

Controlling the Collection of Interfaces, Adjustments and Overrides

NMIS comes with sensible defaults for collecting (interface) data, but naturally no configuration can work perfectly for everybody.

This document briefly describes how you can control what interfaces and related data NMIS should collect.

- [Why is Interface X not being collected?](#)
- [Overriding NMIS' choices for one node](#)
- [Adjusting the node's model](#)
- [Global Overrides](#)
- [Interfaces which have been down for a long time](#)

Why is Interface X not being collected?

If you check the "interfaces" tab for a node (note: *not* the "active interfaces"), you'll see a panel similar to the following screenshot:

Name	Description	Admin Status	Oper Status	Intf. Avail.	Util. 6hrs	Bandwidth	Physical Address	Last Change	Collect on	
GigabitEthernet1/0/1	Odin	up	up	100.00	0.00	1 Gbps	00:0e:38:67:1f:81	164 days, 21:47:57	true	
GigabitEthernet1/0/2	Odin HP Remote Access Card	up	up	100.00	0.00	100 Mbps	00:0e:38:67:1f:82	164 days, 21:47:57	true	
GigabitEthernet1/0/3	Sif	up	up	100.00	0.02	1 Gbps	00:0e:38:67:1f:83	70 days, 17:24:58	true	
GigabitEthernet1/0/4	Magni	up	up	100.00	0.00	1 Gbps	00:0e:38:67:1f:84	164 days, 21:47:54	true	
GigabitEthernet1/0/5	Eris	up	up	100.00	0.00	1 Gbps	00:0e:38:67:1f:85	164 days, 21:47:51	true	
GigabitEthernet1/0/6		up	down			10 Mbps	00:0e:38:67:1f:86	164 days, 21:48:00	false	
GigabitEthernet1/0/7	PA200	up	down	0.00	0.00	10 Mbps	00:0e:38:67:1f:87	164 days, 21:48:00	true	

The most relevant column is the "Collect on" to the right: only interfaces with collect set to `true` will be handled by NMIS. NMIS will collect or not collect an interface for a variety of reasons, most of which are configurable. Let's click on the Interface link for GigabitEthernet1/0/6:

Admin Status	up
Oper Status	down
Name	GigabitEthernet1/0/6
Type	ethernetCsmacd
Physical Address	00:0e:38:67:1f:86
Description	
Bandwidth	10 Mbps
Last Change	164 days, 21:48:00
Collect on	false
Reason	Not Collecting: no Description (ifAlias)

The "Reason" describes why NMIS has decided to not collect this interface; in this particular case the model for the device has specified that interfaces without Description are to be ignored.

Depending on your model and version of NMIS, the labels on the Interface Details page may be slightly different from the example:

- The "Name" generally refers to the SNMP properties `ifDescr` or `ifName` (availability of the latter depends on the hardware in question)
- "Description" almost universally refers to the SNMP property `ifAlias`.
- "Type" refers to `ifType`.
- "Hardware" (if present) refers to `ifDescr`. This is for cases where the interface "Name" is dynamically chosen from the 'best available' source (e.g. `ifName` or `ifDescr` if `ifName` isn't supported).

NMIS 8.6 and newer ship with models that have amended labels which include the relevant source, for example "Hardware (ifDescr)" and "Type (ifType)" for greater clarity.

Overriding NMIS' choices for one node

If you only want to override NMIS for one or two interfaces on a single node then you might use the Node Configuration system to adjust the "collect" setting to your liking. This is documented on the [Node Configuration](#) page.

Adjusting the node's model

It's more likely that you'll want the same collect/nocollect choices for all nodes of a particular type, i.e. sharing the same model. NMIS offers very flexible mechanisms to express this policy: a model can contain a section called "nocollect" containing a number of criteria which will be checked when NMIS analyzes the node (during a `type=update` operation).

