,5
```

- Restart the GCap, otherwise the behaviour may prove to be random.

6.2.2.16.2 high-availability

6.2.2.16.2.1 Introduction

The `high-availability` command of the `advanced-configuration` subgroup enables configuring high availability between 2 GCap (function added from version 2.5.3.105).

Operation:

Refer to the paragraph on *Operation of high availability*.

Type of network configuration:

- **link with 1 interface:** `mon0` is replaced by `ha0`
  The available capture interfaces are therefore `mon1`, `mon2`, etc.
- **link with 2 interfaces:** `mon0` and `mon1` are replaced by `ha0` and `ha1`.
  The available capture interfaces are therefore `mon2`, `mon3`, etc.

A GCap `leader` becomes a `follower` under the following conditions:

- Loss of connection to the GCenter for 1 min
- Loss of the detection engine for five minutes

6.2.2.16.2.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.2.16.2.3 Command

```
set advanced-config high-availability [public-ip IPV6/MASK] [gateway GATEWAY|null]
[peer-ip IPV6] [multicast-group IPV6] [shared-secret SECRET] [peer-pubkey KEY]
[bonding-enabled|bonding-disabled]
```

```
set advanced-config high-availability [enable|disable] [confirm]
```

Explanation of the parameters:

- **bonding-enabled:** enable card aggregation `mon0` + `mon1`.
- **bonding-disabled:** disable card aggregation `mon0` + `mon1`.
- **enable:** enable high availability.
- **disable:** disable high availability.

- **gateway:** IPv6 address of the gateway in case the GCAPs are not in the same subnet.
- **Multicast-group:** IPv6 multicast address for communication between GCaps. Range FD00::/8. Ex: FF02::200.
- **peer-ip:** IPv6 address of the neighbouring GCap among:
  - **Link-local:** if the GCaps are in the same subnet. Range FE80::/10. Ex: FE80::100/64.
  - **Unique Local Address (ULA):** If the GCaps are in different subnets. Range FD00::/7. Ex: FD00::100/64.
  - **Global Unicast:** If the GCap's need to communicate via the internet. Range 2001::/3. Ex: 2001::1/64.
- **peer-pubkey:** Public key of the neighbouring GCap via the `show advanced-configuration high-availability pubkey` command.
- **peer-ip:** IPv6 address of the GCap among:
  - **Link-local:** if the GCaps are in the same subnet. Range FE80::/10. Ex: FE80::100/64.
  - **Unique Local Address (ULA):** If the GCaps are in different subnets. Range FD00::/7. Ex: FD00::100/64.
  - **Global Unicast:** If the GCap's need to communicate via the internet. Range 2001::/3. Ex: 2001::1/64.
- **shared-secret:** secret of 16 bytes encoded in base 64 that must be identical between the 2 GCaps.

6.2.2.16.2.4 Example of configuring high availability on the first GCap

- Enter the following command.

```
(gcap-cli) set advanced-configuration high-availability peer-ip fe80::XXX public-ip␣
↪fe80::YYY multicast-group ff02::200 peer-pubkey 2wtmY/
↪oCaoUGreyr2CROnKAIoEgTXkSOedXlXDvUfBU= shared-secret Xxf4fknh4KoOH2zgrI4Wyw==
```

- Validate.
  The system displays the result.

```
Updating HA configuration
High availability configuration successfully updated
```

6.2.2.16.2.5 Example of configuring high availability on the second GCap

- Enter the following command.

```
(gcap-cli) set advanced-configuration high-availability peer-ip fe80::YYY public-ip␣
↪fe80::XXX multicast-group ff02::200 peer-pubkey␣
↪xehXnrigZ0IZZEvWbWri8XegNh0KaAQk8vC6mKj27Ug= shared-secret Xxf4fknh4KoOH2zgrI4Wyw==
```

- Validate.
  The system displays the result.

```
Updating HA configuration
High availability configuration successfully updated
```

6.2.2.16.2.6 Example of eanbling high availability on each GCap

- Enter the following command.

```
(gcap-cli) set advanced-configuration high-availability enable confirm
```

- Validate.
  The system displays the result.

```
Interfaces naming rules updated, reloading configuration
Operation successful.
High availability configuration successfully updated
```

6.2.2.16.2.7 Example of generating a shared secret with the following Python script

```
import base64
import secrets

shared_secret = base64.b64encode(secrets.token_bytes(16))
```

6.2.2.16.3 interface-names

6.2.2.16.3.1 Introduction

The `interface-names` command of the `set advanced-configuration` subgroup enables:

- Assigning the physical interfaces of the GCap:
  - The management interfaces (`gcp0` and `gcp1`)
  - The capture and detection interfaces `mon0` to `monx` or virtual `monvirt`
    This assignment is done with the `set advanced-configuration interface-names <PCI-ID> <nom des interfaces>...` command
- Resetting the current assignment and returning to an automatic assignment
  This assignment is done with the `set advanced-configuration interface-names reset` command

6.2.2.16.3.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.3.3 Commands

```
set advanced-configuration interface-names {{<PCI-ID> <name>...}|reset}
```

6.2.2.16.3.4 Example for manually assigning GCap interfaces

- To verify the need to assign interfaces
  - Enter the following command.

```
(gcap-cli) show interfaces
```

  - Validate.
    The system displays the available capture interfaces.

```
Waiting 10s for interfaces to be up

Name    State    Physical Address  Status Speed Type    Vendor ID Device ID PCI bus
gcp0    Enabled  00:50:56:01:29:01 UP     1Gb   RJ45    0x8086    0x10d3    0b:00.0
gcp1    Disabled 00:50:56:01:29:02 UP     1Gb   RJ45    0x8086    0x10d3    13:00.0
mon0    Enabled  00:50:56:01:29:03 UP     1Gb   RJ45    0x8086    0x10d3    1b:00.0
mon1    Disabled 00:50:56:01:29:04 UP     1Gb   RJ45    0x8086    0x10d3    04:00.0
mon2    Disabled 00:50:56:01:29:05 UP     1Gb   RJ45    0x8086    0x10d3    0c:00.0
mon3    Disabled 00:50:56:01:29:06 UP     1Gb   RJ45    0x8086    0x10d3    14:00.0
monvirt Enabled  N/A               UP     N/A   Virtual N/A       N/A       N/A
```

In this case, the interface names are correct. The existing assignment was successfully made.
Example of a fault case:

```
Name     State    Physical Address  Status Speed Type        Vendor ID Device ID␣
↪PCI bus
eno12399 N/A      68:05:ca:dd:fe:fa UP     1Gb   1000BASE-SX 0x8086    0x1572    ␣
↪31:00.0
eno12409 N/A      68:05:ca:dd:fe:fb UP     1Gb   1000BASE-SX 0x8086    0x1572    ␣
↪31:00.1
eno12419 N/A      68:05:ca:dd:fe:fc UP     1Gb   1000BASE-SX 0x8086    0x1572    ␣
↪31:00.2
eno12429 N/A      68:05:ca:dd:fe:fd UP     1Gb   1000BASE-SX 0x8086    0x1572    ␣
↪31:00.3
eno8303  N/A      ec:2a:72:02:3a:1c DOWN   N/A   RJ45        0x14e4    0x165f    ␣
↪04:00.0
eno8403  N/A      ec:2a:72:02:3a:1d DOWN   N/A   RJ45        0x14e4    0x165f    ␣
↪04:00.1
monvirt  Disabled N/A               UP     N/A   Virtual     N/A       N/A       ␣
↪N/A
```

Note:

Since the interfaces are unassigned, access via SSH connection on the *gcpx* port does not work.
So only:

- Physical access or
- Via web access on the management console (iDRAC) or
- Via SSH connection on the iDRAC port

In this case, the system was unable to associate each of the network interfaces with its name.

- To correct this problem, perform the following procedure.

Astuce:

To make the assignment, the ID *PCI Bus* must be used.

For all interfaces, there are three parts:
- A set of four interfaces corresponding to the capture interfaces `mon0` to `monx`
- A set of two interfaces corresponding to the management interfaces `gcp0` to `gcp1`
- The `monvirt` line which is not to be assigned

Thus, the assignment to be made is as follows, sorted by:
- Vendor ID
- Device ID
- Physical address (PCI bus)

In this example, this gives:

Detected name (Name)	Vendor ID	Device ID	PCI bus	NAME to be assigned
eno12399	0x8086	0x1572	31:00.0	`mon0`
eno12409	0x8086	0x1572	31:00.1	`mon1`
eno12419	0x8086	0x1572	31:00.2	`mon2`
eno12429	0x8086	0x1572	31:00.3	`mon3`
eno8303	0x8086	0x165f	04:00.0	`gcp0`
eno8403	0x8086	0x165f	04:00.1	`gcp1`
monvirt	N/A	N/A	N/A	no assignment

- Enter the following command.

```
(gcap-cli) set advanced-configuration interface-names 31:00.0 mon0 31:00.1 mon1␣
↪31:00.2 mon2 31:00.3 mon3 04:00.0 gcp0 04:00.1 gcp1
```

- Validate.

Danger:

It is possible to invert gcp0 and gcp1, mon0 and mon1 with a bad configuration of this command. The command will be executed but there is loss of ssh connection and loss of pairing between GCap and GCenter.

6.2.2.16.3.5 Example to reset the current assignment and return to an automatic assignment

- Enter the following command.

