SNMP Tuning

SNMP is a fairly complex protocol, and the fact that it primarily operates over UDP does not exactly help matters. As a consequence, there are a number of potential problems that affect NMIS' ability to collect information from SNMP agents efficiently and quickly.

SNMP Global changes using CLI

Go to **/usr/local/nmis9/Conf/**

vim Config.nmis

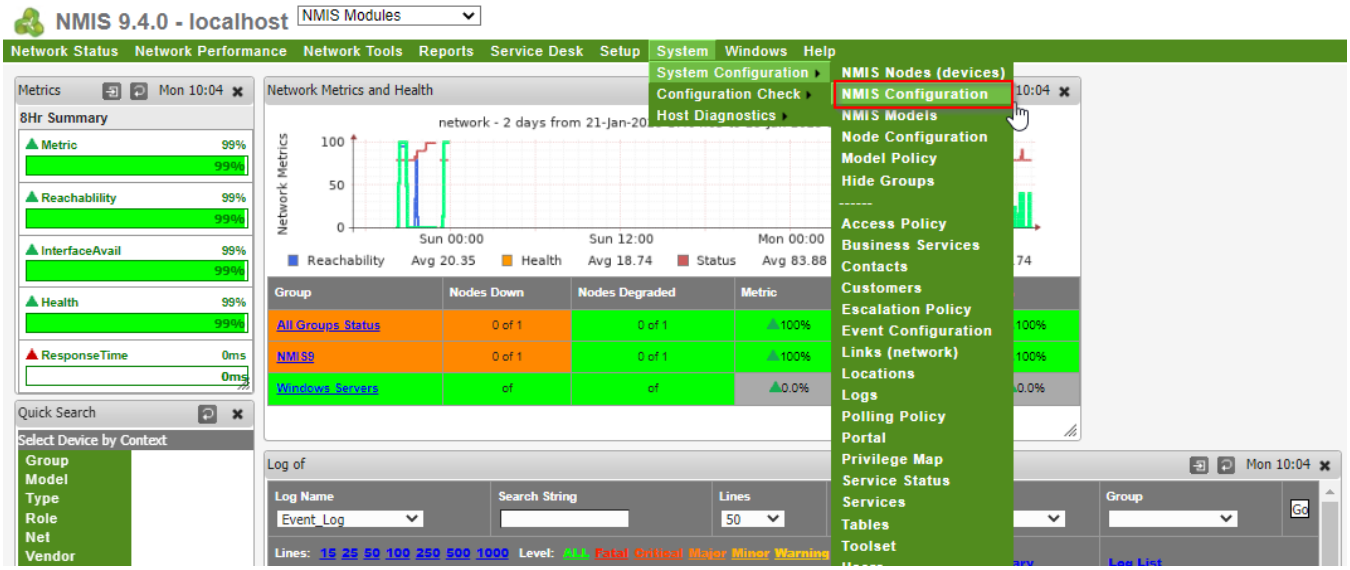
then search for:

```
      'default' => [
        'Major',
        'Minor'
      ],
      'distribution' => [
        'Major',
        'Minor'
      ]
    },
    'snmp_max_msg_size' => 1472,
    'snmp_retries' => 1,
    'snmp_stop_polling_on_error' => 'true',
    'snmp_timeout' => 5,
    'stateless_event_dampening' => 900,
    'threshold_period-default' => '-15 minutes',
    'threshold_period-health' => '-4 hours',
    'threshold_period-interface' => '-15 minutes',
    'threshold_period-pkts' => '-15 minutes',
    'threshold_period-pkts_hc' => '-15 minutes',
```

