OIDs take the form of a set of numbers or strings (again, you can use these interchangeably). An example is `1.3.6.1.4.868.2.4.1.2.1.1.1.3.3562.3`.

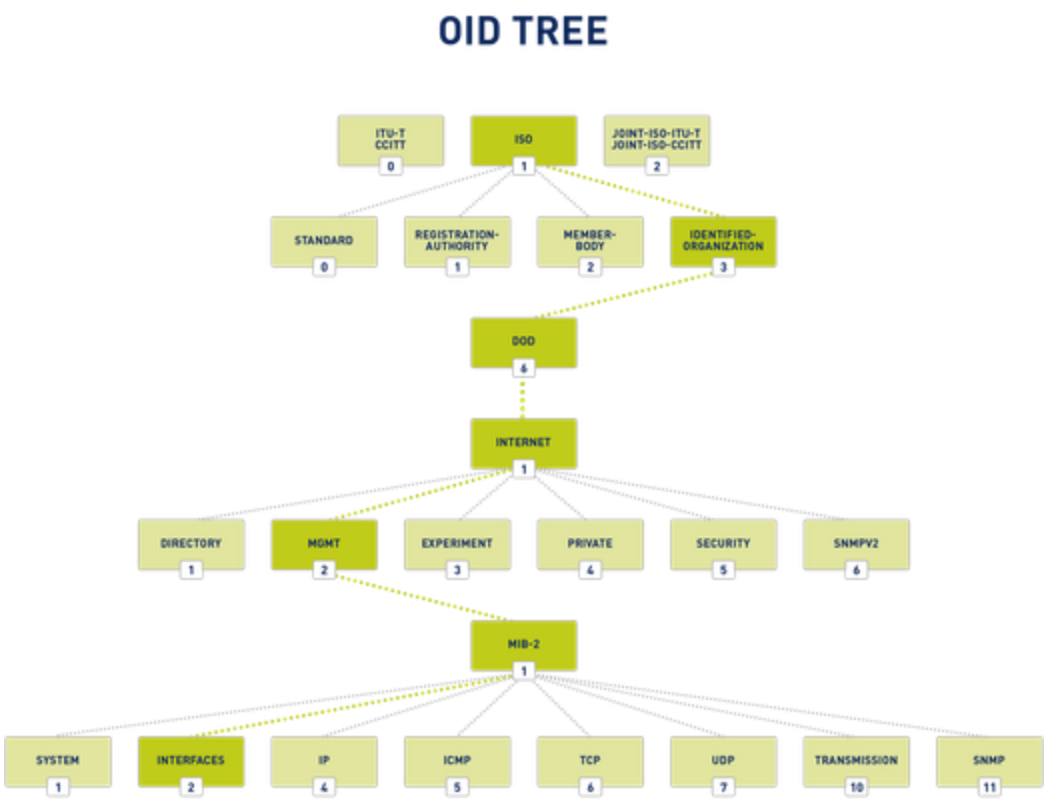
Written with strings, that OID would translate to:

```
iso.org.dod.internet.private.transition.products.chassis.card.
slotCps.
cpsSlotSummary.cpsModuleTable.cpsModuleEntry.cpsModuleModel.35
62.3.
```

For example, if an OID starts with 1.3.6.1.4.1.9, it applies to a Cisco device. Other vendors have their own OID specifications. (Wireshark, the open source network scanner, offers a handy OID lookup tool.) The standard OID prefix, which can be used for almost any device that supports SNMP, is 1.3.6.1.2.

1.3.6.1.2.1.2 OID

- 1 – International Organization for Standardization (ISO)
- 3 – Identified organizations according to ISO/IEC 6523-2
- 6 – US Department of Defense (DOD)
- 1 – Internet protocol
- 2 – Internet Engineering Task Force (IETF) management
- 1 – MIB-2
- 2 – Interfaces
-



OID you can also read the manufacturers as listed by the Internet Assigned Numbers Authority (IANA), for example

- 1.3.6.1.4.1.9 Cisco MIBs
- 1.3.6.1.4.1.311 Microsoft MIBs
- 1.3.6.1.4.1.2636 Juniper MIBs
- 1.3.6.1.4.1.8117 North American Association of Food Equipment Manufacturers MIBs

What is SNMP? NMS, MIBs, OIDs, Traps & Agents

SNMP, which stands for Simple Network Management Protocol, is a communication protocol that lets you monitor managed network devices including Routers, Switches, Servers, Printers and other devices that are IP enabled all through a single management system/software.

