



Tech Info Library

SNMP Technical Introduction, Part 2 of 3

Article Created: 10 November 1992

TOPIC -----

This is the second of three articles that introduce SNMP (Simple Network Management Protocol). This article describes SNMP's history, management model, and architecture.

DISCUSSION -----

SNMP History

Simple Network Management Protocol (SNMP) originated in the Internet community as a means for managing TCP/IP networks and Ethernet networks.

Every large network sooner or later displays a need for management and control, and the worldwide TCP/IP Internet is no exception.

In November 1987, a working group of the Internet Engineering Task Force (IETF) submitted a proposal to address some of these needs. They defined the Simple Gateway Management Protocol (SGMP), an early attempt at TCP/IP Internet management, and the forerunner of SNMP. SGMP is similar to SNMP in design and architecture, however syntax differences make the two incompatible.

In August 1988, the same four authors enhanced their proposal, and submitted a draft standard (Request For Comments 1098 entitled "A Simple Network Management Protocol"). RFCs are a method of defining and discussing enhancements to the Internet.

One of the main design criteria was simplicity. Another was to make the monitoring and control capabilities quite extensive. Due to the variety of devices on the Internet, the authors wanted to create an architecture independent of any specific host/gateway architecture.

In April 1989, RFC 1098 became an Internet Recommended Standard, and was rapidly implemented on the worldwide Internet.

SNMP's appeal broadened rapidly beyond the Internet community. SNMP attracted users looking for a proven and readily available way to manage their networks. SNMP's monitoring and control mechanisms operate

completely independent of the TCP/IP protocol. SNMP requires only the datagram transport mechanism to operate. It can be implemented over any network media or protocol suite, including AppleTalk.

In May 1990, RFC 1098 was re-released, with typographical errors corrected, and a rewritten status section. RFC 1157 is the basis for Apple's network management strategy.

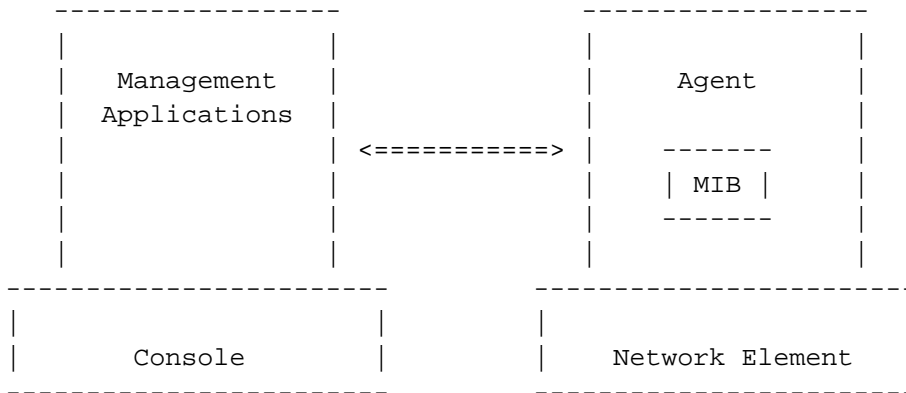
The Management Model

A network management system contains three main components:

- Several network elements, or nodes, which need to be managed.
- At least one management station used to administer the network.
- A management protocol that is used to communicate between the nodes and the manager.

SNMP Architecture

The SNMP architecture conforms to this management model. It defines these three major components: managed nodes (network elements), at least one network management station (console), and a protocol to communicate between the two. The diagram below displays this relationship in SNMP.



Network Elements (Nodes)

Network elements are devices such as hosts, gateways, terminal servers, and desktop computers. They are also known as managed nodes, nodes, or "entities." Network Elements can also be hubs or multiplexors. Although simple in concept, the network elements actually perform most of the work in SNMP, and contain all the information a management console needs to perform its function.

Consoles

Consoles are the management stations. They execute management applications to monitor and control network elements. They display, often graphically, the state of the current network and any alarms present.

The Protocol

SNMP is used to communicate management information between the network

management stations and the agents in the network elements.

Inside the Network Element (Node)

Agents. Network Elements have agents responsible for performing the functions requested by the network management stations. Agents do the real work in SNMP management. Agents execute commands issued by the console, as well as reporting on unusual events.

Agents control and provide access to a Management Information Base (MIB), which contains the information to respond to console requests and updates.

Overview of the MIB

SNMP depends on the relationship between the console, agents, and the MIBs and agent controls.

The MIB contains groups, which are collections of objects.

Objects define data types, such as text strings, counters, gauges, time ticks, addresses, and integers. Each object has a unique name. (The naming convention used is discussed in the next section.)

The data fields that objects define are called variables. These are the fields where the actual data or value is stored. Each variable also has a unique name, using the same convention as the objects.

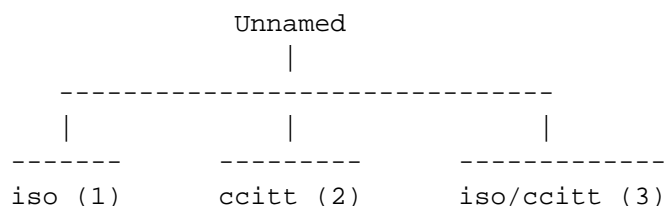
MIBs are encoded using OSI's Abstract Syntax Notation One (ASN.1). This language is used to define the objects in the MIB. ASN.1 is an international standard, implemented by most of the world's computer manufacturers. ASN.1 encoding makes the MIB platform independent.