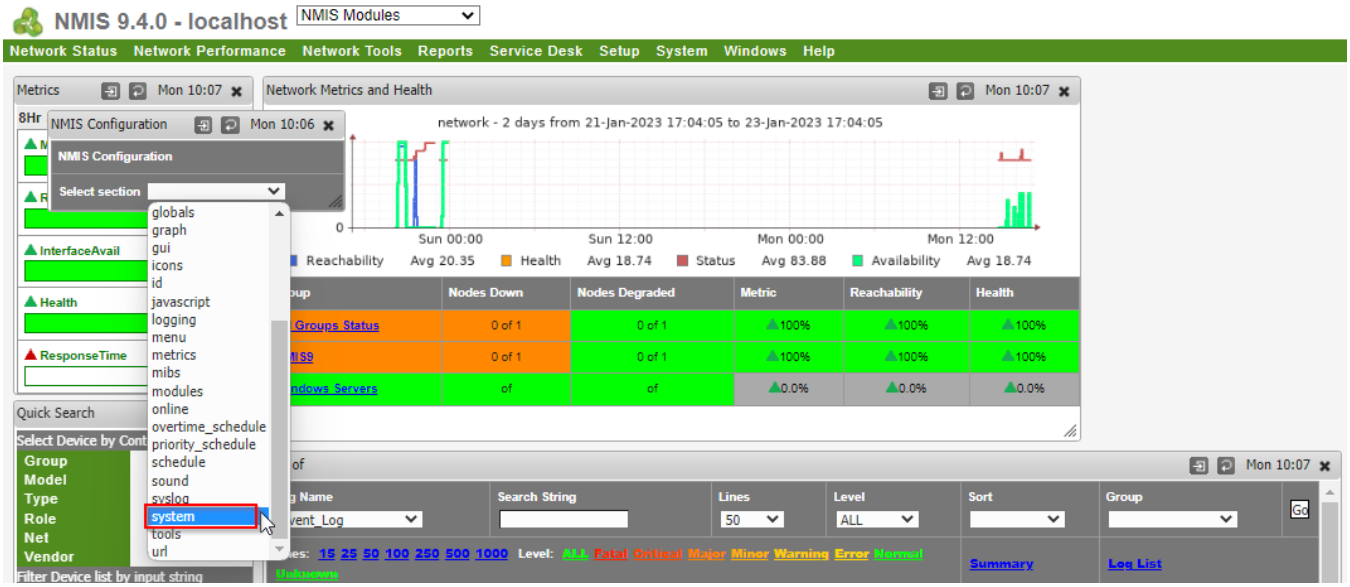
OBSERVATION: When you do any change on the Config.nmis it will apply to all SNMP devices.

SNMP Global changes using WebUI

On NMIS goes to **the System tab ==> System Configuration ==> NMIS Configuration**



On the NMIS Configuration, select section ==> **system**.



After you selected System, it will show all settings.

	snmp_timeout	5	edit
	snmp_retries	1	edit
	snmp_stop_polling_on_error	true	edit
	auto_expand_more_graphs	true	edit delete
	default_authkey		edit delete
	default_authpassword		edit delete
	default_communityRO	public	edit delete
	default_privkey		edit delete
	default_privpassword		edit delete
	default_username		edit delete
	demote_faulty_nodes	true	edit delete
	display_status_summary	true	edit delete
	fastping_interval	1	edit delete
	fastping_target_interval	1	edit delete
	file_size_warning	100000000	edit delete
	hide_groups		edit delete
	interface_availability_value_when_down	U	edit delete
	interface_max_number	5000	edit delete
	interface_util_label	Util. 6hrs	edit delete
	interface_util_period	-6 hours	edit delete
	json_node_fields	uuid,supportGroup,cmdbType	edit delete
	keep_event_history	false	edit delete
	log_max_table_lines	25000	edit delete
	log_node_configuration_events	false	edit delete
	max_child_runtime		edit delete
	model_health_sections	cpu_cpm,entityMib,diskIOTable,ds3Errors,SONETErrors	edit delete
	network_health_view	Group	edit delete
	network_summary_maxgroups	30	edit delete
	network_viewNode_field_list	nodestatus,outage,sysName,host_addr,host_addr_ba	edit delete
	nmis_executable	(/(bin|admin|installer_hooks|conf-default/scripts|conf/scripts)/[a-zA-Z0-9_\.-]+|\.pl|\.sh|/installer)$	edit delete