If the networked device is SNMP capable, you can enable and configure it to start collecting information and monitor as many network devices as you want from a single point.

What does SNMP do?

- Monitor inbound and outbound Traffic flowing through the device
- Early Detection of faults within network devices along with Alerts/Notifications
- Analyzing data collected from devices over long periods of time to identify bottlenecks and performance issues
- Ability to remotely configure compatible devices
- Access and Control devices remotely that are connected via SNMP

The Basics

There are several components that allow SNMP to work correctly, inlcuding:

- SNMP Manager (Network Management System)
- SNMP Agents
- SNMP Port
- Managed Device (includes Servers, Switches, Routers, and more.)
- MIB (Management Information Base, also know as Management Information Database)
- OID (Object Identifier)
- Traps
- Versions

Manager (NMS)

The Manager component is simply a piece of software that is installed on a machine (which when combined, is called the Network Management System) that polls devices on your network how ever often you specify for information.

The Manager has the correct credentials to access information stored by Agents and then compiles them in a readable format for the Network Engineer or Administrator to monitor or diagnose for problems or bottlenecks.

Some NMS software suites are more complex than others, allowing you to configure Email or SMS messages to alert you of malfunctioning devices on your network, while others simply poll devices for more basic information.

Agents

SNMP Agent is a piece of software that is bundled with the network device (router, switch, server, wifi, etc) that, when enabled and configured, does all the Heavy work for the Manager, by compiling and storing all the data from its given device into a database (MIB).

This database is properly structured to allow the Manager software to easily poll information and even send information to the Manager if an error has occured.

What Port Numbers does SNMP Use?

The manager Software polls the agents at regular intervals over **Port UDP 161**.

SNMP Traps allows an Agent to send system and device information to the manager over **Port UDP 162**.

Although UDP is the common protocol used to by SNMP, TCP can also be used as well.

Managed Network Devices

Managed Network Devices, including Routers, Switches, Wifi, Servers (Windows and others), Desktop PC's, Laptops, Printers, UPS's, etc, have agent software built into them that needed to be either enabled and configured or simply configured properly in order to be polled by the NMS.

MIB

In short, MIB files are the set of questions that a SNMP Manager can ask the agent.

Agent collects these data locally and stores it, as defined in the MIB. So, the SNMP Manager should be aware of these standard and private questions for every type of agent.

Agents, as explained above, maintains a organized database of its devices parameters, settings, and more.

The NMS (Network Management system) polls/requests the Agent of a given device, which then shares its organized information from the database its made with the NMS, which then further translates it into alerts, reports, graphs and more.

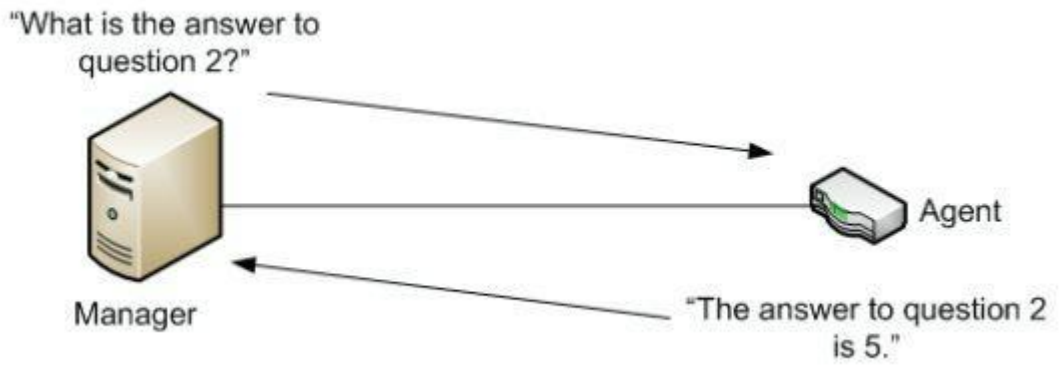
The database that the Agent shares between the Agent is called the **Management Information Base**, or **MIB**.

MIB's contain a set of Values, both statistical and contol, that are defined by the network device. On many occassions, extensions of standard values are defined using a Private MIB by different vendors of networked devices.

To simplify MIB's, think of it like this: MIB files are the set of Questions that an Manager are allowed to ask the agent.