Here is an example:

```
'collect' => {
  'Description' => '(CNOC|Collection)',
  'ifDescr' => '(FastEthernet|ATM)',
},
'nocollect' => {
  'ifOperStatus' => 'null',
  'ifDescr' => 'sublayer|null|controller',
  'Description' => '^#',
  'noDescription' => 'true',
  'ifType' => 'other|softwareloopback|propVirtual|slip'
}
```

The conditions are interpreted as regular expressions (except for `noDescription`). If any one of those criteria matches, then the interface data is NOT collected. Condition entries that are not present in the model are ignored. In addition to the regular expression conditions, there is the `noDescription true/false` setting which controls whether interface without textual description should be ignored or not.

In the example above, the `ifDescr` condition will exclude interfaces that match sublayer for example; the `Description` condition deselects interfaces whose textual description starts with "#", `ifType` ensures that only physical interfaces are considered, and `ifOperstatus` deselects interfaces with an unknown operational status.

To edit any of these with the NMIS GUI, select the "System" menu, sub-menu "System Configuration" and then "NMIS Models". Select either one of the nodes in question, or the model, and then scroll down to the section "interface". Near the top of the various settings you'll see the "nocollect" section.

Please note that changing any entries in the "nocollect" section will only begin to affect nodes when the next `type=update` operation is performed (which by default is once every 24hrs).

Global Overrides

The per-model criteria also interact with (and can be overridden by) certain global settings. Here are the relevant defaults as shipped with NMIS' Config. nmis:

```
'globals' => {
  'global_collect_Description' => '(CNOC|Collection)',
  'global_collect_ifDescr' => '',
  'global_nocollect_ifOperStatus' => '',
  'global_nocollect_ifDescr' => '',
  'global_nocollect_Description' => '',
  'global_nocollect_noDescription' => '',
  'global_nocollect_ifType' => '',
  # ...plus other unrelated global_XYZ settings
},
```

Like before, `global_nocollect_noDescription` is a true/false setting, whereas the others are all interpreted as regular expressions (if not missing or empty). Keep in mind that these are regular expressions, therefore if we wanted to not collect ifType's `propVirtual` & `l2vlan` the syntax would look like this:

```
'global_nocollect_ifType' => 'l2vlan|propVirtual',
```

Please note that the checkable properties aren't necessarily labelled the same on the Interface Details page (e.g. setting `global_collect_ifDescr` compares the `ifDescr` value against the given regular expression, but the `ifDescr` data is likely labelled "Name" in the GUI).

The sequence of checks is as follows:

1. The default for collection is true.
2. `global_collect_Description` and `global_collect_ifDescr` are checked first. If they're non-empty, and the interface matches either, it's collected and no further criteria are applied.
3. If an interface has no administrative status, or if it is 'down' or 'testing', then the interface is not collected and we stop checking criteria.
4. Next, the other `global_nocollect_*` criteria are checked. If we find one that is non-empty and matches the interface, the interface is NOT collected and no further criteria are checked.
5. After that, the model-level collection criteria are applied. The first matching `nocollect` rule causes the interface to NOT be collected and terminates the sequence of checks.

(Should you ever be interested in the gory details, check out `bin/nmis.pl` and look for `sub getIntfInfo`.)

To adjust any of these settings from the NMIS GUI, go to the "System" menu, submenu "System Configuration" and then "NMIS Configuration"; select section "globals".

Please note that changes to any of these globals also don't become effective until the next `type=update` operation is performed.

Interfaces which have been down for a long time

NMIS includes a feature which will disable collecting of interfaces which are down for a period of time, by default, this is 30 days, this is because if down for 1 month, it is likely the interface is not being used anymore. This feature can be disabled by setting the value in the `Config.nmis` for `global_nocollect_interface_down_days` to a very high number, like 5000 or more. Or the interface can be set in Node Config to always collect.

This feature is based in the *ifLastChange* MIB for each interface. By definition, *ifLastChange* is "The value of *sysUpTime* at the time the interface entered its current operational state. If the current state was entered prior to the last re-initialization of the local network management subsystem, then this object contains a zero value.". Some Fortigate devices have been identified as not updating the *ifLastChange* when *ifOperStatus* is changed. That will set the interfaces with operational status *down* as *no collect* after *sysUpTime* days is greater than **`global_nocollect_interface_down_days`** (30 days by default). In this case, setting **`global_nocollect_interface_down_days`** to a really high value is recommended to disable the feature, to prevent the interfaces not being collecting were they should be.