	model_health_sections	cpu_cpm,entityMib,diskIOTable,ds3Errors,SONETErrors	edit delete
	network_health_view	Group	edit delete
	network_summary_maxgroups	30	edit delete
	network_viewNode_field_list	nodestatus,outage,sysName,host_addr,host_addr_backupro	edit delete
	nmis_executable	(/(bin|admin|installer_hooks|conf-default/scripts|conf/scripts)/[a-zA-Z0-9_\.-]+|\.pl|\.sh|/installer)$	edit delete
	nmis_host_protocol	http	edit delete
	node_configuration_events	Node Configuration Change, Node Reset	edit delete
	node_status_uses_status_summary	true	edit delete
	node_summary_field_list	host,uuid,customer,businessService,serviceStatus,nodestatus,	edit delete
	non_stateful_events	Node Configuration Change, Node Configuration Change Detected, Node Reset, NMIS runtime exceeded	edit delete
	os_execperm	0770	edit delete
	overall_node_status_coarse	false	edit delete
	overall_node_status_level	Critical	edit delete
	plugins_enabled	true	edit delete
	polling_interval_factor	0.95	edit delete
	postpone_clashing_schedule	30	edit delete
	selftest_cron_name	(^|/)crond?$	edit delete
	selftest_max_collect_age	900	edit delete
	selftest_max_swap	50	edit delete
	selftest_max_system_cpu	60	edit delete
	selftest_max_system_iowait	20	edit delete
	selftest_max_update_age	604800	edit delete
	selftest_min_diskfree_mb	25	edit delete
	selftest_min_diskfree_percent	10	edit delete
	server_admin	root@localhost	edit delete
	server_role		edit delete
	snmp_max_msg_size	1472	edit delete
	stateless_event_dampening	900	edit delete

	threshold_period-default	-15 minutes	edit delete
	threshold_period-health	-4 hours	edit delete
	threshold_period-interface	-15 minutes	edit delete
	threshold_period-pkts	-15 minutes	edit delete
	threshold_period-pkts_hc	-15 minutes	edit delete
	update_interval_factor	0.95	edit delete
	upnotify_stateful_events	down|proactive|alert	edit delete

SNMP Changes for a Specific Node

Select the specific node then "Edit Node".

NMIS 9.4.0 - localhost NMIS Modules

Network Status | Network Performance | Network Tools | Reports | Service Desk | Setup | System | Windows | Help

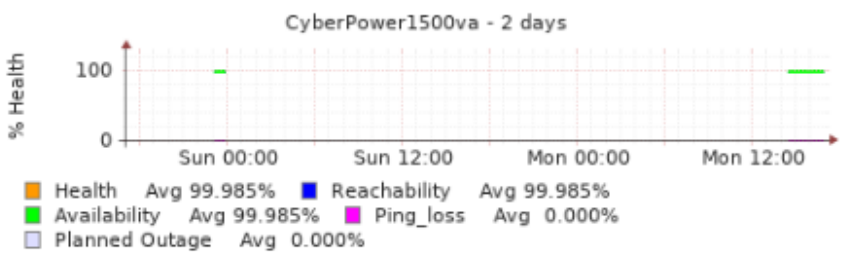
CyberPower1500va Mon 9:59

Node CyberPower1500va | events | outage | Diagnostic | contact | location

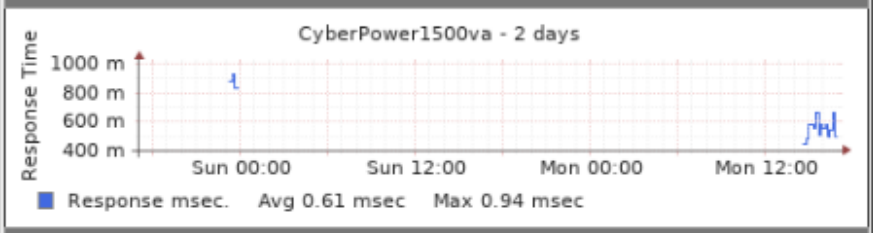
Node Details - CyberPower1500va - Edit Node - Node Configuration - Node Context - SSH to Node

nodestatus	reachable
System Name	RMCARD205
IP Address	192.168.0.107
Backup IP Address	
Group	NMIS9
Customer	Opmantek
Location	Cloud
Business Service	
Service Status	Development
Notes	
Type	generic
Model	Default
Polling Policy	default
Uptime	5 days, 5:04:44
SNMP Location	Server Room
Contact	Administrator
Description	UPS SNMP Card
Interfaces	2
Last Ping	23-Jan-2023 16:58:12
Last Collect	21-Jan-2023 23:54:06
Last Update	23-Jan-2023 14:47:36
Vendor	Cyber Power System Inc.
Object Name	enterprises.3808.1.1.1
Role	core
Net	wan