The Agent just collects these questions and stores them locally and serves them to the NMS when requested.

A Simplified Example of how MIB's work: The NMS will ask the network Device a Question, in this case, what is the Answer to Question 2?



The managed network Devices' Agent then Responds with the Answer to the Question 2. To break this down even further, lets construct another example.

Say we want to know the System Uptime of a Device.

The NMS will send a Request to the Agent requesting the the System Uptime – the request is sent as a number with the MIB and the Object of Interest, along with something called the*Instance*.

OID = 1.3.6.1.2.1.1.3.0

Breakdown of OID Number

MIB	**Object of Interest**	**Instance**
1.3.6.1.2.1.1	3	0
MIB	SysUptime Object	Instance

The first 2 parts of the number sent to the Agent (MIB and Object of Interest, which in this case is System Uptime) is called the **Object Identifier or OID**.

As mentioned above, MIB's are Standard values that a Network Management System already knows about and can poll/request Network Devices about for information.

OID

OID, Object Identifier is a simply a number made up by the MIB, Object of Interest and the Instance. Each identifier is Unique to the device, and when queried will provide information on what has OID has been requested.

There are 2 types of OID's:

- Scalar
- Tabular

Scalar is a Single object Instance – for example, a Device's vendor name. There can only be a Single vendor name, so this would be a scalar OID.

Tabular on the other hand, can have multiple results for its OID – for example, a Quad Core processor would result in 4 different CPU values.

Traps

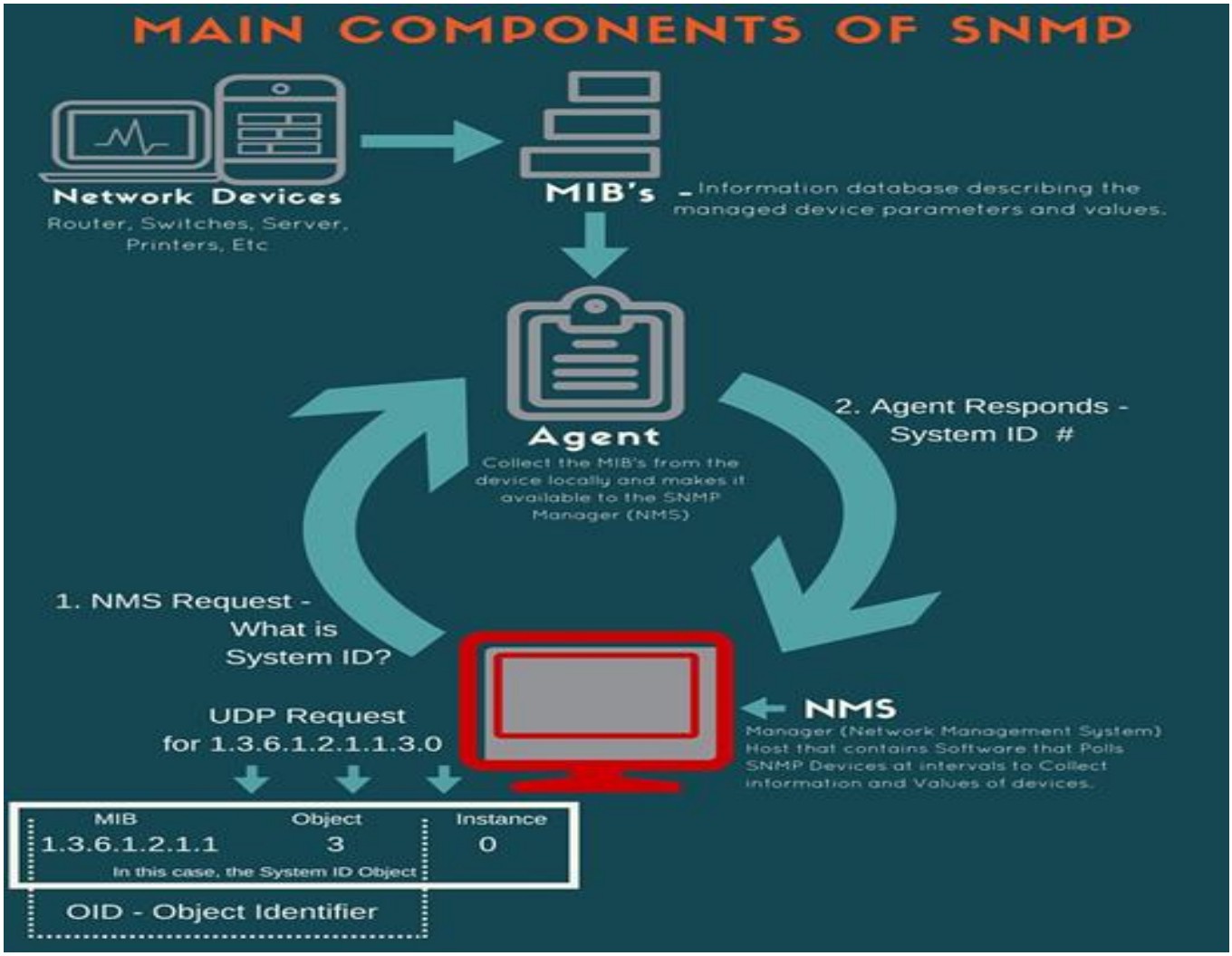
Traps are used when the Device needs to alert the Network Management software of an event without being polled.

