



ENTRUST

KeySafe 5

KeySafe 5 v1.6.1 Metrics

15 January 2026

Table of Contents

1. Introduction	1
2. OpenMetrics	2
2.1. nshield_hsm	2
2.2. nshield_error_conditions	2
2.3. nshield_uptime_seconds	2
2.4. nshield_commands	2
2.5. nshield_replies	2
2.6. nshield_objects_stored	3
2.7. nshield_objects_destroyed	3
2.8. nshield_current_clients	3
2.9. nshield_current_clients_limit	3
2.10. nshield_current_crypto_clients	4
2.11. nshield_audit_db_free_bytes	4
2.12. nshield_audit_db_used_bytes	4
2.13. nshield_queue_in_progress	4
3. Host statistics	5
3.1. nshield_current_clients	5
3.2. nshield_current_crypto_clients	5
3.3. nshield_audit_db_free_bytes	5
3.4. nshield_audit_db_used_bytes	5
3.5. nshield_connection_commands	5
3.6. nshield_connection_replies	6
3.7. nshield_host	6
4. System load	7
4.1. nshield_cpu_load_per_hsm	7
5. Temperatures	8
5.1. nshield_temperature_celsius	8
5.1.1. nshield_temperature_celsius: Module	8
5.1.2. nshield_temperature_celsius: Connect (chassis)	8
5.2. nshield_temperature_limit_celsius	8
5.3. nshield_max_temperature_celsius	9
5.4. nshield_min_temperature_celsius	9
6. Electrical	10
6.1. nshield_platform_voltage_volts	10
6.1.1. stattree mapping	10
7. Fans	11
7.1. nshield_fan_speed_rpm	11

7.2. nshield_fan_speed_limit_rpm	11
8. Memory	12
8.1. nshield_module_mem_bytes	12
8.2. nshield_module_mem_alloc_kernel_bytes	12
8.3. nshield_chassis_mem_alloc_kernel_bytes	12
8.4. nshield_module_mem_alloc_user_bytes	12
8.5. nshield_module_mem_alloc_user_bytes	13
9. Storage	14
9.1. nshield_module_nvram_free_bytes	14
9.2. nshield_module_nvram_erase_per_endurance	14
9.3. nshield_module_worn_blocks_per_nvram	14
10. Internal software statistics	15
10.1. nshield_pci_irqs	15
10.2. nshield_pci_unhandled_irqs	15
10.3. nshield_pci_read_reconnect	15
10.4. nshield_AIS31_preliminary_alarms	15
10.5. nshield_correctable_memory_errors	16
10.6. nshield_spi_communication_attempts	16
11. Metrics	17
11.1. Labels	17
11.2. Historical metrics	17
11.3. Current metrics	18
12. Alerts	19
12.1. HSM PSU Failure	19
12.2. HSM Fan Failure	19
12.3. HSM Chassis Battery	20
12.4. HSM Fan Speed	20
12.5. HSM Memory Usage Percentage	21
12.6. HSM Temperature Percentage	21
12.7. HSM Queue Percentage	22
12.8. HSM Objects Count	23
12.9. Host Hardserver	23
12.10. HSM Liveness	24
12.11. Host Liveness	24
12.12. Licence Expiry	25
12.13. HSM Client Licences Remaining	25
12.14. Certificate Expiry	26

1. Introduction

The following tables display metric exposure in KeySafe 5, the source of the metric, and whether it was available in nShield Monitor or SNMP.

In the tables, the term "resource" refers to whether a statistic belongs to an HSM or a host.

The HSM metrics can apply to the actual HSM ("module") or the HSM "chassis".

2. OpenMetrics

2.1. nshield_hsm

The labels on the HSM.

Type	Details
Information	Has the label "label".

2.2. nshield_error_conditions

Type	Details	Labels	stattree node	stattree ID
stateset	<ul style="list-style-type: none"> failed okay 	source="psu_failed"	HostEnvStats	PSUFailure

2.3. nshield_uptime_seconds

The length of elapsed time since the HSM was last reset. This does not include the HSM chassis.