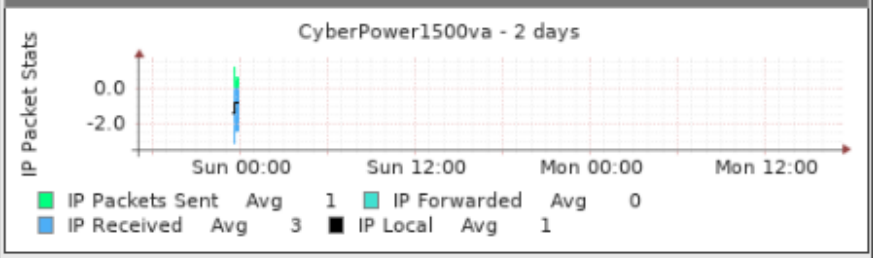
Overall Reachability, Availability and Health

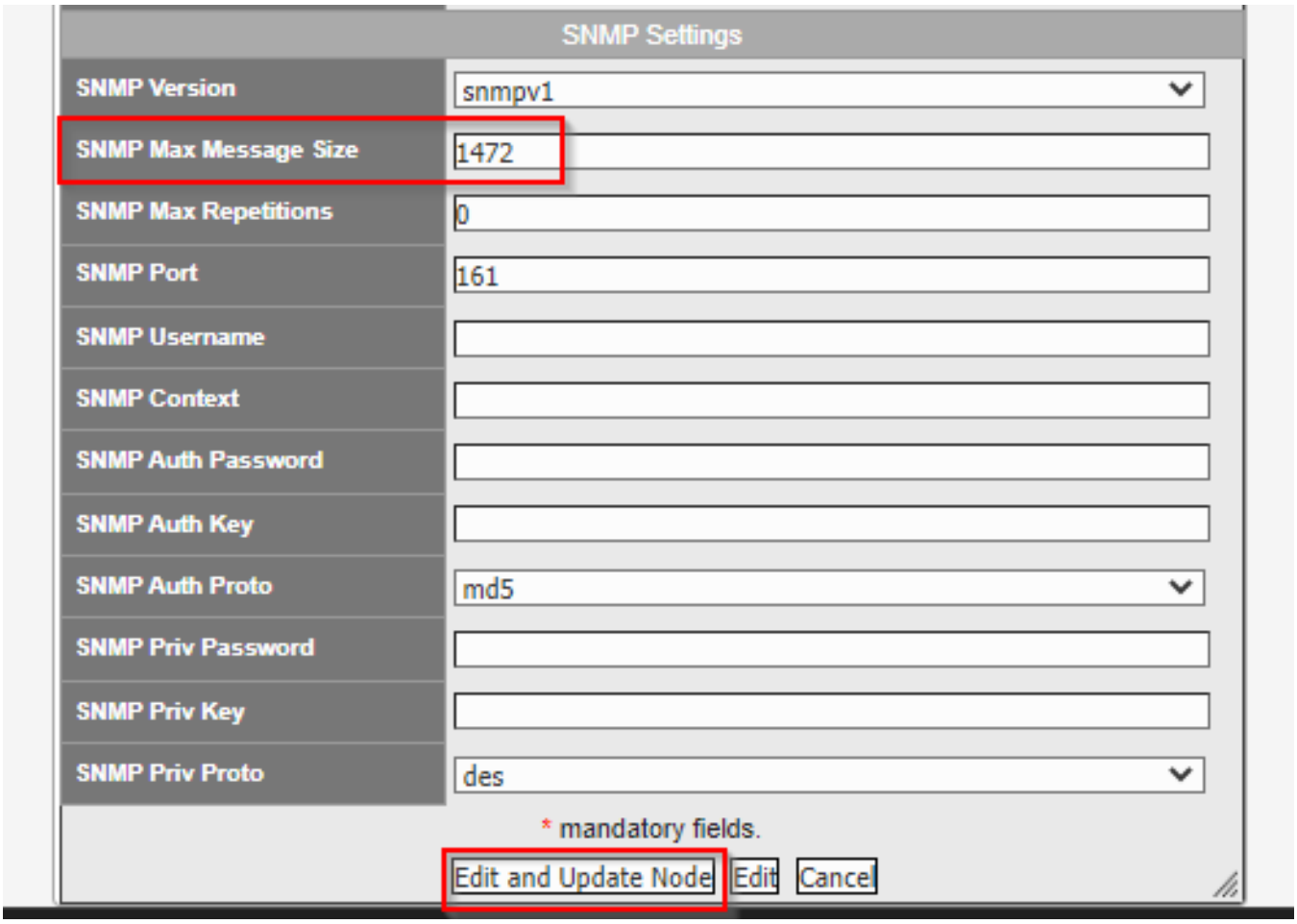


Response Time in milliseconds



IP Utilisation





snmp_timeout and snmp_retries

By default, NMIS has a 5 second SNMP timeout and will retry once before it considers SNMP to have failed. The default settings work in 99% of circumstances, some devices and/or networks require increased timeout or retries to work better, so these settings can be increased, however, it is important to remember that when SNMP is not responding the polling process will now have to wait for the multiple of the timeout and retries, so by default 5 seconds. If the retries were set to 3 then 5 seconds and 3 retries would be 15 seconds before NMIS considers that SNMP is down.

For servers with many nodes, it is not recommended for multiple of timeouts and retries to exceed 20 seconds.

snmp_max_msg_size

The primary tunable NMIS configuration setting for SNMP is `snmp_max_msg_size`, which controls how large a single SNMP packet may be.

This can be set as a system-wide default (in the System menu, under System Configuration), or as a per-host setting (in the Edit Node menu, under Advanced Options).

The default for `snmp_max_msg_size` is 1472 bytes, just below the 1500 byte packet limit for normal Ethernets. In LAN-only scenarios it is possible to increase this past 1500 bytes: this causes IP fragments and packet reassembly, but unless your LAN is saturated and starving for bandwidth fragmentation is not a problem. The benefit of a larger SNMP packet would be that the data to be collected fits into fewer packets.

max_repetitions

This option was added in NMIS 8.5G. It controls how many SNMP PDUs will be packaged into a single SNMP packet. The `max_repetitions` setting is named a bit oddly - that comes from the SNMP module that NMIS uses: Net::SNMP calls it "`-maxrepetitions`".

This option can **only** be set for specific hosts and is not available for SNMP version 1.