Traps ensure that the NMS gets information if an certain event occurs on the device that needs to be recorded without being Polled by the NMS first.

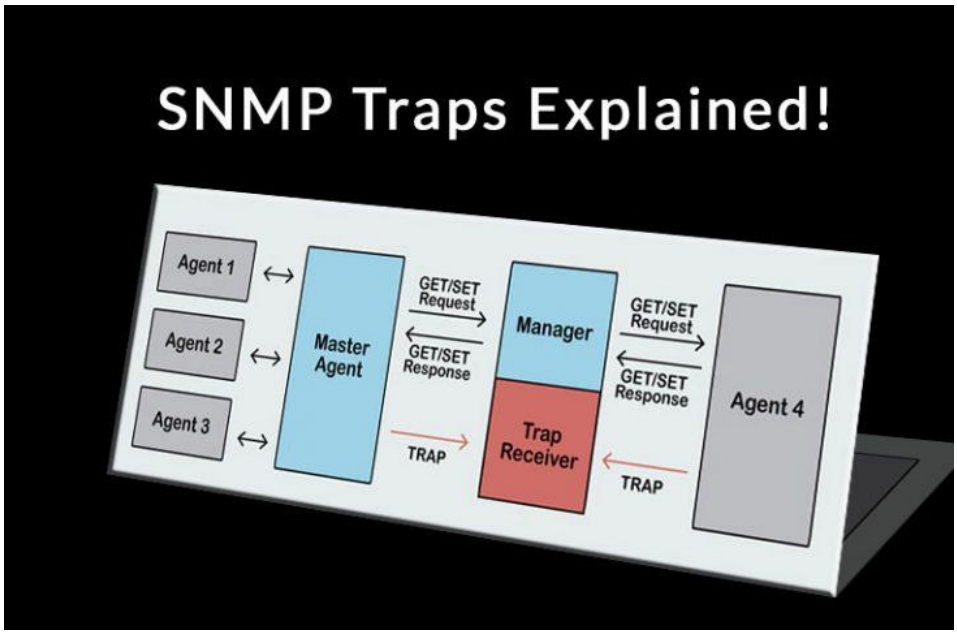
Managed network devices will have Trap MIBs with pre-defined conditions built into them.

Its crucial that the Network management system has these MIBs compiled into them to receive any traps sent by the given device/s.

MIBs are number that identify certain characteristics or values of a device, but if the Network Management system does not have a certain MIB that the network device Trap is sending, there is no way to interpret what the MIB is and will not record the event.


MAIN COMPONENTS OF SNMP
Network Devices
Router, Switches, Server, Printers, Etc
MIB's
- Information database describing the managed device parameters and values.
Agent
Collect the MIB's from the device locally and makes it available to the SNMP Manager (NMS)
2. Agent Responds - System ID #
1. NMS Request - What is System ID?
UDP Request for 1.3.6.1.2.1.1.3.0
NMS
Manager (Network Management System) Host that contains Software that Polls SNMP Devices at intervals to Collect information and Values of devices.
MIB
Object
Instance
1.3.6.1.2.1.1
3
0
In this case, the System ID Object
OID - Object Identifier