Type	Unit	Details	stattree node	stattree ID	SNMP MIB
counter	seconds		ModuleEnvStats	Uptime	moduleStat- sTable > uptime

2.4. nshield_commands

The total number of commands sent for processing from a client to the server or from the server to an HSM. The number of commands currently being processed is the **CmdCount** minus the **ReplyCount**.

For an HSM, this is the number of commands received from any client.

Type	Details	stattree node	stattree ID
counter		ModuleJobStats	CmdCount

2.5. nshield_replies

The total number of replies sent from a server to a client or from an HSM to a server.

For an HSM, this is the number of replies sent to any client.

Type	Details	stattree node	stattree ID
counter		ModuleJobStats	ReplyCount

2.6. nshield_objects_stored

The number of times the object store has had a new object put into it.

Type	Details	stattree node	stattree ID
counter		ModuleObjStats	ObjectsCreated

2.7. nshield_objects_destroyed

The number of items that have been deleted from the HSM's object store and had their corresponding memory released.

Type	Details	stattree node	stattree ID
counter		ModuleObjStats	ObjectsDestroyed

2.8. nshield_current_clients

The number of client connections currently made to the Connect hardserver.

Type	stattree node	stattree ID
gauge	ServerGlobals	ClientCount

2.9. nshield_current_clients_limit

The number of licensed client connections available.

Type	stattree node	stattree ID
counter	ServerGlobals	MaxClients

2.10. nshield_current_crypto_clients

The number of licensable clients that are currently connected, including both active and parked sessions.

This is only relevant when reported from a hardserver with remote clients that have an image version of 13.5 or later.

Type	stattree node	stattree ID
gauge	ServerGlobals	CryptoClientCount

2.11. nshield_audit_db_free_bytes

Type	Unit	stattree node	stattree ID
gauge	bytes	ServerGlobals	AuditDBFreeSpaceMB

2.12. nshield_audit_db_used_bytes

Type	Unit	stattree node	stattree ID
gauge	bytes	ServerGlobals	AuditDBUsedSpaceMB

2.13. nshield_queue_in_progress

All jobs currently in progress on the HSM, including jobs from the SEE machine.

Type	stattree node	stattree ID
gauge	ModuleServerStats	JobsOutstanding

3. Host statistics

3.1. nshield_current_clients

The number of client connections currently made to the server.

Type	stattree node	stattree ID
gauge	ServerGlobals	ClientCount

3.2. nshield_current_crypto_clients

The number of licensable clients that are currently connected, including both active and parked sessions.

This is only relevant when reported from a hardserver with remote clients that have an image version of 13.5 or later.

Type	Unit	stattree node	stattree ID
gauge	bytes	ServerGlobals	CryptoClientCount

3.3. nshield_audit_db_free_bytes

Type	Unit	stattree node	stattree ID
gauge	bytes	ServerGlobals	AuditDBFreeSpaceMB

3.4. nshield_audit_db_used_bytes

Type	Unit	stattree node	stattree ID
gauge	bytes	ServerGlobals	AuditDBUsedSpaceMB

3.5. nshield_connection_commands

The total number of commands sent for processing from a client to the server.

Type	Details	stattree node	stattree ID
counter	Has label "connection".	Connections	CmdCount

3.6. nshield_connection_replies

The total number of replies sent from the server to the client.

Type	Details	stattree node	stattree ID
counter	Has label "connection".	Connections	ReplyCount

3.7. nshield_host

The labels on the host.

Type	Details
info	Has label "label".

4. System load

4.1. nshield_cpu_load_per_hsm

The current processing load on the HSM. This is represented as a number from 0 to 100.

HSMs typically contain several different types of processing resources, such as the main CPU and RSA acceleration. This means that reporting on HSM load can be imprecise.

Normally, HSMs report 100% CPU load when all RSA processing capacity is occupied. When performing non-RSA tasks, the main CPU and other resources, such as the random number generator, can become saturated without this metric reaching 100%.