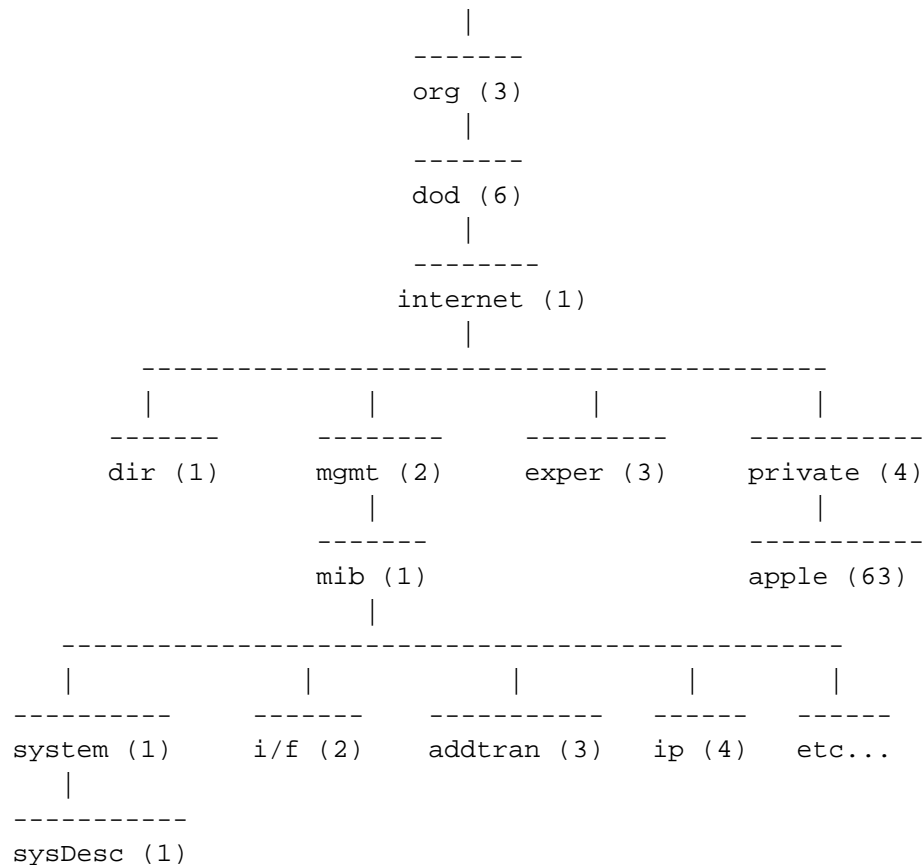
The Structure and Representation of MIB Objects and Variable Names

Names used for MIB objects and their variables are taken from the object identifier namespace, administered by ISO and CCITT. The key idea behind the object identifier is to provide an environment in which all possible objects can be uniquely named.

Authority for parts of the namespace is subdivided at each level, allowing individual groups to assign names without consulting a central authority for each assignment.

This tree structure is used to identify each and every MIB object, and in fact they are accessed programmatically using this structure.





In the figure above, each item has a number associated with it. These numbers correspond to specific paths to locate information in an SNMP network element, and give each object a unique name. For example, the unique name for the object that contains the system description (machine type) of the network element is described as:

```
iso.org.dod.internet.mgmt.mib.sys.sysDesc
```

The number representation of the above map is:

```
1.3.6.1.2.1.1.1
```

SNMP consoles actually use the number string to address the objects in the packets that travel the network, and in the programs used to construct the packets. Although this number string addresses the object, it is not enough to address the first variable containing the data or value, which is the number the console really wants.

There are no zeros on the map because a zero is used to reference the actual data (value) of a variable.

To obtain a value from the object named sysDesc, add a zero to the number string. Therefore, a Get command would use 1.3.6.1.2.1.1.1.0 to obtain this value. An example of a response would be the text string "Macintosh IIX."

A significant benefit of this architecture is that it allows the development of consoles that can manage the products of many different vendors -- because all the objects are unique. Generic SNMP consoles, which often know little or nothing about specific manageable network devices, incorporate, or are able to import, MIBs because the coding and data representation rules are standardized, and the object names are preassigned and unique.

One of the main features of SNMP is allowing vendor-specific implementations of the MIB. Non-standard, vendor-specific MIBs would come under the Enterprise branch of the object identifier namespace. Organizations, such as Apple, can apply to the Internet Activities Board (IAB) for specific Enterprise numbers to keep their MIBs unique. Apple Computer's Enterprise number is 63, and over 100 other companies have applied for unique numbers, to provide an indication of how many vendor-specific MIBs will be implemented over time.

Consoles can compile imported MIBs and issue commands to agents. These consoles then display, often graphically, the information returned by the agents. Because these generic consoles are not product specific (MIBs are) they can even be used to manage network products that ship after the console itself is sold.

Copyright 1992, Apple Computer, Inc.

Keywords: <None>

=====

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 11016