```
(gcap-cli) set advanced-configuration interface-names reset
```

- Validate.
  The system displays the following message:

```
Network interfaces will be refreshed and corresponding configuration applied
Rebooting in 10 seconds...
You can still abort by pressing CTRL+C.
```

6.2.2.16.4 load-balancing

6.2.2.16.4.1 Introduction

The `load-balancing` command of the `set advanced-configuration` subgroup enables an advanced load balancing configuration of the captured flows per capture interface using load balancing methods (algorithm).

Below are the configuration options:

- The `kernel-native (RPS)` method is configured by default on all capture interfaces
- The `custom (XDP)` method enables advanced configuration:
  - Algorithm `3-tuple`: enables choosing an algorithm with three tuples (VLAN ID + IP Addresses + IP Protocol) for load balancing captured flows
  - algorithm `5-tuple`: enables choosing an algorithm with five tuples (VLAN ID + IP Addresses + IP Protocol + L4 Ports if applicable) for load balancing the captured flows
  - keyword `seed`: enables defining a seed to better distribute the identical traffic

Note:

The functionality is compatible with some GCap models (see model datasheet).

6.2.2.16.4.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.4.3 Command with kernel-native method

```
set advanced-configuration load-balancing method kernel-native
```

6.2.2.16.4.4 Command with custom method (XDP)

```
set advanced-configuration load-balancing method custom algorithm {3-tuple|5-tuple} seed
INTEGER
```

6.2.2.16.4.5 Help

6.2.2.16.4.6 Help on the set advanced-configuration load-balancing command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing  ?
```

- Validate.
  The system displays:

```
Update current load balancing configuration
===========================================

Available commands:
  - mon2: Update load balancing configuration for interface mon2
  - confirm: Accept the risks and confirm running the procedure
```

6.2.2.16.4.7 Help on the set advanced-configuration load-balancing mon2 command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing mon2 ?
```

- Validate.
  The system displays:

```
Update load balancing configuration for interface mon2
======================================================

Available commands:
  - method: Set load balancing method for interface
  - algorithm: Set load balancing algorithm for interface (custom mode only)
  - seed: Set load balancing seed for interface (custom mode only)
  - confirm: Accept the risks and confirm running the procedure
```

6.2.2.16.4.8 Help on the set advanced-configuration load-balancing mon2 method command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing  mon2 method ?
```

- Validate.
  The system displays:

```
Set load balancing method for interface
=======================================

Available commands:
  - kernel-native: Set load balancing to kernel native method (RPS)
  - custom: Set load balancing to custom (XDP)
```

6.2.2.16.4.9 Help on the set advanced-configuration load-balancing mon2 custom command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing  mon2 method custom ?
```

- Validate.
  The system displays:

```
Set load balancing to custom (XDP)
==================================

Available commands:
   - mon2: Update load balancing configuration for interface mon2
   - algorithm: Set load balancing algorithm for interface (custom mode only)
   - seed: Set load balancing seed for interface (custom mode only)
   - confirm: Accept the risks and confirm running the procedure
```

6.2.2.16.4.10 Help on the set advanced-configuration load-balancing mon2 method custom algorithm command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing  mon2 method custom algorithm ?
```

- Validate.
  The system displays:

```
Set load balancing algorithm for interface (custom mode only)
=============================================================

Available commands:
  - 3-tuple: Set load balancing algorithm to 3-tuple
  - 5-tuple: Set load balancing algorithm to 5-tuple
```

6.2.2.16.4.11 Help on the set advanced-configuration load-balancing mon2 method custom algorithm 5-tuple seed command

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing  mon2 method custom algorithm 5-
↪tuple seed ?
```

- Validate.
  The system displays:

```
Set load balancing seed for interface (custom mode only)
========================================================

Available commands:
  - <seed>: Set the load balancing seed (positive integer)
```

6.2.2.16.4.12 Example of applying the custom XDP method with a 5-tuple algorithm for the mon2 interface

- Enter the following command.

```
(gcap-cli) set advanced-configuration load-balancing mon2 method custom algorithm 5-tuple
```

- Validate.

```
This feature is experimental and possibly unstable.
Traffic may become corrupted and detection might fail.
Here be dragons.
Please type 'CONFIRM' in uppercase to continue
```

- Enter `CONFIRM`.

```
Updating load balancing methods
Updating method for interface mon2
Done.

Updating load balancing parameters
Updating parameters for interface mon2
```

6.2.2.16.5 local-rules

6.2.2.16.5.1 Introduction

The `local-rules` command of the `set advanced-configuration` subgroup enables modifying the local rules of the GCap probe.
These rules can be global or per interface.
These modifications are made locally in the GCap and are therefore not visible at the GCenter level.

6.2.2.16.5.2 Details of the Rules

The locally modified rules include:

- In the `Rules:` file, the local Sigflow rules, i.e.:
  - Detection rules and
  - File rebuilding rules
- In the `threshold:` file:
  - Thresholds or limits defined by the keyword "threshold"
  - Deletion rules defined by the keyword "suppress"

6.2.2.16.5.3 Use cases

There are several use cases:

- Making signatures confidential without the GCenter operators being able to see them (need-to-know concept)
- Modifying the local signatures of probes in complex cases.
- If the GCenter is entrusted to a third party and the latter cannot handle markers or signatures of a certain level.

Note:

In multi-tenant mode, it is possible to modify the rules for only one capture interface (configured tenant): there is one file per interface.
In single tenant mode, changes apply to all interfaces at once: there is a single file for all interfaces.

6.2.2.16.5.4 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.5.5 Command

```
set advanced-configuration local-rules TENANT
```

The TENANT parameter can take the following values:

- Single-tenant: all
- Multi-tenant by int: {mon0|mon1|mon2|mon3|monvirt}
- Multi-tenant by vlan:
  - default
  - VLAN X
  - VLAN X Y

6.2.2.16.5.6 General process

When the `set advanced-configuration local-rules ...` command is executed, two rule files are opened successively through the Nano text editor.
The file modifying process is as follows:

- Nano automatically opens the first file.
  It enables modifying the rules of the `Rules:` category , i.e.:
  - detection rules
  - rebuilding rules
- Modify the contents of this file

  Note:

  Once in the interface, a copy/paste of the detection rules can be made.
  There is no limitation in the number of signatures for the interfaces. However, they must not have the same SID as the other rules already present.

- Close (**CTRL** + **X**) after saving
- Nano automatically opens the second file.
  It enables modifying the rules of the `threshold:` category , i.e.:
  - thresholds or limits defined by the keyword "threshold"
  - deletion rules defined by the keyword "suppress"

Note:

Other types of rules can be added to limit or remove certain alerts.
There are:

- **Suppress Rules** that suppress an alert based on the source or destination IP address,
- but also **Threshold Rules** limiting the number of alerts to be displayed based on one or more networks.

6.2.2.16.5.7 Example of modifying the rules in single tenant mode

Important:

Changes made in single tenant mode will be applied to all capture interfaces.

- Enter the following command.

```
(gcap-cli) set advanced-configuration local-rules all
```

- Validate.
  The text editor opens with the file enabling the modification of the rules of the `Rules:` category.
  See the General Process paragraph above.

6.2.2.16.5.8 Example of modifying the rules in multi-tenant mode for the `mon0` interface

Important:

Changes made in multi-tenant mode for the `mon0` interface will only be applied to that interface.
It is therefore possible to set detection rules and thresholds per capture interface.

- Enter the following command.

```
(gcap-cli) set advanced-configuration local-rules mon0
```

- Validate.
  The text editor opens with the file enabling the modification of the rules of the `Rules:` category.
  See the General Process paragraph above.

6.2.2.16.5.9 Example of modifying the multi-tenant rules for vlan 10

Important:

Changes made in multi-tenant mode for vlan 10 will only be applied to that vlan.

- Enter the following command.

```
(gcap-cli) set advanced-configuration local-rules VLAN 10
```

- Validate.
  The text editor opens with the file enabling the modification of the rules of the `Rules:` category.
  See the General Process paragraph above.

6.2.2.16.5.10 Example of modifying the multi-tenant rules for the vlan by default

- Enter the following command.

```
(gcap-cli) set advanced-configuration local-rules default
```

- Validate.
  The text editor opens with the file enabling the modification of the rules of the `Rules:` category.
  See the General Process paragraph above.

6.2.2.16.6 mtu (Maximum Transfert Unit)

6.2.2.16.6.1 Introduction

The `mtu` command in subgroup `set advanced-configuration` enables displaying changing the MTU byte value of enabled network interfaces (`mon0`, `mon1`, ... `monx`, `gcp0`, `gcp1`, clusters).
This value must be between `1280` and `9000` bytes.

Note:

Load Balancing and XDP Filtering features are not supported if the MTU > 3000.

6.2.2.16.6.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.6.3 Command

```
set advanced-configuration mtu {mon0|mon1|mon2|mon3|monvirt}
```

6.2.2.16.6.4 Example of changing the MTU value of the `mon1` interface

- Enter the following command.

  ```
  (gcap-cli) set advanced-configuration mtu mon1 1500
  ```

- Validate.
  The system displays the result.

  ```
  Updating Network MTU configuration to:
        - mon1: 1500
  ```

6.2.2.16.7 packet-filtering

6.2.2.16.7.1 Introduction

The `packet-filtering` command of the `set advanced-configuration` subgroup enables specifying static rules for filtering the flows captured by the capture interfaces.

This enables excluding the flows:

- that are not analysable
- that could saturate the device's resources (CPUs, etc.)

Below are the configuration options:

- **Creating a filter rule**
  To create a filter rule, the following steps must be taken:
  - Set the native vlan
    The `set advanced-configuration packet-filtering interface mon1 change-native-vlan` command enables specifying the untagged 802.1q or 802.1ad VLAN number (nested VLANs) to frames that do not have a VLAN.
  - Define the capture interface `interface`
  - Set the vlan `vlan`
    The syntax for 802.1AD (Q-in-Q) support is X:Y:
    - X is the "outer TAG". "The outer TAG can be tagged as 0x88A8,802.1AD
    - Y is the "inner TAG". "The inner TAG can be tagged as 0x9100, 0x9200, 0x8100 (Cisco)
  - Specify the flow (`prefix`, `port-range`, `protocol`, `ciphered-protocols`)
  - The `confirm` keyword enables the command to be confirmed
- **Deleting a filter rule**
  To delete a filter rule, follow these steps:
  - Define the rule id using the command: `show advanced-config packet-filtering`.
  - Delete a single rule with the rule ID: `set advanced-configuration packet-filtering delete ID`.
  - Delete a group of rules with the syntax: `set advanced-configuration packet-filtering delete ID_BEGIN-ID_END`.

Note:

Packet-filtering functionality is not supported if the MTU > 3000.

6.2.2.16.7.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.2.16.7.3 Command

To set the native vlan: `set advanced-configuration packet-filtering interface {mon0|mon1|mon2|mon3} change-native-vlan VLAN_ID confirm`

`set advanced-configuration packet-filtering interface {mon0|mon1|mon2|mon3} drop vlan VLAN_ID prefix PREFIX_NETWORK port-range {BEGIN:END} confirm`

To add a rule to the `monx` capture interface for filtering encrypted flows in vlan ID: `set advanced-configuration packet-filtering interface {mon0|mon1|mon2|mon3} drop ciphered-protocols vlan VLAN_ID confirm`

To delete a single rule with the rule ID: `set advanced-configuration packet-filtering delete ID`

To delete a group of rules with the syntax: `set advanced-configuration packet-filtering delete {BEGIN-END}`

6.2.2.16.7.4 Example of adding an encrypted flow filtering rule of vlan 110 to the `mon1` capture interface

- Enter the following command.