Type	Details	Labels	stattree node	stattree ID
gauge	A value between 0 and 1. Has label "source".	source="module"	ModuleJobStats	CPULoadPercent
		source="chassis"	HostEnvStats	CPULoadPercent

5. Temperatures

5.1. nshield_temperature_celsius

The current temperature of different parts of the HSM.

Type	Units	Details
gauge	celsius	Has label "sensor".

See the following sections for more details about the labels.

5.1.1. nshield_temperature_celsius: Module

The current temperature of the main circuit board. First-generation HSMs do not have a temperature sensor and so do not return temperature statistics.

Labels	stattree node	stattree ID
sensor="module_cpu_temp"	ModuleEnvStats	CurrentCPUtemp1
sensor="module_module_sp_temp"	ModuleEnvStats	TempSP
sensor="module_crypto_co_proc_temp"	ModuleEnvStats	CurrentCPUtemp2

5.1.2. nshield_temperature_celsius: Connect (chassis)

The ambient sensors for the chassis.

Labels	stattree node	stattree ID
sensor="chassis_left"	HostEnvStats	CurrentTempC
sensor="chassis_right"		CurrentTemp2C

5.2. nshield_temperature_limit_celsius

The minimum and maximum acceptable temperature values for each sensor.

Type	Units	Details	Labels
gauge	celsius	Has labels "sensor" and "limit".	For sensor, see nshield-temperature_celsius limit="maximum"

5.3. nshield_max_temperature_celsius

The maximum temperature recorded by the HSM's temperature sensor. This is stored in non-volatile memory and is cleared when the unit is initialized.

Type	Units	Details
gauge	celsius	Has label "sensor".

See the following table for more details about the labels:

Labels	stattree node	stattree ID
sensor="module_cpu_temp"	ModuleEnvStats	MaxTempC
sensor="chassis_left"	HostEnvStats	MaxTempC
sensor="chassis_right"	HostEnvStats	MaxTemp2C

5.4. nshield_min_temperature_celsius

The minimum temperature recorded by the HSM's temperature sensor. This is stored in non-volatile memory and is cleared when the unit is initialized.

Type	Units	Details
gauge	celsius	Has label "sensor".

See the following table for more details about the labels:

Labels	stattree node	stattree ID
sensor="module_cpu_temp"	ModuleEnvStats	MinTempC
sensor="chassis_left"	HostEnvStats	MinTempC
sensor="chassis_right"	HostEnvStats	MinTemp2C

6. Electrical

6.1. nshield_platform_voltage_volts

6.1.1. stattree mapping

Type	Units	Details
gauge	volts	Has label "voltage_sensor".

See the following table for more details about the labels:

Labels	stattree node	stattree ID
voltage_sensor="cpu_core"	ModuleEnvStats	CPUVoltage1
voltage_sensor="t1022_ifc_io"		CPUVoltage2
voltage_sensor="t1022_serdes"		CPUVoltage3
voltage_sensor="t1022_serdes_io"		CPUVoltage4
voltage_sensor="fpga_serdes_core"		CPUVoltage5
voltage_sensor="fpga_serdes_io"		CPUVoltage6
voltage_sensor="msp_avcc"		CPUVoltage7
voltage_sensor="ddr4_access"		CPUVoltage8
voltage_sensor="ddr4_io"		CPUVoltage9
voltage_sensor="pci_bus"		CPUVoltage10
voltage_sensor="module_battery"		CPUVoltage11

7. Fans

7.1. nshield_fan_speed_rpm

The fan speed for each fan in the HSM.

KeySafe 5 assumes fan speeds greater than 120,000rpm are errors, and instead reports a speed of zero.

Type	Units	Details
gauge	rpm	Has label "fan_id".