SNMP Traps

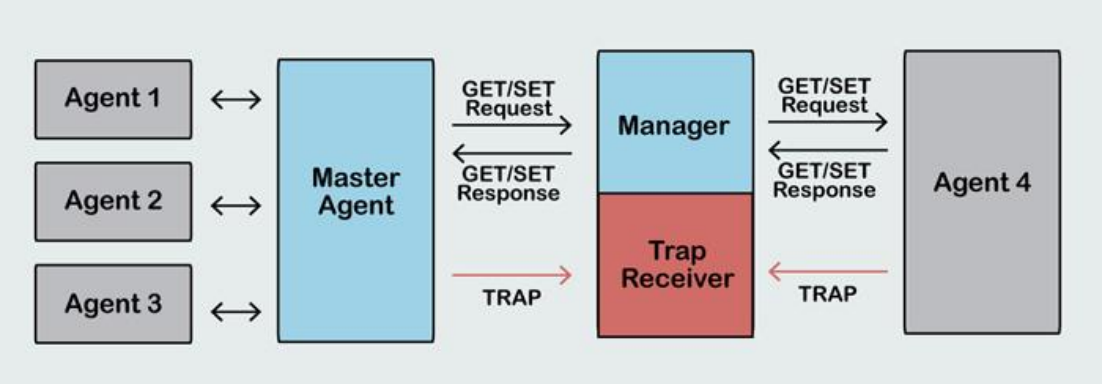


Simple Network Management Protocol (SNMP) is used by different devices (Routers, Switches, Printers, etc.) on the network to check each other's activity and communicate critical information.

Today, SNMP is one of the most widely accepted protocols for network monitoring, which enables many network devices to operate together.

SNMP relies on an architecture which consists of a manager and an agent. SNMP Managers can be any machine on the network that is running SNMP to collect and process information from the devices on either the LAN or WAN.

These network devices are Agents that can be Servers, Routers, Switches, Desktops, or any other Equipment.



SNMP messages are categorized into five basic types such as TRAP, GET, GET-NEXT, GET-RESPONSE, and SET.

SNMP manager and SNMP agent use these messages to communicate with each other.

Devices That Support SNMP Traps

There is one of the two device types most commonly used to issue SNMP traps. Newer devices alert the SNMP manager on their own to send the traps when an issue occurs.

The older devices, on the other hand, do not support SNMP, so the SNMP RTU is used to collect the alert information from different devices which converts them into SNMP traps and transmits them back to the SNMP manager.

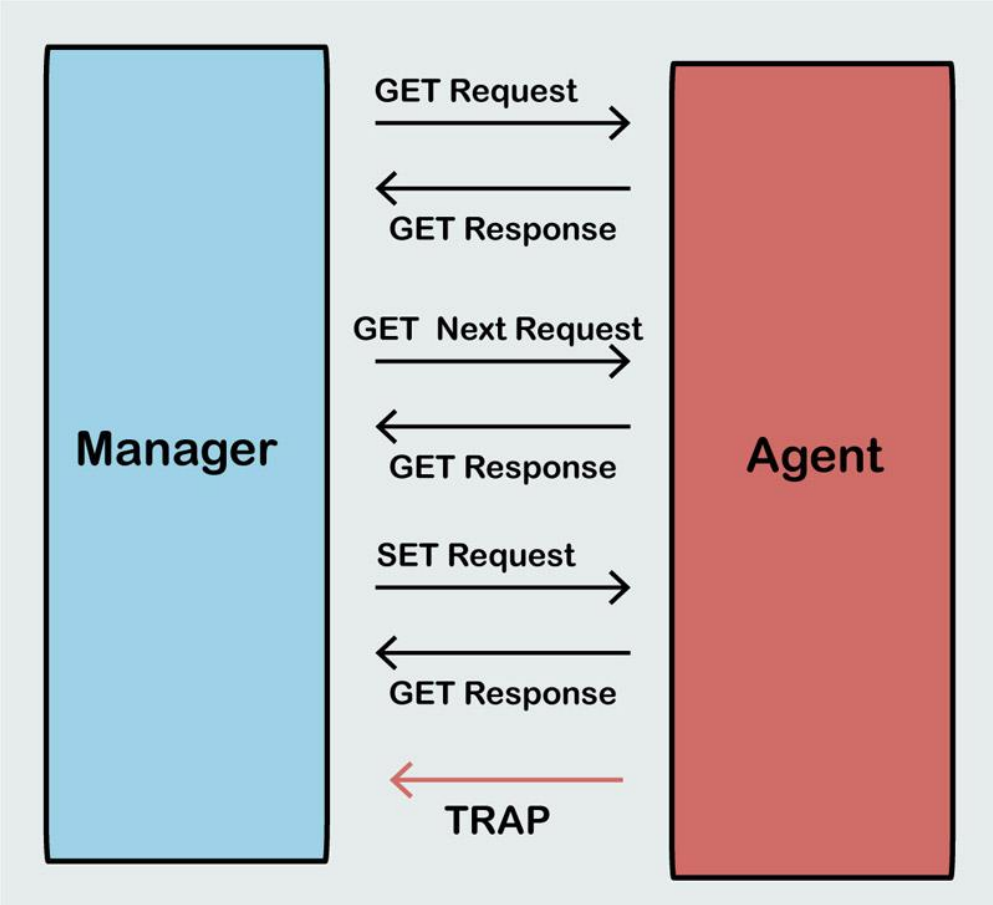
What are SNMP Traps?