```
(gcap-cli) set advanced-configuration packet-filtering interface mon1 drop ciphered-
↪protocols vlan 110 confirm
```

- Validate.
  The system displays the result.

```
Adding rules:
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto ESP
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto AH
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto L2TP
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto GRE
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 22:22
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 443:443
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 465:465
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 500:500
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 993:993
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 995:995
- iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 4500:4500
```

6.2.2.16.7.5 Example of defining the native vlan

- Enter the following command.

```
(gcap-cli) set advanced-configuration packet-filtering interface mon1 change-native-vlan␣
↪10
```

- Validate.
  The system displays the result.

```
The following rules will be created:
   - iface mon1 native vlan 10

Do you want to continue? [y/N]
```

- Enter **y**

6.2.2.16.7.6 Example of deleting a filter rule

- Enter the following command.

```
(gcap-cli) show advanced-configuration packet-filtering
```

- Validate.
  The system displays the result.

```
Current XDP filters:
    - 0: iface mon1 native vlan 10
    - 1: iface mon2 native vlan 1
    - 2: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 22:22
    - 3: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 443:443
    - 4: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 465:465
    - 5: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 993:993
    - 6: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 995:995
    - 7: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 500:500
    - 8: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 4500:4500
    - 9: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto GRE
    - 10: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto ESP
    - 11: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto AH
    - 12: iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto L2TP
```

- Enter the following command.

```
(gcap-cli) set advanced-configuration packet-filtering delete 4 confirm
```

- Validate.

```
Deleting the following rules:
  - iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto TCP range 465:465
```

- Enter the following command.

```
(gcap-cli) set advanced-configuration packet-filtering delete 6-9 confirm
```

- Validate.

```
Deleting the following rules:
  - iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 500:500
  - iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto UDP range 4500:4500
```

(suite sur la page suivante)

(suite de la page précédente)

```
  - iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto GRE
  - iface mon1 drop vlan 110 prefix 0.0.0.0/0 proto ESP
```

6.2.2.16.8 rescan-interfaces

6.2.2.16.8.1 Introduction

The `rescan-interfaces` command of the `advanced-configuration` subgroup enables:

- scanning network interfaces
- synchronising the detected network interfaces with the predefined names in the system.

This command is particularly useful if the interfaces are misnamed or out of order. This can happen in some cases with old or unrecognised material.

6.2.2.16.8.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

6.2.2.16.8.3 Command

```
set advanced-configuration rescan-interfaces [no-reboot]
```

6.2.2.16.8.4 Example of scanning GCap interfaces without rebooting

> **Note:**
>
> As the interfaces might be unassigned, access via the SSH connection might not work. Hence, only physical access or access via the management console (iDrac) is possible.

- Enter the following command.

```
(gcap-cli) set advanced-configuration rescan-interfaces no-reboot
```

- Validate.

```
Operation successful.
```

6.2.3 services

6.2.3.1 Introduction

The GCap 'eve-log' are the analysis logs of the network anomaly detection service. These events are time-stamped and sequenced according to the time of capture.
The list of monitored services is as follows:

Service	Function
local-alerts	• Alerts are automatically sent to the GCenter for their processing with appropriate tools. • The local-alerts service enables alerts to be stored locally. • This service, monopolising resources (CPU + disk space), should only be activated to perform advanced diagnostics in collaboration with the Gatewatcher support service. • Remember to switch off this service after use. This service is not started up natively.
eve-generation	• Generation of eve logs and storage of events on the GCap. • Stopping this service also stops the capture of files
eve-compress	• Compression of eve logs on GCap enables compression of eve logs but consumes CPU power • In the event of intermittent connectivity, or any other problem preventing logs from being sent to the GCenter it is advisable to enable this feature to maximise the time the logs are kept on the GCap.
eve-upload	• Sending eve logs to the GCenter. • Stopping this service has no influence on the extraction of files
file-extraction	File extraction by the GCap probe
file-upload	Sending the extracted files to the GCenter
filter-fileinfo	• fileinfo filtering (`event_type: fileinfo` in elasticsearch) • Automatically removes or retains `fileinfo` events about files that would not be retained for analysis by the GCenter • The aim is to reduce the signal to noise ratio and limit the amount of logs sent to the GCenter • These are replicas (`fileinfo.stored: false` in elasticsearch)

Each of these services can be:

- Started: refer to the *start* command
- Stopped: refer to the *stop* command

To view the current status of the services, refer to the *status* command.

6.2.3.2 show

6.2.3.2.1 Introduction

The `show retention-periods` command of the `services` subgroup enables displaying the retention periods for GCap files.
At the end of this retention period, the files are deleted from the GCap.
These values can be configured from the `GCap variables` section on the GCenter.

List of retention periods:

- `unsent` files: rebuilt files not transmitted to the GCenter.

  The default value is 1296000s or 15 days.

- `sent` files: rebuilt files transmitted to the GCenter.

  The default value is 86400s or 24 hours.

- `eve` files: the eve logs.

  The default value is 1296000s or 15 days.

6.2.3.2.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:** N/A

6.2.3.2.3 Command

```
services show retention-periods
```

6.2.3.2.4 Example of displaying the value of the retention periods

- Enter the following command.

```
(gcap-cli) services show retention-periods
```

- Validate.
  The system displays the current values.

```
Current file retention periods:
  - unsent files: 1296000
  - sent files: 86400
  - eve files: 1296000
```

Note:

Periods are expressed in seconds.

6.2.3.3 start

6.2.3.3.1 Introduction

To view the list of services and their relevance and possible restrictions, refer to the *Introduction* section.
The `start` command of the `services` subgroup enables starting a GCap service although this depends on the current status of these services (see *services status* command).
By default, the following services are started:

- The eve-generation service
- The eve-upload service
- The file-extraction service
- The file-upload service

6.2.3.3.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:** the service must be stopped in order to start it

---

6.2.3.3.3 Command

Depending on the complement to the `services start` command, it is possible to perform different operations.

```
services start {eve-generation|eve-upload|file-extraction|file-upload|filter-fileinfo|local-alerts|eve-c
}
```

To start...	complete the `services start` command with...	prerequisites
generation of eve logs	`eve-generation`	None
sending eve logs to GCenter	`eve-upload`	You must activate eve-generation
extraction of files by the GCap	`file-extraction`	None
sending the extracted files to the GCenter	`file-upload`	File-extraction must be activated
fileinfos filtering	`filter-fileinfo`	It is necessary to enable: 1. eve-generation 2. eve-upload 3. file-extraction 4. file-upload
displaying alerts	`local-alerts`	None
compression of eve logs on GCap	`eve-compress`	eve-generation must be activated

---

6.2.3.3.4 Example of starting fileinfos filtering (accessible from the gviewadm account)

- Enter the following command.

```
(gcap-cli) services start filter-fileinfo
```

- Validate.
  The system indicates that the **filter-fileinfo** service is starting.

```
Starting services filter-fileinfo
```

6.2.3.3.5 Example of starting the compression of eve logs on the GCap

- Enter the following command.

```
(gcap-cli) services start eve-compress
```

- Validate.
  The system indicates that the **eve-compress** service is starting.

```
Starting services eve-compress
```

6.2.3.3.6 Example of starting local alerts

- Enter the following command.

```
(gcap-cli) services start local-alerts
```

- Validate.
  The system indicates that the **local-alerts** service is starting.

```
Starting services local-alerts
```

6.2.3.3.7 Example of starting the generation of eve logs at the GCenter

- Enter the following command.

```
(gcap-cli) services start eve-generation
```

- Validate.
  The system indicates that the eve-generation service is starting.

```
Starting service eve-generation
```

6.2.3.3.8 Example of starting the sending of eve logs to the GCenter

- If the eve-generation service is not active then start it, see the procedure above.
- If the eve-generation service is active then continue the procedure.
  - Enter the following command.

    ```
    (gcap-cli) services start eve-upload
    ```

  - Validate.
    The system indicates that the eve-upload service is starting.

    ```
    Starting service eve-upload
    ```

6.2.3.3.9 Example of starting the extraction of files

- Enter the following command.

  ```
  (gcap-cli) services start file-extraction
  ```

- Validate.
  The system indicates that the file-extraction service is starting.

  ```
  Starting service file-extraction
  ```

6.2.3.3.10 Example of starting the sending of the extracted files to the GCenter

- If the file-extraction service is not active then start it, see the procedure above.
- If the file-extraction service is active then continue the procedure.
  - Enter the following command.

    ```
    (gcap-cli) services start file-upload
    ```

  - Validate.
    The system indicates that the `file-upload` service is starting.

    ```
    Starting service file-upload
    ```

6.2.3.4 status

6.2.3.4.1 Introduction

To view the list of services, please refer to the *Introduction* section.
The `status` command of the `services` subgroup enables displaying the status of each of the GCap services:

- Status `up`: enabled status
- Status `down`: disabled status.

6.2.3.4.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:** none

6.2.3.4.3 Command

```
services status {eve-generation|eve-compress|eve-upload|file-extraction|file-upload|filter-fileinfo|local
```

6.2.3.4.4 Example of displaying the status of all services

- Enter the following command.

```
(gcap-cli) services status
```

- Validate.

```
up - Service eve-generation
up - Service eve-upload
up - Service file-extraction
up - Service file-upload
down - Service filter-fileinfo
down - Service local-alerts
down - Service eve-compress
```

6.2.3.4.5 Example of displaying the status of the file-upload service

- Enter the following command.

```
(gcap-cli) services status file-upload
```

- Validate.