Labels	stattree node	stattree ID	EnvMon
fan_id="chassis1"	HostEnvStats	CurrentFanRPM	fan1_rpm
fan_id="chassis2"		CurrentFan2RPM	fan2_rpm
fan_id="chassis3"		CurrentFan3RPM	fan3_rpm
fan_id="chassis4"		CurrentFan4RPM	fan4_rpm

7.2. nshield_fan_speed_limit_rpm

The fan speed limits for each fan in the HSM.

These are hardcoded for each HSM type.

Type	Units	Details	Labels
gauge	rpm	Has labels "fan_id" and "limit".	For fan_id, see nshield_fan_speed_rpm . limit = "maximum" and "minimum".

8. Memory

8.1. nshield_module_mem_bytes

The total amount of RAM, allocated and free, available to the HSM. This is equal to the installed RAM size, minus various fixed overheads.

It is a static value that is calculated by KeySafe 5.

Type	Units	stattree node	stattree ID
gauge	bytes	ModuleEnvStats	MemTotal

8.2. nshield_module_mem_alloc_kernel_bytes

The total amount of RAM allocated for kernel use, or non-SEE use, in a module. This is mainly used for the object store, for example, for keys and logical tokens, and for big-number buffers.

Type	Units	stattree node	stattree ID
gauge	bytes	ModuleEnvStats	MemAllocKernel

8.3. nshield_chassis_mem_alloc_kernel_bytes

The total amount of RAM allocated for kernel use, or non-SEE use, in a module. This is mainly used for the object store, for example, for keys and logical tokens, and for big-number buffers.

Type	Units	stattree node	stattree ID
gauge	bytes	HostEnvStats	MemAllocKernel

8.4. nshield_module_mem_alloc_user_bytes

The total amount of RAM allocated for user-mode processes in the module. This will be zero for non-SEE use.

This value includes the size of the SEE Machine image and the total heap space available to it. The module's kernel does not know, and therefore cannot report, how much of the user-mode's heap is currently free and how much is in use.

Type	Units	stattree node	stattree ID
gauge	bytes	ModuleEnvStats	MemAllocUser

8.5. nshield_module_mem_alloc_user_bytes

The total amount of RAM allocated for user-mode processes in the module. This will be zero for non-SEE use.

This value includes the size of the SEE Machine image and the total heap space available to it. The module's kernel does not know, and therefore cannot report, how much of the user-mode's heap is currently free and how much is in use.

Type	Units	stattree node	stattree ID
gauge	bytes	HostEnvStats	MemAllocUser

9. Storage

9.1. nshield_module_nvram_free_bytes

The total amount of free space in the NVRAM of the HSM.

This is only available on XC and nShield 5 HSM variants.

Type	Units	stattree node	stattree ID
gauge	bytes	ModuleEnvStats	NVMFreeSpace

9.2. nshield_module_nvram_erase_per_endurance

The wear level of the HSM's NVRAM, expressed as a percentage of the "erase count:endurance" ratio.

This is only available on XC and nShield 5 HSM variants.

Type	Details	stattree node	stattree ID
gauge	A value between 0 and 1	ModuleEnvStats	NVMWearLevel

9.3. nshield_module_worn_blocks_per_nvram

The percentage of worn blocks in the NVRAM of the HSM

This is only available on XC and nShield 5 HSM variants.

Type	Details	stattree node	stattree ID
gauge	A value between 0 and 1	ModuleEnvStats	NVMWornBlocks

10. Internal software statistics

10.1. nshield_pci_irqs

The number of interrupts from the host. This is approximately equal to the total of `HostReadCount` and `HostWriteCount`.

This is only applicable to PCI HSMs.

Type	Details	stattree node	stattree ID
counter		ModulePCISStats	HostIRQs

10.2. nshield_pci_unhandled_irqs

The number of unidentified interrupts from the host. If this reports a nonzero value, it is likely that there is a problem with a driver or the PCI bus.

This is only applicable to PCI HSMs.

Type	Details	stattree node	stattree ID
counter		ModulePCISStats	HostUnhandledIRQs

10.3. nshield_pci_read_reconnect

The number of deferred reads that have now completed. This should be the same as `HostReadDeferred`, or one less than it if there is a currently deferred read.