SNMP Traps are the most frequently used alert messages sent from a remote SNMP-enabled device (an agent) to a central collector, the “SNMP manager.”

For instance, a Trap might report an event of overheating in a machine.

As mentioned earlier, the Trap messages are the main form of communication between an SNMP Agent and SNMP Manager. They are used to inform an SNMP manager when a significant event occurs at the Agent level.

What makes the Trap unique from other messages is that they are triggered instantaneously by an agent, rather than waiting for a status request from the SNMP Manager.



Types of SNMP Traps

There are two different methods used to encode alarm data in SNMP traps. The first one is to use what are known as “granular traps.” Granular traps each have a unique object identifier (OID) number so that SNMP managers can distinguish them from one another.

The SNMP manager getting the SNMP traps from the network devices or agents will store the OID in a translation file which is known as the Management Information Base or MIB.

Now, because the granular traps use unique numbers to support this method and all of the details are available in the MIB, no actual information about the alert needs to be contained within the SNMP trap.

So, the manager only needs OID to look up the information in MIB.

This approach prevents the SNMP traps from sending redundant information through the network, and they consume very little bandwidth.

In the second type, SNMP traps may be configured to contain information about the alerts as payloads. In this case, it's very usual for all SNMP traps sent from the device to use the same OID.

To understand these types of traps, the SNMP manager needs to analyze the data contained in each Trap.

Data is stored within an SNMP trap in a simple key-value pair configuration. Each pair is known as a "variable binding" containing extra information related to the Trap.

As an example, a single SNMP trap may have the variable bindings for "site name," "severity," and "alarm description."

Summary

SNMP trap is a popular mechanism used to manage and monitor devices' activities across a small or a global network.

Routing platforms are capable of generating a range of events that can be highly useful to the network administrators.

Furthermore, it's up to the operations team to select and configure the alerts for each event.

A proper SNMP deployment can:

- Help to detect data loss and to achieve reliable data transmission.
- Identify latency issues and packet loss
- Keep bandwidth usage below predefined service levels
- Avoiding network congestion and blackouts
- Reducing the cost and Time of Network Troubleshooting and a Lot More!

RMON Remote Network Monitoring

Remote monitoring allows authorized users to manage data that is integral to daily operations. Having streamlined control of your clients' networks lets them set up new connections, give authority and access to documents and maintain computer security.

Companies use SNMP software across local area networks (LANs) to monitor network devices that warrant attention or configuration vs. RMON software, which lets users control the flow of information in their networks and manage and monitor individual users across software platforms.

SNMP vs. RMON

Benefits of SNMP

Simple Network Management Protocol (SNMP) is a common protocol for managing a computer network. SNMP collects information from and configures network devices including servers, hubs, switches and routers over an Internet Protocol (IP) network.

SNMP allows a user to:

- Monitor network performance
- Audit network usage
- Detect network faults
- Detect inappropriate access
- Configure remote devices

SNMP is simple, as its name implies. It collects information that has a minimal impact on daily operations and continues to work despite other network errors. It is the first line of defense when retrieving information across a network.

Benefits of RMON

Remote Network Monitoring (RMON) is part of a management information base (MIB) that is an extension of an SNMP, allowing an administrator or a set of authorized personnel to monitor, analyze and troubleshoot network issues from a central site. RMON differs from SNMP in that it can set thresholds for alarming, can be supported through different types of software and goes more in depth with alerting and trapping generation. RMON collects groups of statistics that can be analyzed for long-term trends.