```
up - Service file-upload
```

6.2.3.5 stop

6.2.3.5.1 Introduction

To view the list of services, please refer to the *Introduction* section.
The `stop` command of the `services` subgroup enables stopping a GCap service although this depends on the current status of these services (see *services status* command).
By default, the following services are started:

- The eve-generation service
- The eve-upload service
- The file-extraction service
- The file-upload service

6.2.3.5.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:** the service must be started in order to stop it

6.2.3.5.3 Command

```
services stop {eve-generation|eve-upload|file-extraction|file-upload|filter-fileinfo|local-alerts|eve-co
}
```

Depending on the complement to the `services stop` command, it is possible to perform different operations.

To stop...	complete the `services stop` command with...	prerequisites
• generation of eve logs • consequence: file capture stopped	`eve-generation`	None
• sending eve logs to GCenter • no influence on the generation of logs	`eve-upload`	None
• extraction of files by the GCap	`file-extraction`	None
• sending the extracted files to the GCenter • no influence on the extraction of files	`file-upload`	None
• filtering of fileinfos	`filter-fileinfo`	None
• displaying alerts	`local-alerts`	None
• compression of eve logs on GCap	`eve-compress`	None

6.2.3.5.4 Example of stopping the generation of eve logs at the GCenter

- Enter the following command.

```
(gcap-cli) services stop eve-generation
```

- Validate.
  The system indicates that the **eve-generation** service is stopping.

```
Stopping service eve-generation
```

6.2.3.5.5 Example of stopping the sending of eve logs to the GCenter

- Enter the following command.

```
(gcap-cli) services stop eve-upload
```

- Validate.
  The system indicates that the **eve-upload** service is stopping.

```
Stopping service eve-upload
```

6.2.3.5.6 Example of stopping the extraction of files

- Enter the following command.

```
(gcap-cli) services stop file-extraction
```

- Validate.
  The system indicates that the **file-extraction** service is stopping.

```
Stopping service file-extraction
```

6.2.3.5.7 Example of stopping the sending of the extracted files to the GCenter

- Enter the following command.

```
(gcap-cli) services stop file-upload
```

- Validate.
  The system indicates that the **file-upload** service is stopping.

```
Stopping service file-upload
```

6.2.3.5.8 Example of stopping the filtering of fileinfos

- Enter the following command.

```
(gcap-cli) services stop filter-fileinfo
```

- Validate.
  The system indicates that the **filter-fileinfo** service is stopping.

```
Stopping services filter-fileinfo
```

6.2.3.5.9 Example of stopping local alerts

- Enter the following command.

```
(gcap-cli) services stop local-alerts
```

- Validate.
  The system indicates that the **local-alerts** service is stopping.

```
Stopping services local-alerts
```

6.2.3.5.10 Example of stopping the compression of eve logs on the GCap

- Enter the following command.

```
(gcap-cli) services stop eve-compress
```

- Validate.
  The system indicates that the **eve-compress** service is stopping.

```
Stopping services eve-compress
```

6.2.4 system

6.2.4.1 reload-drivers

6.2.4.1.1 Introduction

The `reload-drivers` command of the `system` subgroup enables reloading the network card drivers and resetting the network card statistics.

6.2.4.1.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.4.1.3 Command

```
reload-drivers
```

6.2.4.1.4 Example of restarting a GCap

- Enter the following command.

```
(gcap-cli) system reload-drivers
```

- Validate.
  The system displays the result.

```
Reloading NIC drivers... Please wait a few seconds...
```

  Then after a few seconds, the command prompt is displayed indicating that the drivers were reloaded.

```
(gcap-cli system)
```

6.2.4.2 restart

6.2.4.2.1 Introduction

The `restart` command of the `system` subgroup enables restarting the GCap.
If before start-up the detection engine is activated (**UP** status), it will be activated after start-up.
If the GCap is paired with the GCenter before start-up, it will be paired after start-up.

6.2.4.2.2 Prerequisites

- **User:** setup
- **Dependencies:** None

6.2.4.2.3 Command

```
system restart
```

6.2.4.2.4 Example of restarting a GCap

- Enter the following command.

```
(gcap-cli) system restart
```

- Validate.
  The SSH connection will be interrupted.

6.2.4.3 shutdown

6.2.4.3.1 Introduction

The `shutdown` command of the `system` subgroup enables shutting down the GCap.

> **Important:**
>
> Once the Gcap is turned off, it will need to be turned back on via remote access through the iDRAC.

6.2.4.3.2 Prerequisites

- **User:** setup
- **Dependencies:** the detection engine must be switched off

---

6.2.4.3.3 Command

```
system shutdown
```

6.2.4.3.4 Example of shutting down the GCap.

- Enter the following command.

```
(gcap-cli) system shutdown
```

- Validate.

6.2.4.4 unlock

6.2.4.4.1 Introduction

The `unlock` command of the `system` subgroup enables resetting the lock of the `gview`, `gviewadm` and `setup` accounts after unsuccessful authentication attempts.

6.2.4.4.2 Prerequisites

- **User:** setup
- **Dependencies:** N/A

6.2.4.4.3 Command

```
system unlock {setup|gview|gviewadm}
```

6.2.4.4.4 Example of unlocking the setup account

- Enter the following command.

```
(gcap-cli) system unlock setup
```

- Validate.
  The system displays the result.

```
User setup successfully unlocked
```

6.2.5 monitoring-engine

6.2.5.1 Introduction

The GCap detection engine captures network traffic and analyses it to generate security events such as alerts and metadata.
The `monitoring-engine` command enables:

- Starting the detection engine
- Stopping the detection engine
- Visualising the status of the detection engine

> **Note:**
>
> For this command, there are advanced options (see the `set monitoring-engine` section).
> Once the capture engine is enabled, some GCap configuration commands are no longer accessible.
> This information is indicated by the "Dependencies" field in the description of each command.
> The capture engine must be disabled to make them accessible again.
> If the GCap configuration is changed via the GCenter, the detection engine is reloaded automatically.
> If the GCap device is restarted, there is no impact on the detection engine status.

6.2.5.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:**
  - Add the IP of the GCenter (`set gcenter-ip`).
  - Pair the GCap and GCenter.
  - Choose the GCenter compatibility version.
  - Activate at least one capture interface.

Note:

If the `sanity-checks` option is set to `enable`, the detection engine starts only after verifying that at least one `` `monx` `` capture interface has been activated and that a cable is connected.

6.2.5.3 Command

`monitoring-engine {status|start|stop}`

6.2.5.4 Example of displaying the status of the detection engine

- Enter the following command.

  ```
  (gcap-cli) monitoring-engine status
  ```

- Validate.
  The system displays the engine status:

  ```
  Detection engine is down
  ```

  Meaning:
  - Detection engine `down`: means that the engine status is inactive
  - Detection engine `up`: means that the engine status is active

6.2.5.5 Example of starting the detection engine

The system displays the following command prompt:

```
Monitoring DOWN gcap-name (gcap-cli)
```

The command prompt indicates the status of the detection engine: here it is stopped.

- Enter the following command.

  ```
  (gcap-cli) monitoring-engine start
  ```

- Validate.
- Check the status of the detection engine:
  The system displays the following command prompt:

  ```
  [Monitoring UP] gcap-name (gcap-cli)
  ```

  The command prompt indicates the status of the detection engine: here it is running.

6.2.5.6 Example of stopping the detection engine

The system displays the following command prompt:

```
[Monitoring UP] gcap-name (gcap-cli)
```

The command prompt indicates the status of the detection engine: here it is running.

- Enter the following command.

  ```
  (gcap-cli) monitoring-engine stop
  ```

- Validate.
- Check the status of the detection engine:

  ```
  Monitoring DOWN gcap-name (gcap-cli)
  ```

  The command prompt indicates the status of the detection engine: here it is stopped.

6.2.6 pairing

6.2.6.1 Introduction

The `pairing` command enables configuring the IPsec pairing with the GCenter.
The pairing is done through the `gcp0` interface (not configurable).

6.2.6.2 Prerequisites

- **User:** setup
- **Dependencies:**
  - the detection engine must be switched off
  - the network interfaces must be correctly configured
  - the IP address of the GCenter must be entered via the `set gcenter-ip` command
  - the compatibility of the GCenter must be entered via the `set compatibility-mode` command

6.2.6.3 Command

```
pairing fingerprint FINGERPRINT otp OTP
```

6.2.6.4 Example of pairing a GCap version 2.5.3.107 with a GCenter

- Retrieve the FQDN (hostname + domain) of the GCap via the `show status` command.

  ```
  (gcap-cli) show status
  ```

- Go to the GCenter WEB interface to add the full FQDN (Fully Qualified Domain Name) of the probe
  For more information, please refer to the GCenter documentation.
- Enter the SSH fingerprint of the GCenter in the `pairing` command.
- Enter the generated OTP in the `pairing` command.

  ```
  (gcap-cli) pairing fingerprint XXX otp XXX
  ```

- Validate the pairing with the `show status` command.

```
(gcap-cli) show status
```

For more information on this procedure, refer to the *Pairing procedure between a GCap and a GCenter*.

6.2.7 replay

6.2.7.1 Introduction

A file with the pcap extension is one in which raw network traffic has been captured.
The `replay` command enables:

- Listing the available pcap files
- Asking the detection engine to analyse this network traffic to rebuild the packets contained in this flow
- Replaying it with the possibility of modifying the speed compared to that of the initial capture.

Below are the configuration options:

- **List the available pcap files**
  - `list`
- **Choose the name of the pcap file**
  - `pcap`
- **Choose the replay speed**
  - `speed`
- **Choose a loop replay**
  - `forever`

> **Note:**
>
> Adding pcap is only possible with supported versions of the GCenter software.
> Adding pcap is only possible via the command line with the *root* account, otherwise contact Gatewatcher support.

6.2.7.2 Prerequisites

- **Users:** setup, gviewadm
- **Dependencies:**
  - The detection engine is started (`UP`)
  - The `monvirt` interface is activated
  - At least one pcap file must be present in the pcap directory

---

6.2.7.3 Command

```
replay pcap name.pcap {speed FACTOR} {forever}
```

```
replay list
```

Available commands:

- `forever`: means to replay the pcap file until **CTRL** + **C** is pressed
- `speed x`: x is a number specifying the replay speed of the pcap file (X times the nominal speed)

6.2.7.4 Example of displaying the list of available pcap files

- Enter the following command.

```
[Monitoring UP] GCap-lab (gcap-cli) replay list
```

- Validate.