This is only applicable to PCI HSMs.

Type	Details	stattree node	stattree ID
counter		ModulePCISStats	HostReadReconnect

10.4. nshield_AIS31_preliminary_alarms

The number of times the AIS31 random number test has failed. Because this test is a statistical test, a small number of failures is expected. If it fails too often, it will trigger a SOS-HRAO alarm and the module will fail.

Type	Details	stattree node	stattree ID
counter		ModuleEnvStats	AIS31PrelimAlarms

10.5. nshield_correctable_memory_errors

The number of correctable memory errors that have been corrected by the error checking and correction (ECC) mechanisms. Typically, this count should be 0, although a small number of errors are to be expected occasionally. If this count increases rapidly, by multiple thousands per second, there has been a malfunction.

Type	Details	stattree node	stattree ID
counter		ModuleEnvStats	MceCount

10.6. nshield_spi_communication_attempts

The number of times the main processor on an XC module has had to repeat an attempt to communicate with the security processor due to a communication failure. Loss of communication between the main processor and the security processor cause the module to enter an alarm state and fail. This sometimes triggers an SOS-HV alarm.

Type	Details	stattree node	stattree ID
counter		ModuleEnvStats	SpiRetries

11. Metrics

Metrics can be viewed for single HSMs or summed across all HSMs.

11.1. Labels

- esn: The ESN of the HSM
- sensor: The CPU temperature sensor providing the reading
- voltage_type: The type of voltage being measured
- fan_id: The fan number
- source: Whether the metric came from the chassis or the module
- limit: Whether the value is the minimum or maximum limit.
- host: The UUID of the host

11.2. Historical metrics

nShield Monitor History	Notes	KeySafe5 historical metric type	Valid parameters	OpenMetric	Supported HSM types
Object Count	The difference between the number of objects created and the number of objects destroyed.	Objects Stored	esn, vcm	HSM: <code>nshield_objects_stored</code>	Full, tenant
		Objects Destroyed	esn, vcm	HSM: <code>nshield_objects_destroyed</code>	Full, tenant
Command Count		Commands Received	esn, vcm	HSM: <code>nshield_commands</code>	Full, tenant
		Replies sent	esn, vcm	HSM: <code>nshield_replies</code>	Full, tenant
	Metric reported per connection	Client Commands Received	host	Host: <code>nshield_connection_commands</code>	Not applicable
	Metric reported per connection	Client Replies Sent	host	Host: <code>nshield_connection_replies</code>	Not applicable
		Job Queue	esn, vcm	HSM: <code>nshield_queue_in_progress</code>	Full, tenant

11.3. Current metrics

The instantaneous value of metrics, reported in a way that enables the KeySafe5 UI to determine whether they are "good" or "bad".

KS5 current metric type	Notes	Valid parameters	OpenMetrics used	Supported HSM types
HSM Fan Speed	A comprehensive view of current HSM fan speeds.	esn	<code>nshield_fan_speed_rpm</code> <code>nshield_fan_speed_limit_rpm</code>	Full, platform
HSM Temperature Percentage	A comprehensive view of HSM temperature, reported as a percentage of the maximum temperature reached.	esn	<code>nshield_temperature_celsius</code> <code>nshield_temperature_limit_celsius</code>	Full, platform
HSM Memory Usage Percentage	The percentage of module memory currently in use.	esn	<code>nshield_module_mem_bytes</code> <code>nshield_module_mem_alloc_kernel_bytes</code> <code>nshield_module_mem_alloc_user_bytes</code>	Full, platform

12. Alerts

The alerts are listed here by their KeySafe 5 alert type.

The "nShield Monitor Alert" row, where available, displays the legacy alert name from nShield Monitor.

Other data returned

Data marked with "*" are returned only as part of the summary and not as a specific alert record field. For more information about what data is returned, see [Labels](#).