RMON collects nine kinds of information:

1. Packets sent
2. Bytes sent
3. Packets dropped
4. Statistics by host
5. Conversations between two sets of addresses
6. Events
7. User bandwidth
8. User traffic
9. Accessed Web sites

Features of RMON include:

- Manage and monitor servers and workstations
- Deliver remote support in real-time
- Offer asset/inventory tracking
- Patch management and client reporting
- Offer security features like antivirus and backup solutions
- Automated monitoring allows you to spot problems early and fix them
- Immediate contact with your customer base as individuals or in groups
- Offline monitoring
- Proactive monitoring
- Value added data

RMONs save you time and money without the hassle of extra setup and equipment, all from one workstation. The more streamlined your clients' information — the less strain there is on their network management systems.

SNMP Monitoring Tools & Software for Servers & Network Devices

Almost every single piece of equipment in your infrastructure has the ability to be Monitored using SNMP protocol.

SNMP Protocol allows you to poll device MIBS to extract useful and critical information that will allow you to be proactive about fixing issues that arise before you hear them for your end-users or clients.

On top of keeping your up-time high for all your systems and users, monitoring hardware/software Faults, Availability and Performance issues will be at your finger tips, as many of these programs and tools listed before have intuitive and simple interfaces with built-in reporting and graphs to help you manage everything in a centralized location.

1. Solarwinds NPM

Network Performance Monitor by Solarwinds is SNMP Monitoring tools. Insight for Cisco ASA that really brings a new level of monitoring to your Cisco ASA devices and helps you automate the monitoring and management of ASA devices within their platform.

NPM has native SNMP mibs for many of the top hardware and software packages available on the market,which can be automatically scanned and added into your inventory within the dashboard. NPM regularly scanning your network for SNMP devices, you can stay on top of

your network and devices, espeically if you have a large network that is dynamic and is always scaling.

2. PRTG

PRTG Network Monitor by Paessler is another great option for your SNMP network management and monitoring tasks. They provide a list of features and advantages of using their flagship product for monitoring devices including:

- Pre-configured SNMP Sensors for Cisco, HP, Dell, Synology and more!
- Pre-Built Alarms and reporting tools
- Support for SNMP v1, V2c and V3
- Auto-Discovery of SNMP Devices within your Network
- and much more!

On top of SNMP monitoring features, PRTG offers WMI, Netflow/IPFix and Packet Sniffing capabilities within their software as well. PRTG offers a Free Version of their Popular Network Monitor that allows you to monitor up to 100 FREE of Charge.

3. WhatsUp Gold

WhatsUp Gold 2017 by IpSwitch provide great support for SNMP Traps and Real-time performance updates with assist with troubleshooting tasks. WhatsUp Gold also has a slew of Built-in scripts for SNMP devices that assist in the discovery and mapping of network components within their dashboard.

4. ManageEngine OpManager

OpManager by ManageEngine provides enterprise level reporting and SNMP monitoring capabilities in an easy-to-use managment interface. Out of the Box support for all major hardware and software,

From VMWare Server Monitoring to Hyper-V Virtual Machine monitoring capabilities to Citrix XenServer monitoring, you can monitor your entire Virtualized environment right off the bat.

On top of supporting SNMP v1, v2 and v3, they also can monitor via ICMP, WMI and telnet if necessary for devices non-SNMP devices.

Above are some of the paid software solutions that are the most popular in terms of SNMP monitoring, but we understand that not everyone is inclined to using a commercially available product and would like to have either great flexibility with the software they're using or your budget is non-existent.

- Nagios Core
- Pandora FMS
- MRTG
- Spiceworks
- Zabbix
- LibreNMS
- OpenNMS
- Observium (free version available)
- Cacti
- Icinga
- Zenoss Core
- and many more

The software from above have freely available downloads or freeware versions that can be used within your network as you see fit. Some of them have fully featured versions that need additional licensing and have incurring costs associated with them.

