# Network Management

Network Managementas monitoring, testing, configuring, and troubleshooting network components to meet a set of requirements defined by an organization. These requirements include the smooth, efficient operation of the network that provides the predefined Quality of Service for users. To accomplish this task, a network management system uses hardware, software, and humans.

The International Organization for Standardization (ISO) defines five areas of network management: Configuration Management, Fault Management, Performance Management, Security Management, And Accounting Management, as shown in Figure 1.

Network Management

Configuration Fault Performance Management Management Management

Security Accounting Management Management

 **Figure. 1** *Areas of network management*

## Configuration Management

A large network is usually made up of hundreds of entities that are physically or logically connected to each other. These entities have an initial configuration when the network is set up, but can change with time. Desktop computers may be replaced by others; application software may be updated to a newer version; and users may move from one group to another. The *configuration management* system must know, at any time, the status of each entity and its relation to other entities. Configuration management can be divided into two subsystems: *reconfiguration* and *documentation*.

#### Reconfiguration:

Reconfiguration can be a daily occurrence in a large network. There are three types of reconfigurations: *hardware reconfiguration*, *software reconfiguration,* and *user-account reconfiguration*.

#### Hardware Reconfiguration:

Hardware reconfiguration covers all changes to the hardware. For example, a desktop computer may need to be replaced. A router may need to be moved to another part of the network. A subnetwork may be added or removed from the network. All of these need the time and attention of network management. In a large network, there must be specialized personnel trained for quick and efficient hardware reconfiguration. Unfortunately, this type of reconfiguration cannot be automated and must be manually handled case by case.

#### Software Reconfiguration:

Software reconfiguration covers all changes to the software. For example, new software may need to be installed on servers or clients. An operating system may need updating. Fortunately, most software reconfiguration can be automated. For example, an update for an application on some or all clients can be electronically downloaded from the server.

#### User-Account Reconfiguration:

User-account reconfiguration is not simply adding or deleting users on a system. We must also consider the user privileges, both as an individual and as a member of a group. For example, a user may have both read and write permission with regard to some files, but only read permission with regard to other files. User-account reconfiguration can be, to some extent, automated. For example, in a college or university, at the beginning of each quarter or semester, new students are added to the system. The students are normally grouped according to the courses they take or the majors they pursue. The members of each group have specific privileges; computer science students may need to access a server providing different computer language facilities, while engineering students may need to access servers that provide computer assisted design (CAD) software.

#### Documentation

The original network configuration and each subsequent change must be recorded meticulously. This means that there must be documentation for hardware, software, and user accounts.

#### Hardware Documentation:

*Hardware documentation* normally involves two sets of documents: maps and specifications.

* ***Maps:*** *Maps* track each piece of hardware and its connection to the network. There can be one general map that shows the logical relationships between subnetworks. There can also be a second general map that shows the physical location of each sub- network. For each subnetwork, then, there is one or more maps that show all pieces of equipment. The maps use some kind of standardization to be easily read and understood by current and future personnel.
* ***Specifications:*** Maps are not enough. Each piece of hardware also needs to be documented. There must be a set of *specifications* for each piece of hardware connected to the network. These specifications must include information such as hardware type, serial number, vendor (address and phone number), time of purchase, and warranty information.

####  Software Documentation:

All software must also be documented. *Software documentation* includes information such as the software type, the version, the time installed, and the license agreement.

#### User-Account Documentation:

Most operating systems have a utility that allows *user account documentation*. The management must make sure that the files with this information are updated and secured. Some operating systems record access privileges in two documents; one shows all files and access types for each user; the other shows the list of users that have access to a particular file.

## Fault Management

Complex networks today are made up of hundreds and sometimes thousands of components. Proper operation of the network depends on the proper operation of each component individually and in relation to each other. *Fault management* is the area of network management that handles this issue. An effective fault management system has two subsystems: reactive fault management and proactive fault management.

#### 2.1 Reactive Fault Management

A *reactive fault management* system is responsible for detecting, isolating, correcting, and recording faults. It handles short-term solutions to faults.