12.1. HSM PSU Failure

This alert has a resolution.

Valid parameters	for
OpenMetrics used	nshield_error_conditions(source="psu_failed")
Basic Prometheus query	nshield_error_conditions{source="\psu_failed\", nshield_error_conditions="\failed\"} == 1
Other data returned	esn
Summary returned	HSM PSU has failed
nShield Monitor alert	snmp psuAlert / NShieldPowerSupplyFailure

12.2. HSM Fan Failure

Valid parameters	for
OpenMetrics used	nshield_error_conditions(source="fanX")
Basic Prometheus query	nshield_error_conditions{source="\fanX\", nshield_error_conditions="\failed\"} == 1

Other data returned	esn, source*
Summary returned	HSM Fan {source} has failed.

12.3. HSM Chassis Battery

Valid parameters	for
OpenMetrics used	nshield_error_conditions(source="chassis_battery_low")
Basic Prometheus query	nshield_error_conditions{source=\"chassis_battery_low\", nshield_error_conditions=\"failed\"} == 1
Other data returned	esn
Summary returned	HSM chassis battery has failed

12.4. HSM Fan Speed

This alert has a resolution.

The alert is triggered if the fan speed drops below the minimum limit or exceeds the maximum limit.

Valid parameters	for, over
OpenMetrics used	nshield_fan_speed_rpm, nshield_fan_speed_limit_rpm
Basic Prometheus query	nshield_fan_speed_rpm > ignoring(limit) nshield_fan_speed_limit_rpm{limit="maximum"}, nshield_fan_speed_rpm < ignoring(limit) nshield_fan_speed_limit_rpm{limit="minimum"} (2 queries for each trigger)
Other data returned	esn, fan_id
Summary returned	HSM Fan {fan_id} maximum limit exceeded. HSM Fan {fan_id} is below its minimum limit.

nShield Monitor alert	snmp fanfailureAlert / NShieldXCFanSpeedZero
------------------------------	--

12.5. HSM Memory Usage Percentage

This is the sum of the kernel and user memory expressed as a percentage of the total amount of available memory, or $(\text{kernel} + \text{user}) / \text{total} * 100$.

Valid parameters	min < x or max > y, for, over
OpenMetrics used	nshield_module_mem_bytes, nshield_module_mem_alloc_kernel_bytes, nshield_module_mem_alloc_user_bytes
Basic Prometheus query	$((\text{nshield_module_mem_alloc_kernel_bytes} + \text{nshield_module_mem_alloc_user_bytes}) / \text{nshield_module_mem_bytes}) * 100 > \text{max}$
Other data returned	esn
Summary returned	HSM Memory Usage exceeds {max}%. HSM memory usage is below {min}%.
nShield Monitor alert	snmp memoryUsageHighAlert / memoryUsageOkAlert

12.6. HSM Temperature Percentage

The maximum value for any sensor. The sensor value is calculated the current value divided by the limit maximum value: $\text{maximum}(100 * \text{current}(\text{sensor}) / \text{limit maximum}(\text{sensor})) > \text{max}$.

The maximum value is a percentage between 0 and 150.

Valid parameters	max > y, for, over
OpenMetrics used	nshield_temperature_celsius, nshield_temperature_limit_celsius

Basic Prometheus query	$(nshield_temperature_celsius / ignoring(limit) nshield_temperature_limit_celsius\{limit="maximum"\}) * 100 > max$
Other data returned	esn, sensor*
Summary returned	HSM Sensor: "{sensorName}" temperature exceeds {max}% of maximum temperature.
nShield Monitor alert	NShieldTemperaturePeak

12.7. HSM Queue Percentage

This alert type maps to two nShield Monitor alerts:

DeviceNShieldUtilizationOverloads

A two layer event, warning and critical, with a single resolution.

```
nShieldMonitor was load = (cmdCount - replyCount) * 100 / nshield.maxQ
load = jobs in progress / maximum queue length.
```

The minimum and maximum values values are percentages, from 0 to 150.