SNMP traps and queries

https://help.fortinet.com/fweb/560/Content/FortiWeb/fortiweb-admin/snmp_traps.htm

Action and Syntax	Details
Display SNMP trap server on/off status and version information. `show trap config`	SNMP monitor service and SNMP trap settings are independent, but SNMP monitor service must be enabled before you activate the SNMP trap configuration.
Display a table of SNMP trap events and settings. `show trap events`	
Save SNMP trap events settings for editing or later use. `save trap --location <filestore_alias> [--default]`	Saves default trap settings for editing. If "--default" is not specified, saves current trap settings. Example: `(config)# save trap --location samba-fs`
Enable or disable SNMP traps. `set trap service --status <enabled|disabled>`	SNMP monitor service and SNMP trap settings are independent, but SNMP monitor service must be enabled to activate the SNMP trap configuration. Example: `(config)# set trap service --status enabled`
Load SNMP trap events configuration from a file. `load trap --location <filestore_alias> --file <name>`	Enter the name of a predefined remote filestore alias. Example: `(config)# load trap --location samba-fs --file list123`
Send a test trap to verify SNMP communication. `test trap event`	If there is a problem sending the test trap, verify the engine ID and authentication settings and values, and verify that the network allows communication between the appliance and the SNMP manager.

Configure SNMP v1 traps for alerting. `set trap v1 --community <name> --ip <ip_address> --port <port>`	Enter a community name, trap server IP address, and port for traps sent by the appliance. The community name must be 5 to 64 characters long, with no spaces. Alphanumeric characters and $ ( ) . _ @ = * < > - : , % [ ] | / Example: `(config)# set trap v1 --community myv1community --ip 10.0.0.14 --port 162`
Configure SNMP v2c traps for alerting. `set trap v2c --community <name> --ip <ip_address> --port <port>`	Enter a community name, trap server IP address, and port for traps sent by the appliance. The community name must be 5 to 64 characters long, with no spaces. Alphanumeric characters and $ ( ) . _ @ = * < > - : , % [ ] | / Example: (config)# set trap v2c --community myv2community --ip 10.0.0.13 --port 162

Configure SNMP v3 traps for alerting.
`set trap v3 --engineid <id> --ip <ip_address> --port <port> --securitylevel <level> ...`
	There are 3 levels of security available for SNMP v3 traps:
	No authentication or encryption:
	`noAuthNoPriv`
	Authentication only:
	`authNoPriv`
	Authentication and encryption:
	`authPriv`
	See full syntax for each security level, immediately below.

Configure SNMP v3 traps with no authentication or encryption. `set trap v3 --engineid <id> --ip <ip_address> --port <port> --securitylevel noAuthNoPriv --user <username>`	Specify the engine ID, IP address, port, and user name to use for communication with your SNMP manager. The engine ID is a hexadecimal number between 10 and 64 characters long. The number cannot be all 0 or F characters, and the length of the string must be an even number. User is the account name to use for SNMP communication. Enter a name between 1 and 15 characters, with no spaces. Only alphanumeric characters can be used. Example: `(config)# set trap v3 --engineid 8000000001020304 --ip 10.0.0.13 --port 162 --securitylevel noAuthNoPriv --user trapuser`
Configure SNMP v3 traps with authentication only. `set trap v3 --engineid <id> --ip <ip_address> --port <port> --securitylevel authNoPriv --user <username> --authentication <md5|sha>`	Specify the engine ID, IP address, port, and user name to use for communication with your SNMP manager. The engine ID is a hexadecimal number between 10 and 64 characters long. The number cannot be all 0 or F characters, and the length of the string must be an even number. User is the account name to use for SNMP communication. Enter a name with 1-15 alphanumeric characters, with no spaces. Specify the authentication protocol used on the trap server (md5 or sha). You are prompted for a password. Enter a password between 1 and 64 characters, with no spaces. All other ASCII characters are okay. Example: `(config)# set trap v3 --engineid 0x802a0581 --ip 10.17.32.5 --port 162 --securitylevel authNoPriv --authentication sha --user test Password: ********`
Configure SNMP v3 traps with authentication and encryption. `set trap v3 --engineid <id> --ip <ip_address> --port <port> --securitylevel authPriv --user <username> --authentication <md5|sha> --encrypt <des|aes>`	Specify the engine ID, IP address, port, and user name to use for communication with your SNMP manager. The engine ID is a hexadecimal number between 10 and 64 characters long. The number cannot be all 0 or F characters, and the length of the string must be an even number. User is the account name to use for SNMP communication. Enter a name with 1-15 alphanumeric characters, with no spaces. Specify the authentication protocol used on the trap server (md5 or sha), and the SNMP encryption protocol (des or aes). You are prompted for a password and encryption key. The 1 to 64 characters, and the key 8 to 64 characters long, with no spaces. All other ASCII characters can be used. Example: `(config)# Set trap v3 --engineid 8000000001020304 --ip 10.0.0.25 --port 162 --securitylevel authPriv --user trapuser --authentication sha --encrypt md5`