#### Detecting Fault

The first step taken by a reactive fault management system is to find the exact location of the fault. A fault is defined as an abnormal condition in the system. When a fault occurs, either the system stops working properly or the system creates excessive errors. A good example of a fault is a damaged communication medium.

####  Isolating Fault

The next step taken by a reactive fault management system is isolating the fault. A fault, if isolated, usually affects only a few users. After isolation, the affected users are immediately notified and given an estimated time of correction.

#### Correcting Fault

The next step is correcting the fault. This may involve replacing or repairing the faulty components.

#### Recording Fault

After the fault is corrected, it must be documented. The record should show the exact location of the fault, the possible cause, the action or actions taken to correct the fault, the cost, and the time it took for each step. Documentation is extremely important for several reasons:

* The problem may reoccur. Documentation can help the present or future administrator or technician solve a similar problem.
* The frequency of the same kind of failure is an indication of a major problem in the system. If a fault happens frequently in one component, the component should be replaced with a similar one or the whole system should be changed to avoid the use of that type of component.
* The statistic is helpful to another part of network management, performance management.

#### 2.2 Proactive Fault Management

*Proactive fault management* tries to prevent faults from occurring. Although this is not always possible, some types of failures can be predicted and prevented. For example, if a manufacturer specifies a lifetime for a component or a part of a component, it is a good strategy to replace it before that time. As another example, if a fault happens frequently at one particular point in a network, it is wise to carefully reconfigure the network to prevent the fault from happening again.

## Performance Management

*Performance management,* which is closely related to fault management, tries to monitor and control the network to ensure that it is running as efficiently as possible. Performance management tries to quantify performance using some measurable quantity, such as capacity, traffic, throughput, or response time. Some protocols, such as SNMP can be used in performance management.

#### Capacity

One factor that must be monitored by a performance management system is the *capacity* of the network. Every network has a limited capacity and the performance management system must ensure that it is not used above this capacity. For example, if a LAN is designed for 100 stations at an average data rate of 2 Mbps, it will not operate properly if 200 stations are connected to the network. The data rate will decrease and blocking may occur.

#### Traffic

*Traffic* can be measured in two ways: internally and externally. **Internal traffic is measured by the number of packets (or bytes) travelling inside the network. External traffic is measured by the exchange of packets (or bytes) outside the network**. During peak hours, when the system is heavily used, blocking may occur if there is excessive traffic.

#### Throughput

We can measure the *throughput* of an individual device (such as a router) or a part of the network. Performance management monitors the throughput to make sure that it is not reduced to unacceptable levels.

#### Response Time

*Response time* is normally measured from the time a user requests a service to the time the service is granted. Other factors such as capacity and traffic can affect the response time. Performance management monitors the average response time and the peak-hour response time. Any increase in response time is a very serious condition as it is an indication that the network is working above its capacity.

## Security Management

*Security management* is responsible for controlling access to the network based on pre-defined policy. Security tools such as encryption and authentication. Encryption allows privacy for users; authentication forces the users to identify themselves.

1. Accounting Management

*Accounting management* is the controlling of users’ access to network resources through charges. Under accounting management, individual users, departments, divisions, or even projects are charged for the services they receive from the network. Charging does not necessarily mean cash transfer; it may mean debiting the departments or divisions for budgeting purposes.

Today, organizations use an accounting management system for the following reasons:

* + - * It prevents users from monopolizing limited network resources.
			* It prevents users from using the system inefficiently.
			* Network managers can do short- and long-term planning based on the demand for network use.

# Simple Network Management Protocol (SNMP)

Several network management standards have been devised during the last few decades. The most important one is **Simple Network Management Protocol (SNMP),** used by the Internet. We discuss this standard in this section. SNMP is a framework for managing devices in an internet using the TCP/IP protocol suite. It provides a set of fundamental operations for monitoring and maintaining an internet. SNMP uses the concept of manager and agent. That is, a manager, usually a host, controls and monitors a set of agents, usually routers or servers (see Figure. 2).