DeviceNShieldUtilizationPeakEvent

This alert has duration.

Valid parameters	min < x or max > y, for, over
OpenMetrics used	nshield_queue_in_progress, nshield_queue_length_limit
Basic Prometheus query	
Other data returned	esn, vcm
Summary returned	Amount of Active HSM Jobs exceed {max}% of queue length. Amount of Active HSM Jobs is below {min}% of queue length.

nShield Monitor alerts	DeviceNShieldUtilizationOverloads
	DeviceNShieldUtilizationPeakEvent

12.8. HSM Objects Count

This alert has a duration. It is a two layer event, warning and critical, with a single resolution.

Valid parameters	min < x or max > y, for, over
OpenMetrics used	nshield_objects_stored_total, nshield_objects_destroyed_total
Basic Prometheus query	nshield_objects_stored_total - nshield_objects_destroyed_total > max
Other data returned	esn, vcm
Summary returned	HSM Objects Count exceeds {max}. HSM Objects Count is below {min}.
nShield Monitor alert	DeviceNShieldHigHObjectCount

12.9. Host Hardserver

Valid parameters	for
OpenMetrics used	nshield_hardserver_liveness
Basic Prometheus query	nshield_hardserver_liveness == 0
Other data returned	esn, vcm
Summary returned	HSM is not responsive

nShield Monitor alert	ClientHostHardserverFailure
------------------------------	-----------------------------

12.10. HSM Liveness

KeySafe5 is not communicating with the HSM.

Valid parameters	for
Default alert (unlicensed)	Y (for 5 minutes)
OpenMetrics used	nshield_hsm_liveness
Basic Prometheus query	nshield_hsm_liveness == 0
Other data returned	esn, vcm
Summary returned	HSM is not responsive
nShield Monitor alert	DeviceConnStatus

12.11. Host Liveness

KeySafe 5 is not communicating with the host.

Valid parameters	for
Default alert (unlicensed)	Y (for 5 minutes)
OpenMetrics used	nshield_host_liveness
Basic Prometheus query	nshield_host_liveness == 0
Other data returned	host

Summary returned	Host is not responsive
nShield Monitor alert	KeySafe5 is not communicating with the host

12.12. Licence Expiry

Valid parameters	min < x
Default alert (unlicensed)	Y (min < 0) Y (min < 30 days)
OpenMetrics used	keysafe5_licence_expiry
Basic Prometheus query	keysafe5_licence_expiry - current_time < min
Other data returned	licence
Summary returned	Licence {licence} will expire in less than {min} days. Licence {licence} will expire in less than 1 day. Licence {licence} has expired.

12.13. HSM Client Licences Remaining

The actual number of licences remaining.

Valid parameters	min < y, for, over
OpenMetrics used	nshield_current_crypto_clients nshield_current_crypto_clients_limit
Basic Prometheus query	nshield_current_crypto_clients_limit - nshield_current_crypto_clients < min
Other data returned	esn

Summary returned	Fewer than {min} client licences remaining. No HSM client licences remaining.
-------------------------	--

12.14. Certificate Expiry

Valid parameters	min < x
Default alert (unlicensed)	Y (min < 0) Y (min < 30 days)
OpenMetrics used	keysafe5_certificate_expiry
Basic Prometheus query	keysafe5_certificate_expiry{type="agent"} < min keysafe5_certificate_expiry{type="central"} < min keysafe5_certificate_expiry{type="ca", agent=""} < min keysafe5_certificate_expiry{type="ca", agent=~".+"} < min
Other data returned	type, agent ¹
Summary returned	<cert> will expire in less than {min} days. <cert> will expire in less than 1 day. <cert> has expired.

1. **type** and **agent** values determine which of the following variations is used for the <cert> portion of the summary:
 - Central platform should have type "central" or type "ca" (System certificate, System CA certificate)
 - Agents will have type "agent" or type "ca" but will have an agent id (Agent <agent id> certificate, Agent <agent id> CA certificate)