```
Available pcaps are:

test-dga-v1.pcap
test-malcore-v1.pcap
test-powershell-v1.pcap
test-shellcode-v1.pcap
test-sigflow-v1.pcap
```

The list of the pcap files present is displayed.
The files listed above were installed during a new installation or an update if no other pcap file is present on the GCap.
Each of these files allows you to test a different engine.

> **Note:**
>
> For the test-sigflow-v1.pcap file, it is possible to replay this pcap file but:
>
> - If one of the following 2 signatures is present in the ruleset applied to the Gcap then the alerts at the Gcenter level are visible:
>   - sid:2020716 => ET POLICY Possible External IP Lookup ipinfo.io
>   - sid:2013028 ==> ET POLICY curl User-Agent Outbound
> - If none of these signatures is present in the ruleset then there is no GCenter alert so it will not be known if the sigflow engine is working correctly

6.2.7.5 Example of replaying a pcap file with the capture speed

- Enter the following command.

```
(gcap-cli) replay pcap name.pcap speed 4
```

- Validate.

```
Test start: 2022-05-13 14:49:31.287043 ...
Actual: 38024 packets (43981183 bytes) sent in 5.00 seconds
Rated: 8795627.9 Bps, 70.36 Mbps, 7604.27 pps
Actual: 58291 packets (66785902 bytes) sent in 10.00 seconds
Rated: 6678332.4 Bps, 53.42 Mbps, 5828.87 pps
Actual: 83666 packets (95744520 bytes) sent in 15.02 seconds
Rated: 6374049.4 Bps, 50.99 Mbps, 5569.93 pps
Actual: 110051 packets (125880214 bytes) sent in 20.02 seconds
Rated: 6285776.9 Bps, 50.28 Mbps, 5495.35 pps
Actual: 147566 packets (169410025 bytes) sent in 25.02 seconds
Rated: 6769298.3 Bps, 54.15 Mbps, 5896.45 pps
Actual: 169247 packets (193816539 bytes) sent in 30.03 seconds
Rated: 6453918.8 Bps, 51.63 Mbps, 5635.77 pps
Actual: 195575 packets (223882527 bytes) sent in 35.06 seconds
Rated: 6385197.7 Bps, 51.08 Mbps, 5577.85 pps
Actual: 221886 packets (253884171 bytes) sent in 40.09 seconds
```

(suite sur la page suivante)

(suite de la page précédente)

```
Rated: 6331801.8 Bps, 50.65 Mbps, 5533.77 pps
Actual: 260874 packets (298969988 bytes) sent in 45.11 seconds
Rated: 6627011.6 Bps, 53.01 Mbps, 5782.57 pps
Actual: 280646 packets (321206175 bytes) sent in 50.19 seconds
Rated: 6399274.4 Bps, 51.19 Mbps, 5591.20 pps
Test complete: 2022-05-13 14:50:24.974433
Actual: 300745 packets (344377408 bytes) sent in 53.68 seconds
Rated: 6414493.3 Bps, 51.31 Mbps, 5601.78 pps
Flows: 3774 flows, 70.29 fps, 296049 flow packets, 4696 non-flow
Statistics for network device: injectiface
    Successful packets:        300745
    Failed packets:            0
    Truncated packets:         0
    Retried packets (ENOBUFS): 0
    Retried packets (EAGAIN):  0
```

The system displays the counters approximately every five seconds:

- Throughput in Bps
- Throughput in Mbps
- Throughput in pps (packets)

then the final counters.

6.2.8 help

To obtain help with the available commands, it is possible to:

- Prefix it with `help` (example `help show config-files`)
- Suffix the command with `?` (example `show config-files ?`)

Help enables displaying the available commands and a description of the command in the current context.

6.2.8.1 Use of `?`

The `?` command can be used:

- Alone: in this case, it has the same function as the `help` command
- After the command for which help is to be displayed: suffixing

6.2.8.1.1 Prerequisites for `?`

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.8.1.2 Command ?

- (gcap-cli) ? to display the list of available commands
- (gcap-cli) show status ? to display the help for the status command of the show set

6.2.8.1.3 Using the ? of suffixing

To list the configuration files accessible via the CLI:

- Use the show config-files command followed by ?

```
(gcap-cli) show config-files ?
```

- Validate
  The system displays the following information:

```
(gcap-clInspect configurations
======================

Available commands:
    - Sigflow-config: View detection engine configuration
    - rules-scirius: View user-defined detection ruleset
    - rules-files: View file-related detection ruleset
    - threshold: View threshold configuration files
```

6.2.8.2 Use of help

The help command can be used:

- Alone: in this case, the system displays the commands available in the current level
- Before the command for which the help is to be displayed: prefixing
- After the command for which the help is to be displayed, but --help or -h must be entered

6.2.8.2.1 Prerequisites for help

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.8.2.2 Command help

- (gcap-cli) help to display the list of available commands
- (gcap-cli) show status --help to display the help for the command status of the set show
- (gcap-cli) help show status to display the help for the command status of the set show

6.2.8.2.3 Using `help` alone

- Enter the following command.

```
(gcap-cli) help
```

- Validate.
  The system displays the following information:

```
   CLI entrypoint
   ==============

Available commands:
        - show: Show system configuration
      - set: Modify system configuration
      - services: Manage service
      - system: Handle system operations
          - monitoring-engine: Handle Monitoring Engine
          - help: Display command help message
          - colour: Handle colour support for current CLI session
          - gui: Start a graphical session (deprecated)
          - exit: Exit configuration tool
```

6.2.8.2.4 Example of prefixing: display the commands available in the monitoring-engine context from the root of gcap-cli

- Enter the following command.

```
(gcap-cli) help monitoring-engine
```

- Validate.
  The system displays the following information:

```
   Available commands:
       - start: Start the Monitoring Engine
       - status: View current Monitoring Engine status
```

6.2.8.2.5 Example of suffixing: displaying the information of a command

- Enter the following command.

```
   (gcap-cli system) shutdown --help
```

- Validate.
  The system displays the following information:

```
   Shutdown GCap
```

6.2.9 colour

6.2.9.1 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

The `colour` command enables or disables colours in the output of the current instance of gcap-cli.

6.2.9.2 Command

```
colour {disable|enable}
```

6.2.9.3 Example to display service statuses with colour.

- Enter the following command.

```
(gcap-cli) colour enable
```

- Validate.
  The system then displays the information in colour.

```
    <pre>
    <span style="color:green;">[Monitoring UP]</span> <span style="color:red;">GCap</
↪span><span style="color:blue;"> (gcap-cli)</span> service status
    <span style="color:green;">up</span> - Service eve-generation
    <span style="color:green;">up</span> - Service eve-upload
    <span style="color:green;">up</span> - Service file-extraction
    <span style="color:green;">up</span> - Service file-upload
    <span style="color:red;">down</span> - Service filter-fileinfo
    <span style="color:red;">down</span> - Service eve-compress

    <span style="color:green;">[Monitoring UP]</span> <span style="color:red;">GCap</
↪span><span style="color:blue;"> (gcap-cli)</span> colour disable
    </pre>
```

6.2.9.4 Example for displaying service reports without colour

- Enter the following command.

```
(gcap-cli) colour disable
```

- Validate.
  The system then displays the information without colour(see example below).

```
    <pre>
    [Monitoring UP] GCap (gcap-cli) service status

    up - Service eve-generation
    up - Service eve-upload
    up - Service file-extraction
    up - Service file-upload
```

(suite sur la page suivante)

(suite de la page précédente)

```
    down - Service filter-fileinfo
    down - Service eve-compress
    </pre>
```

6.2.10 gui (deprecated)

6.2.10.1 Introduction

The `gui` command from `gcap-cli` enables the GCap configuration GUI.
The GUI is deprecated.

6.2.10.2 Prerequisites

- **Users:** setup, gviewadm,gview
- **Dependencies:** N/A

6.2.10.3 Command

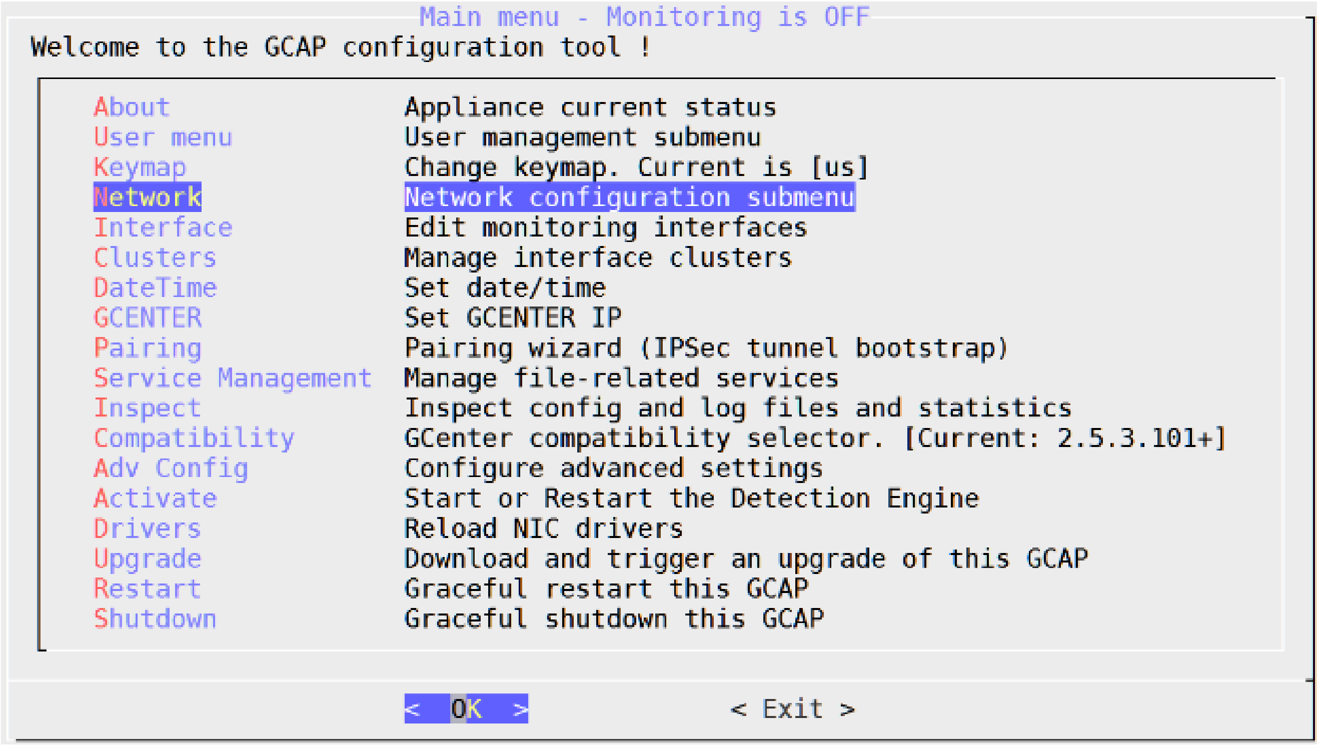
`gui`

6.2.10.4 Example to launch the GCap configuration GUI

- Enter the following command.