**Figure. 2** *SNMP concept*

Agent variables

Agent

Agent

Agent

Manager

Agent

SNMP is an application-level protocol in which a few manager stations control a set of agents. The protocol is designed at the application level so that it can monitor devices made by different manufacturers and installed on different physical networks.

## Managers and Agents

A management station, called a *manager,* is a host that runs the SNMP client program. A managed station, called an *agent,* is a router (or a host) that runs the SNMP server program. Management is achieved through simple interaction between a manager and an agent.

The agent keeps performance information in a database. The manager has access to the values in the database. For example, a router can store in appropriate variables the number of packets received and forwarded. The manager can fetch and compare the values of these two variables to see if the router is congested or not.

The manager can also make the router perform certain actions. For example, a router periodically checks the value of a reboot counter to see when it should reboot itself. It reboots itself, for example, if the value of the counter is 0. The manager can use this feature to reboot the agent remotely at any time. It simply sends a packet to force a 0 value in the counter.

Agents can also contribute to the management process. The server program running on the agent

can check the environment and, if it notices something unusual, **it can send a warning message**

**(called a *Trap*) to the manager.**

In other words, management with SNMP is based on three basic ideas:

1. A manager checks an agent by requesting information that reflects the behaviour of the agent.
2. A manager forces an agent to perform a task by resetting values in the agent database.
3. An agent contributes to the management process by warning the manager of an unusual situation.

## Management Components

To do management tasks, SNMP uses two other protocols: **Structure of Management Information (SMI)** and **Management Information Base (MIB).** In other words, management on the Internet is done through the cooperation of three protocols: SNMP, SMI, and MIB, as shown in Figure 3.

**Figure. 3.** *Components of network management on the Internet*

Management

MIB

SMI

**SNMP**

#### Role of SNMP

SNMP defines the format of the packet to be sent from a manager to an agent and vice versa. It also interprets the result and creates statistics (often with the help of other management software). The packets exchanged contain the object (variable) names and their status (values). SNMP is responsible for reading and changing these values.

**SNMP defines the format of packets exchanged between a manager and an agent. It reads and changes the status of objects (values of variables) in SNMP packets.**

#### Role of SMI

To use SNMP, we need rules for naming objects. This is particularly important because the objects in SNMP form a hierarchical structure (an object may have a parent object and some child objects).

**SMI defines the general rules for naming objects, defining object types (including range and length), and showing how to encode objects and values.**

Role of MIB

MIB creates a set of objects defined for each entity in a manner similar to that of a database (mostly metadata in a database, names and types without values).

**MIB creates a collection of named objects, their types, and their relationships to each other in an entity to be managed.**

An Example:

A manager station (SNMP client) wants to send a message to an agent station (SNMP server) to find the number of UDP user datagrams received by the agent.

Figure 4 shows an overview of steps involved.

MIB is responsible for finding the object that holds the number of UDP user datagrams received.

SMI, with the help of another embedded protocol, is responsible for encoding the name of the object.

SNMP is responsible for creating a message, called a GetRequest message, and encapsulating the encoded message. Of course, things are more complicated than this simple overview, but we first need more details of each protocol.

**Figure 4.** *Management overview*

6

agent**Six user datagrams received.a**

6

Time **Six user datagrams received.**

Manager

6

Time **Six user datagrams received.**

SMI

MIB SNMP

Integer values are defined using three attributes

The object has an integer value and is called udpInDatagram with id 1.3.6.1.2.1.7.1.0

Encapsulate the request in a GetRequest message

Internet

**1**

Question **What is the number of UDP user datagrams received?**

2

3

SNMP packet

**4**

SNMP packet

**5**

6

Answer **Six user datagrams received.**

Response

GetRequest

##### The Syntax of Three Protocols (SMI. SNMP and MIB)

SMI is a guideline for SNMP. It emphasizes three attributes to handle an object: name, data type, and encoding method. Its functions are:

* + - * To name objects.
			* To define the type of data that can be stored in an object.
			* To show how to encode data for transmission over the network.