Its primary purpose is to overrule Net::SNMP's heuristic for maximizing the efficiency of bulk transfers: the goal is to fit the maximum number of PDUs into each packet, which of course depends on the size of the PDUs (and their sizes are unknown until the operation is attempted). Like any other heuristic, this one can fail under certain circumstances: If large SNMP tables are collected then it may be necessary to reduce this setting to 10-20 (when used with the default packet size). We have observed this problem in a small number of situations, for example when collecting virtual machine info from VMware ESXi hosts - the strings contained in these tables are *really* long.

If you observe SNMP error messages in the logs which look similar to *"SNMP ERROR (X) (Y) The message size exceeded the buffer maxMsgSize of N"*, then you should set a lower `max_repetitions` value (or increase the `snmp_max_msg_size` if you're operating in a LAN-only scenario). Otherwise, a value of 40-50 minimizes the number of SNMP packets and thus speeds up collection. Not setting this option at all leaves it to the Net::SNMP module to guess a suitable value.

There is one special setup for `max_repetitions`: if it is set to 0 it will behave with the default of the NET-SNMP Perl library, which appears to be 25, or if set to 1 the efficient bulk transfer is disabled and a slower but more robust transfer mechanism is employed.

The setting max_repetitions should be added to a node entry in the Nodes.nmis file and is an option in the NMIS8 GUI when editing nodes.

New in NMIS 8.6: Automatic `max_repetitions` adjustment

As outlined in the NMIS 8 Release Notes, from version 8.6.0 onwards NMIS will dynamically reduce the `max_repetitions` parameter if necessary.

If a "message size exceeded" error is encountered, the issue is logged and the current `max_repetitions` value is reduced by 25% before the request is retried. If that retry works, the updated value is used for the SNMP session lifetime, i.e. the remainder of this node's collect of update operations. Up to four reduce-and-retry iterations are performed before NMIS gives up on the request and returns an error.

If you have not set a `max_repetitions` value, the first retry will use the value 20.

Whenever such an automatic adjustment is attempted, NMIS logs a warning message similar to this example:
`"WARNING (servername) SNMP get_table failed with message size exceeded, retrying with maxrepetitions reduced to 36"`