```
(gcap-cli) gui
```

- Validate.
  The graphical menu is displayed:



```
                    Main menu - Monitoring is OFF
 Welcome to the GCAP configuration tool !

     About                   Appliance current status
     User menu               User management submenu
     Keymap                  Change keymap. Current is [us]
     Network                 Network configuration submenu
     Interface               Edit monitoring interfaces
     Clusters                Manage interface clusters
     DateTime                Set date/time
     GCENTER                 Set GCENTER IP
     Pairing                 Pairing wizard (IPSec tunnel bootstrap)
     Service Management      Manage file-related services
     Inspect                 Inspect config and log files and statistics
     Compatibility           GCenter compatibility selector. [Current: 2.5.3.101+]
     Adv Config              Configure advanced settings
     Activate                Start or Restart the Detection Engine
     Drivers                 Reload NIC drivers
     Upgrade                 Download and trigger an upgrade of this GCAP
     Restart                 Graceful restart this GCAP
     Shutdown                Graceful shutdown this GCAP

                        <  OK  >           < Exit >
```

6.2.11 exit

6.2.11.1 Introduction

The `exit` command enables:

- Returning to the root (gcap cli) if the prompt is elsewhere in the tree structure
- Leave the SSH session if the prompt is already at the root (gcap-cli)

The **CTRL** + **D** shortcut enables calling the `exit`command.

6.2.11.2 Prerequisites

- **Users:** setup, gviewadm, gview
- **Dependencies:** N/A

6.2.11.3 Command

```
exit
```

6.2.11.4 Example of exiting the "set protocols-selector logging" context

- Enter the following command.

```
(gcap-cli set protocols-selector logging) exit
```

- Validate.
  The prompt changed and shows the root context:

```
(gcap-cli)
```

6.2.11.5 Example of exiting the CLI

- Enter the following command.

```
(gcap-cli) exit
```

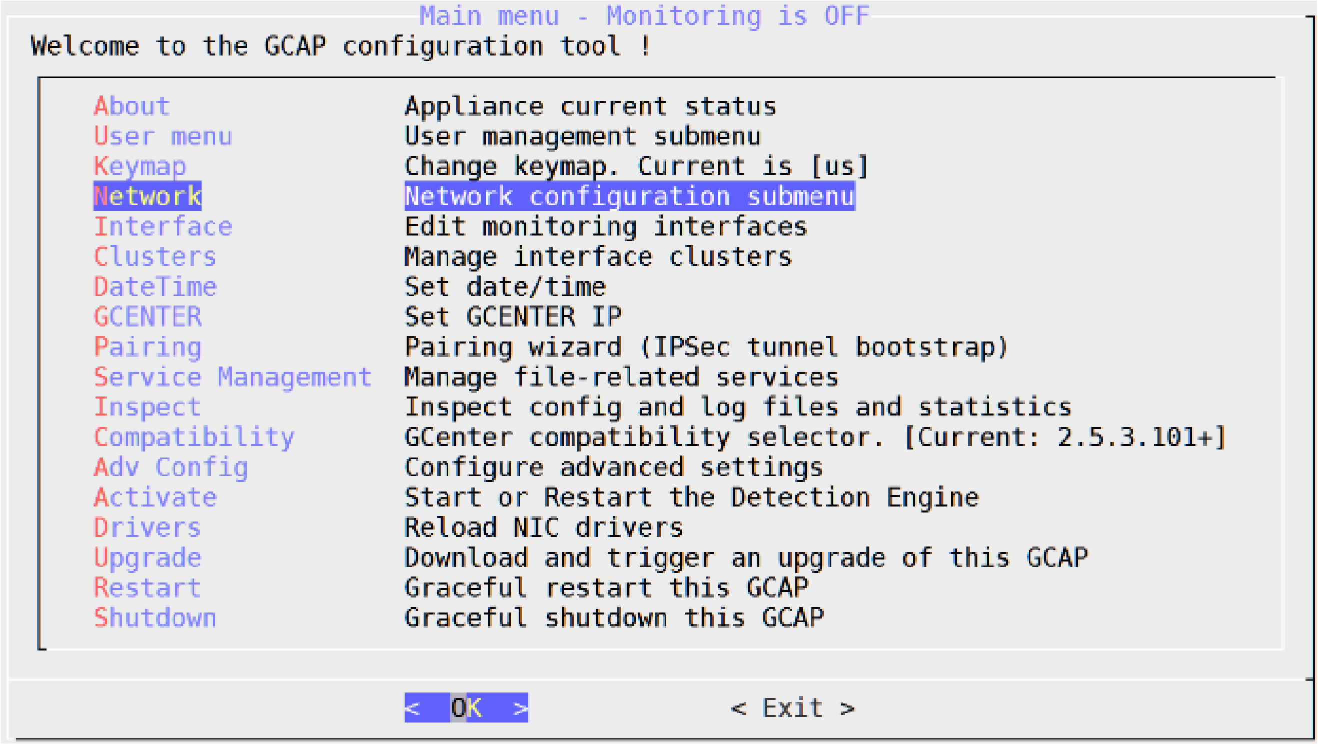
- Validate.

6.3 gui (deprecated)

Important:

The GCap GUI menu is deprecated. This will be removed in version 2.5.3.107.

Below is an overview of the menu in GUI:



```
Main menu - Monitoring is OFF
Welcome to the GCAP configuration tool !

About                 Appliance current status
User menu             User management submenu
Keymap                Change keymap. Current is [us]
Network               Network configuration submenu
Interface             Edit monitoring interfaces
Clusters              Manage interface clusters
DateTime              Set date/time
GCENTER               Set GCENTER IP
Pairing               Pairing wizard (IPSec tunnel bootstrap)
Service Management    Manage file-related services
Inspect               Inspect config and log files and statistics
Compatibility         GCenter compatibility selector. [Current: 2.5.3.101+]
Adv Config            Configure advanced settings
Activate              Start or Restart the Detection Engine
Drivers               Reload NIC drivers
Upgrade               Download and trigger an upgrade of this GCAP
Restart               Graceful restart this GCAP
Shutdown              Graceful shutdown this GCAP

                < OK >                < Exit >
```

To access the menu,use the *GUI* command from the CLI.
Not recommended: the GUI menu can be set as *default*.

Chapter 7

Metrics

7.1 List of metrics comparison version 2.5.3.105 vs 2.5.3.104

Version 2.5.3.105 uses new counters and renames others.
The tables below provide a comparison between version 2.5.3.105 vs. 2.5.3.104 counters.

7.1.1 Internal metrics version 2.5.3.105 vs 2.5.3.104

Name on V105 version	Name on V104 version	Difference between versions
netdata.runtime_proc_net_dev		counter added with V105
netdata.runtime_xdp_filter	netdata.runtime_xdp_filter_local	counter renamed with V105
netdata.runtime_disk_usage	netdata.runtime_disk_usage_local	counter renamed with V105
netdata.runtime_proc_meminfo		counter added with V105
netdata.runtime_proc_loadavg		counter added with V105
netdata.runtime_proc_uptime		counter added with V105
netdata.runtime_proc_vmstat		counter added with V105
netdata.runtime_proc_stat		counter added with V105
netdata.runtime_high_availability		counter added with V105
netdata.runtime_sys_block		counter added with V105
netdata.runtime_proc_net_softnet_stat		counter added with V105
netdata.runtime_suricata	netdata.runtime_suricata_local	counter renamed with V105
netdata.runtime_codebreaker	netdata.runtime_codebreaker_local	counter renamed with V105
netdata.web_thread[1-6]_cpu		counter renamed with V105
netdata.plugin_diskspace_dt		counter renamed with V105
netdata.plugin_diskspace		counter renamed with V105
	netdata.plugin_proc_cpu	counter renamed with V105
	netdata.plugin_proc_modules	counter renamed with V105

7.1.2 System information version 2.5.3.105 vs 2.5.3.104

Name on V105 version	Name on V104 version	Difference between versions
disk_space.**<partition>**		counter added with V105
disk_inodes.**<partition>**		counter added with V105
disk_usage.mountpoint.**<mount>**		counter added with V105
sys_block.blocks.**<disque>**		counter added with V105
proc_stat.processes	system.processes	counter renamed with V105
proc_stat.interrupts	system.intr	counter renamed with V105
proc_stat.cpu.cpu(0-n)	system.cpu.cpu(0-n)	counter renamed with V105
proc_vmstat.swapio	system.swapio	counter renamed with V105
proc_vmstat.pgpio	system.pgpgio	counter renamed with V105
proc_vmstat.pagefaults	mem.pgfaults	counter renamed with V105
proc_uptime.uptime	system.uptime	counter renamed with V105
proc_loadavg.Load_average	system.load	counter renamed with V105
proc_loadavg.Active_processes	system.active_processes	counter renamed with V105
proc_meminfo.RAM	system.ram	counter renamed with V105
proc_meminfo.available	mem.available	counter renamed with V105
proc_meminfo.swap	system.swap	counter renamed with V105
proc_meminfo.kernel	mem.kernel	counter renamed with V105
proc_meminfo.hugepages	mem.transparent_hugepages	counter renamed with V105
	system.io	counter deleted with V105
	system.net	counter deleted with V105

7.1.3 Network information version 2.5.3.105 vs 2.5.3.104

Name on V105 version	Name on V104 version	Difference between versions
proc_net_dev.net_drops.**<iface>**	proc_net_dev_local.net_drops.**<iface>**	counter renamed with V105
proc_net_dev.net_drops.**<iface>**	proc_net_dev_local.net_errors.**<iface>**	counter renamed with V105
proc_net_dev.net_pkts.**<iface>**	proc_net_dev_local.net_pkts.**<iface>**	counter renamed with V105
proc_net_dev.net.**<iface>**	proc_net_dev_local.net.**<iface>**	counter renamed with V105
proc_net_softnet_stat.cpu[0-n].sched		counter added with V105
proc_net_softnet_stat.cpu[0-n].packets		counter added with V105
proc_net_softnet_stat.summed.sched		counter added with V105
proc_net_softnet_stat.summed.packets		counter added with V105

7.1.4 Device and detection information version 2.5.3.105 vs 2.5.3.104

Name on V105 version	Name on V104 version	Difference between versions
high_availability.ha_status		counter added with V105
high_availability.leader_status		counter added with V105
high_availability.last_status		counter added with V105
high_availability.health_status		counter renamed with V105
xdp_filter.dropped_bytes		counter added with V105
xdp_filter.dropped_packets		counter added with V105
xdp_filter.bypassed_half_flows		counter added with V105
codebreaker.shellcode_samples	codebreaker_local.shellcode_samples	counter renamed with V105

7.2 List of available metrics from version 2.5.3.105

7.2.1 Internal metrics

Name	Unit Dimensions	Comments
netdata.runtime_proc_net_dev	run time ms	Execution time of the script for collecting information on the interfaces
netdata.runtime_xdp_filter	run time ms	Execution time of the script for collecting information on XDP filters
netdata.runtime_disk_usage	run time ms	Execution time of the script for collecting information on disk usage
netdata.runtime_proc_meminfo	run time ms	Execution time of the script for collecting information on memory usage
netdata.runtime_proc_loadavg	run time ms	Execution time of the script for collecting information on the GCap load
netdata.runtime_proc_uptime	run time ms	Execution time of the script for collecting information on the uptime
netdata.runtime_proc_vmstat	run time ms	Execution time of the script for collecting information on the virtual memory
netdata.runtime_proc_stat	run time ms	Execution time of the script for collecting information on CPU usage details
netdata.runtime_high_availability	run time ms	Execution time of the script for collecting information on the high availability
netdata.runtime_sys_block	run time ms	Execution time of the script for collecting information on the I/O disks
netdata.runtime_proc_net_softnet_stat	run time ms	Execution time of the script for collecting information on the network stack
netdata.runtime_suricata	run time ms	Execution time of the script for collecting information on Sigflow
netdata.runtime_codebreaker	run time ms	Execution time of the script for collecting information on Codebreaker
netdata.web_thread[1-6]_cpu	user system ms/s	CPU usage time of netdata threads
netdata.plugin_diskspace_dt	duration ms/run	Execution time of the script for collecting information on disk space
netdata.plugin_diskspace	user system ms/s	CPU usage time of the disk space information collection plugin

7.2.2 Details of Sigflow counters

7.2.2.1 Alerts counter details - Number of Sigflow alerts found

Name	Dimensions	Comments
suricata.alert	Alerts.value	Number of Sigflow alerts found

7.2.2.2 Codebreaker samples counter details - Files analysed by Codebreaker

Name	Dimensions	Comments
codebreaker.shellcode_samples	plain encoded	Shellcodes detected without encoding / Shellcodes detected with encoding
codebreaker.powershell_samples	Powershell.value	Number of malicious Powershell scripts detected

7.2.2.3 Details of the Protocols counters - Lists of protocols seen by Sigflow

The following counters display the number of events observed by Sigflow about each protocol.

Name	Dimensions	Units	Comments
suricata.dhcp	DHCP.value	number	DHCP protocol
suricata.dnp3	DNP3.value	number	DNP3 protocol
suricata.dns	DNS.value	number	DNS protocol
suricata.ftp	FTP.value	number	FTP protocol
suricata.http	HTTP.value	number	HTTP protocol
suricata.http2	HTTP2.value	number	HTTP2 protocol
suricata.ikev2	IKEv2.value	number	IKEv2 protocol
suricata.krb5	krb5.value	number	KRB5 protocol
suricata.mqtt	MQTT.value	number	MQTT protocol
suricata.netflow	NETFLOW.value	number	NETFLOW Protocol
suricata.nfs	NFS.value	number	NFS protocol
suricata.rdp	RDP.value	number	RDP protocol
suricata.rfb	RFB.value	number	RFB protocol
suricata.sip	SIP.value	number	SIP protocol
suricata.smb	SMB.value	number	SMB protocol
suricata.smtp	SMTP.value	number	SMTP protocol
suricata.snmp	SNMP.value	number	SNMP protocol
suricata.ssh	SSH.value	number	SSH protocol
suricata.tftp	TFTP.value	number	TFTP protocol
suricata.tls	TLS.value	number	TLS protocol
suricata.tunnel	tunnel.value	number	tunnel protocol

7.2.2.4 Details of the Detection Engine Stats counters - Statistics of Sigflow (monitoring-engine)

Name	Dimensions	Comments
suricata.Status	alive.value	Status of the Sigflow container and the detection engine (boolean)
suricata.total	total.value	Total number of events observed
suricata.fileinfo	• extracted • sent • duplicated	• Number of files extracted • Number of files sent • Number of files duplicated
suricata.received_packets	• ReceivedPackets.value • DroppedPackets.value	• Number of packages captured • Number of packets dropped
suricata.rules	• RulesLoaded.value • RulesFailed.value	• Number of rules loaded and validated • Number of rules that could not be loaded
suricata.tcp_sessions	TcpSessions.value	Number of TCP sessions observed by Sigflow
suricata.tcp_pkt_on_wrong_thread	TcpPktOnWrongThread.value	Misrouted packets by Sigflow
suricata.flows	• FlowTCP.value • FlowUDP.value	• Number of TCP sessions observed • Number of UDP sessions observed

7.2.3 Details of GCap statistics counters and health information.

7.2.3.1 Details of quota counters

Name	Dimensions	Commens
quotas.uid.block	• block.used • block.soft_limit • block.hard_limit	• Number of blocks used • Software limit • Hardware limit
quotas.uid.file	• file.used • file.soft_limit • file.hard_limit	• Number of files used • Software limit • Hardware limit
quotas.uid.grace	• grace.block • grace.file	• Grace time for the blocks • Grace time for the files

7.2.3.2 Details of cpu_stats counters - CPU statistics

Name	Dimensions	Unit	Comments
proc_stat.interrupts	• interrupts	• intr/s	• Number of interruptions per second
proc_stat.processes	• running • blocked	• processes	• Status of the processes
proc_stat.cpu.cpu[0-n]	• softirq • irq • user • system • nice • iowait • idle	• percentage	• Percentage of CPU usage

7.2.3.3 System information

Name	Dimensions	Unit	Comments
sys_block.blocks.<disque>	read written	bytes	I/O on the disk **<disk>**
proc_uptime.uptime	uptime.uptime	seconds	System uptime
disk_inodes.<partition>	avail used reserved for root	inodes	Use of the partition's inodes <partition>
xdp_filter.dropped_bytes	dropped_bytes	bytes	Volume dropped per XDP
xdp_filter.dropped_packets	dropped_packets	pkts	Packets dropped per XDP
xdp_filter.bypassed_half_flows	bypassed_half_flows	half flows	Number of half flows dropped per XDP

7.2.3.4 Details of high_availability counters - High availability (HA) information

Name	Dimensions	Unit	Comments
high_availability.ha_status	ha.status	boolean	HA enabled (1) or not (0)
high_availability.leader_status	ha.health_status	boolean	Node status (0: slave or not configured / 1: leader)
high_availability.health_status	ha.health_status	boolean	Ability of the node to become a leader (0: no or not configured / 1: OK)
high_availability.last_received_status	ha.last_status	seconds	Duration since change of status

7.2.3.5 Details of interface counters - Statistics on network interfaces

Name	Dimensions	Unit	Comments
proc_net_dev.net.**<iface>**	• received • sent	bytes	Traffic on the interface <**iface**>
proc_net_dev.net_drops.**<iface>**	• rx drops • tx drops	pkts	Number of packets lost on the interface <**iface**>
proc_net_dev.net_errors.**<iface>**	• rx errors • tx errors	pkts	Number of packets in error on the interface <**iface**>
proc_net_dev.net_pkts.**<iface>**	• received • sent	pkts	Number of packets on the interface <**iface**>

7.2.3.6 Details of loadavg counters - Statistics on the GCap average load

Name	Dimensions	Comments
proc_loadavg.Load_average	• load.load1 • load.load5 • load.load15	• Average load over the last minute • Average load over the last five minutes • Average load over the last fifteen minutes
proc_loadavg.Active_processes	active_processes.active	Number of active processes

7.2.3.7 Details of meminfo counters - Statistics on RAM

Name	Dimensions	Comments
suricata.memuse	• MemUseTCP.value • MemUseTCPReassembly • MemUseFlow.value • MemUseHTTP.value • MemUseFTP.value	• TCP memory • TCP reassembly memory • Flows memory • HTTP memory • FTP memory
suricata.memcap	• MemCapTCPSession.value • MemCapTCPSegment.value • MemCapFlow.value • MemCapHTTP.value • MemCapFTP.value	• TCP session allocation failures • TCP segment allocation failures • Flow allocation failures • HTTP allocation failures • FTP allocation failures
proc_meminfo.ram	• free • used • cached • buffers	• Unused memory in kilobytes • Memory used • Memory used by the cache • Memory used by operations
proc_meminfo.available	available	Total physical memory in kilobytes
proc_meminfo.swap	• swap_free • swap_used • swap_cached	• swap file available • swap file used • swap file used for caching
proc_meminfo.kernel	• kernel.slab • kernel.kernel_stack • kernel.page_tables • kernel.v_malloc_used	• Memory used by kernel data structures • Memory used by kernel stack allocations • Memory used for page management • Memory used by large memory areas allocated by the kernel
proc_meminfo.hugepages	• hugepages_free • hugepages_used • hugepages.surplus • hugepages.reserved	• Number of huge transparent pages available • Number of huge transparent pages used • Number of extra huge transparent pages • Number of huge transparent pages reserved

7.2.3.8 Details of numastat counters - Statistics on NUMA nodes

Name	Dimensions	Unit	Comments
numa_stat	numa_hit	MiB	Memory successfully allocated in this node as expected
	numa_stat	MiB	• Memory allocated in this node despite process preferences • Each numa_miss has a numa_foreign in another node
	numa_foreign	MiB	Memory intended for this node, but currently allocated in a different node
	other_node	MiB	Memory allocated in this node while a process was running in another node
	interleave_hit	MiB	Interleaved memory successfully allocated in this node
	local_node	MiB	Memory allocated in this node while a process was running on it

7.2.3.9 Details of softnet counters - Statistics on received packets according to processor cores

Name	Dimensions	Unit	Comments
proc_net_softnet_stat.cpu[0-n].packets	• Processed • Dropped • Flow limit count • Process queue lengths	pkts	Packets processed on the relevant cpu
proc_net_softnet_stat.cpu[0-n].sched	• Received RPS (IPI schedules) • Time squeeze	events	network stack events on the relevant cpu
proc_net_softnet_stat.summed.packets	• Processed • Dropped • Flow limit count • Input/Process queue lengths	pkts	Packets processed by the network stack

7.2.3.10 Details of virtualmemory counters - Information on swap space

Name	Dimensions	Unit	Comments
proc_vmstat.swapio	• in • out	pkts	I/O swap
proc_vmstat.pagefaults	• minor • major	faults/s	Memory Page Faults /s

7.3 Retrieving the metrics

The GCap metrics are retrieved through the Netdata session hosted on the GCenter.
To find out about the different access methods, please refer to the *Monitoring* section of the GCenter documentation.
Metrics are collected at a steady interval:

- Every 10 seconds for system-related metrics
- Every minute for detection-related metrics

Chapter 8

Appendices

8.1 Event files

It is possible to consult the event files of the various GCap services via the *show logs* command.

To display...	file name...
detection engine events	detection-engine-logs
kernel events	var-log-kernel
the aggregation of different logs	var-log-messages
GCap authentication information	var-log-auth
the launch information of scheduled tasks	var-log-cron
information about the activity of the various applications used	var-log-daemon
information on the activity of the GCap users	var-log-user
debugging events	var-log-debug

8.1.1 Detection engine events: detection-engine-logs

This log contains the events of the detection engine. They enable obtaining additional information on the status or errors of the detection engine.
Some examples of useful lines:

- End of startup

```
[97] <Info> -- All AFP capture threads are running.
```

- End of rule reload

```
 [76] <Info> -- cleaning up signature grouping structure... complete
 [76] <Notice> -- rule reload complete
```

- Rule loading error

```
[76] <Error> -- [ERRCODE: SC_ERR_UNKNOWN_PROTOCOL(124)] - protocol "dnp3" cannot be used in a␣
↪signature.  Either detection for this protocol is not yet supported OR detection has been␣
↪disabled for protocol through the yaml option app-layer.protocols.dnp3.detection-enabled
 [76] <Error> -- [ERRCODE: SC_ERR_INVALID_SIGNATURE(39)] - error parsing signature "alert␣
↪dnp3 $EXTERNAL_NET any -> $INTERNAL_NET any (msg: "Failing rule"; sid:2000001; rev:1;) from␣
↪file /etc/suricata/rules/local_all.rules at line 1
```

8.1.2 Kernel related events: var-log-kernel

This log contains information about kernel events.
Some examples of useful information:

- Change of link status

```
2022-02-03T12:48:39.578422+00:00 GCap.domain.tld kernel: [ 9149.189652] i40e 0000:17:00.0␣
↪mon0: NIC Link is Down
2022-02-03T12:48:40.457410+00:00 GCap.domain.tld kernel: [ 9150.068228] i40e 0000:17:00.0␣
↪mon0: NIC Link is Up, 10 Gbps Full Duplex, Flow Control: None
```

8.1.3 GCap authentication information: var-log-auth

This log contains the GCap authentication information.
Some examples of useful lines:

- SSH authentication error

```
2022-02-03T14:10:17.680152+00:00 GCap.domain.tld sshd: root [pam]#000[338683]: level=error␣
↪msg="failed to check credentials for \"root\": \"invalid password: password mismatch\""    ␣
↪
2022-02-03T14:10:26.682897+00:00 GCap.domain.tld sshd[338675]: error: PAM: Authentication␣
↪failure for root from 1.2.3.4
2022-02-03T14:10:26.785321+00:00 GCap.domain.tld sshd[338675]: Connection closed by␣
↪authenticating user root 1.2.3.4 port 3592 [preauth]
```

- IPSec events

```
2022-02-03T13:38:10.770453+00:00 GCap.domain.tld charon: 06[IKE] reauthenticating IKE_SA␣
↪GCenter[4]                                                      2022-02-
↪03T13:38:10.771116+00:00 GCap.domain.tld charon: 06[IKE] deleting IKE_SA GCenter[4] between␣
↪10.2.19.152[C=FR, O=GATEWATCHER, CN=lenovo-se350-int-sla.gatewat
cher.com]...2.3.4.5[CN=GCenter.domain.tld.com]
2022-02-03T13:38:13.085957+00:00 GCap.domain.tld charon: 16[IKE] IKE_SA deleted
2022-02-03T13:38:13.141553+00:00 GCap.domain.tld charon: 16[IKE] initiating IKE_SA GCenter[5]␣
↪to 2.3.4.5                                               2022-02-03T13:38:13.
↪364748+00:00 GCap.domain.tld charon: 07[IKE] establishing CHILD_SA GCenter{18} reqid 2
2022-02-03T13:38:14.827308+00:00 GCap.domain.tld charon: 12[IKE] IKE_SA GCenter[5]␣
↪established between 10.2.19.152[C=FR, O=GATEWATCHER, CN=GCap.domain.tld]...2.3.4.
↪5[CN=GCenter.domain.tld.com]
```

8.1.4 Information on the activity of the various applications used: var-log-daemon

This log contains information about the activity of the different applications used.
Some examples of useful lines:

- Configuration synchronisation with the GCenter

```
2022-02-03T16:25:35.583926+00:00 GCap.domain.tld GCenter_gateway.xfer [xfer] : [INFO]␣
↪Successfully rsynced GCap.domain.tld-rules/suricata_configuration.json:
2022-02-03T16:25:35.840272+00:00 GCap.domain.tld GCenter_gateway.xfer [xfer] : [INFO]␣
↪Successfully rsynced GCap.domain.tld-rules-static/v2.0/codebreaker_shellcode.rules:
2022-02-03T16:25:35.840643+00:00 GCap.domain.tld GCenter_gateway.xfer [xfer] : [INFO]␣
```

(suite sur la page suivante)

(suite de la page précédente)

```
↪Codebreaker file /data/containers/suricata/etc/suricata/rules/codebreaker_shellcode.rules␣
↪was identical
2022-02-03T16:25:35.975630+00:00 GCap.domain.tld GCenter_gateway.xfer [xfer] : [INFO]␣
↪Successfully rsynced GCap.domain.tld-rules-static/v2.0/codebreaker_powershell.rules:
2022-02-03T16:25:35.975771+00:00 GCap.domain.tld GCenter_gateway.xfer [xfer] : [INFO]␣
↪Codebreaker file /data/containers/suricata/etc/suricata/rules/codebreaker_powershell.rules␣
↪was identical
```

8.1.5 User activity information: var-log-user

This log contains information about the activity of the GCap users.
Some examples of useful lines:

- Detection engine start-up

```
2022-02-03T14:18:26.428461+00:00 GCap.domain.tld root: [GCap_suricata_tools.suricata-INFO]␣
↪Detection Engine successfully started!
```

- Actions performed via the `gcap-cli` command

```
2022-02-03T16:47:50.636706+00:00 GCap.domain.tld GCap-setup (root) [main main.py handle_shell␣
↪656] : [GCap_cli.main-NOTICE] Starting CLI
2022-02-03T16:47:50.636768+00:00 GCap.domain.tld GCap-setup (root) [main main.py handle_shell␣
↪676] : [GCap_cli.main-INFO] Acquiring lock                            2022-02-03T16:47:50.
↪636832+00:00 GCap.domain.tld GCap-setup (root) [main main.py handle_shell 686] : [GCap_cli.
↪main-INFO] Running single CLI command
2022-02-03T16:47:50.784347+00:00 GCap.domain.tld GCap-setup (root) [main main.py default 530]␣
↪: [GCap_cli.main-NOTICE] [user root] Running CLI command 'show logs var-log-kernel'        ␣
↪                                                                                            ␣
↪                                                                       2022-02-03T16:47:50.
↪784889+00:00 GCap.domain.tld GCap-setup (root) [inspect inspect.py run 332] : [GCap_setup.
↪inspect-NOTICE] Starting inspect procedure
2022-02-03T16:47:50.784930+00:00 GCap.domain.tld GCap-setup (root) [inspect inspect.py run␣
↪339] : [GCap_setup.inspect-NOTICE] Selecting inspection action: `View kernel logs (/var/log/
↪kern.logs)`
2022-02-03T16:47:51.714026+00:00 GCap.domain.tld GCap-setup (root) [inspect inspect.py run␣
↪336] : [GCap_setup.inspect-NOTICE] Stopping inspect procedure
2022-02-03T16:47:51.718373+00:00 GCap.domain.tld GCap-setup (root) [main main.py handle_shell␣
↪710] : [GCap_cli.main-NOTICE] [user root] Stopping CLI
```

8.1.6 Debug events: var-log-debug

This log contains debug events.
This entry is mainly used by support during advanced troubleshooting.

8.1.7 Aggregation of different logs: var-log-messages

This log contains the aggregation of the different logs listed above.

8.1.8 Scheduled task start information: var-log-cron

This log contains the launch information of scheduled tasks.

Chapter 9

Glossary

CLI
The Command Line Interface (CLI) is the means used to administer and configure the GCap. It is the set of commands in text mode.

FQDN
The Fully Qualified Domain Name (FQDN) refers to the host.domain name.

GCap
GCap is the detection probe for the Trackwatch/Aioniq solution. It retrieves the network flow from the TAP and reconstructs the files it sends to the GCenter.

GCenter
GCenter is the component administering the GCap and performing the analysis of the files sent by the GCap.

gview
Account name intended for an operator.

gviewadm
Account name intended for a manager.

MTU
The Maximum Transfer Unit (MTU) is the largest packet size that can be transmitted at one time, without fragmentation, over a network interface.

OTP
The One Time Password (OTP) is a single-use password defined on the GCenter.

RAID1
RAID 1 consists of using n redundant disks. Each disk in the cluster contains exactly the same data at all times, hence the use of the word "mirroring".

RAID5
RAID 5 uses several hard disks (at least 3) grouped together in a cluster to form a single logical unit. The data is duplicated and allocated to two different disks.

setup
Account name intended for a system administrator.

SIGFLOW
The detection engine, also called Sigflow, is responsible for rebuilding the files and is also one of the engines for intrusion detection.

TAP
The Test Access Point (TAP) is a passive device enabling a network flow to be duplicated.

`GCap Documentation : pdf format`

Index