CHAPTER 20

NETWORK MANAGEMENT

20.1 Network Management Requirements
- Fault Management
- Accounting Management
- Configuration and Name Management
- Performance Management
- Security Management

20.2 Network Management Systems
- Architecture of a Network Management System

20.3 Simple Network Management Protocol (SNMP)
- Simple Network Management Protocol Version 1 (SNMPv1)
- Simple Network Management Protocol Version 2 (SNMPv2)
- Simple Network Management Protocol Version 3 (SNMPv3)

20.4 Recommended Reading and Web Sites

20.5 Key Terms, Review Questions, and Problems

Chapter Objectives

After reading this chapter, you should be able to

- ♦ List and define the key requirements that a network management system should satisfy.
- ♦ Give an overview of the architecture of a network management system and explain each of its key elements.
- ♦ Describe SNMP and list the differences among versions 1, 2, and 3.

Networks and distributed processing systems are of critical and growing importance in enterprises of all sorts. The trend is toward larger, more complex networks supporting more applications and more users. As these networks grow in scale, two facts become painfully evident:

- The network and its associated resources and distributed applications become indispensable to the organization.
- More things can go wrong, disabling the network or a portion of the network or degrading performance to an unacceptable level.

A large network cannot be put together and managed by human effort alone. The complexity of such a system dictates the use of automated network management tools. The urgency of the need for such tools is increased, and the difficulty of supplying such tools is also increased, if the network includes equipment from multiple vendors. Moreover, the increasing decentralization of network services as exemplified by the increasing importance of workstations and client/server computing makes coherent and coordinated network management increasingly difficult. In such complex information systems, many significant network assets are dispersed far from network management personnel.

This chapter provides an overview of network management. We begin by looking at the requirements for network management. This should give some idea of the scope of the task to be accomplished. To manage a network, it is fundamental to know something about the current status and behavior of that network.

For either LAN management alone, or for a combined LAN/WAN environment, what is needed is a network management system that includes a comprehensive set of data gathering and control tools and that is integrated with the network hardware and software. We look at the general architecture of a network management system and then examine the most widely used standardized software package for supporting network management: SNMP.

20.1 NETWORK MANAGEMENT REQUIREMENTS

Table 20.1 lists key areas of network management as suggested by the International Organization for Standardization (ISO). These categories provide a useful way of organizing our discussion of requirements.

Table 20.1 ISO Management Functional Areas

Fault management
The facilities that enable the detection, isolation, and correction of abnormal operation of the OSI environment.
Accounting management
The facilities that enable charges to be established for the use of managed objects and costs to be identified for the use of those managed objects.
Configuration and name management
The facilities that exercise control over, identify, collect data from, and provide data to managed objects for the purpose of assisting in providing for continuous operation of interconnection services.
Performance management
The facilities needed to evaluate the behavior of managed objects and the effectiveness of communication activities.
Security management
Those aspects of OSI security essential to operate OSI network management correctly and to protect managed objects.

Fault Management

OVERVIEW To maintain proper operation of a complex network, care must be taken that systems as a whole, and each essential component individually, are in proper working order. When a fault occurs, it is important, as rapidly as possible, to

- Determine exactly where the fault is.
- Isolate the rest of the network from the failure so that it can continue to function without interference.
- Reconfigure or modify the network in such a way as to minimize the impact of operation without the failed component or components.
- Repair or replace the failed components to restore the network to its initial state.

Central to the definition of fault management is the fundamental concept of a fault. Faults are to be distinguished from errors. A **fault** is an abnormal condition that requires management attention (or action) to repair. A fault is usually indicated by failure to operate correctly or by excessive errors. For example, if a communications line is physically cut, no signals can get through. Or a crimp in the cable may cause wild distortions so that there is a persistently high bit error rate. Certain errors (e.g., a single bit error on a communication line) may occur occasionally and are not normally considered to be faults. It is usually possible to compensate for errors using the error control mechanisms of the various protocols.

USER REQUIREMENTS Users expect fast and reliable problem resolution. Most end users will tolerate occasional outages. When these infrequent outages do occur, however, the user generally expects to receive immediate notification and expects that the problem will be corrected almost immediately. To provide this level of fault resolution requires very rapid and reliable fault detection and diagnostic management functions. The impact and duration of faults can also be minimized by the use

of redundant components and alternate communication routes, to give the network a degree of fault tolerance. The fault management capability itself should be redundant to increase network reliability.

Users expect to be kept informed of the network status, including both scheduled and unscheduled disruptive maintenance. Users expect reassurance of correct network operation through mechanisms that use confidence tests or analyze dumps, logs, alerts, or statistics. After correcting a fault and restoring a system to its full operational state, the fault management service must ensure that the problem is truly resolved and that no new problems are introduced. This requirement is called problem tracking and control.

As with other areas of network management, fault management should have minimal effect on network performance.

Accounting Management

OVERVIEW In many enterprise networks, individual divisions or cost centers, or even individual project accounts, are charged for the use of network services. These are internal accounting procedures rather than actual cash transfers, but they are important to the participating users nevertheless. Furthermore, even if no such internal charging is employed, the network manager needs to be able to track the use of network resources by user or user class for a number of reasons, including the following:

- A user or group of users may be abusing their access privileges and burdening the network at the expense of other users.
- Users may be making inefficient use of the network, and the network manager can assist in changing procedures to improve performance.
- The network manager is in a better position to plan for network growth if user activity is known in sufficient detail.

USER REQUIREMENTS The network manager needs to be able to specify the kinds of accounting information to be recorded at various nodes, the desired interval between successive sendings of the recorded information to higher-level management nodes, and the algorithms to be used in calculating the charging. Accounting reports should be generated under network manager control.

To limit access to accounting information, the accounting facility must provide the capability to verify users' authorization to access and manipulate that information.

Configuration and Name Management

OVERVIEW Modern data communication networks are composed of individual components and logical subsystems (e.g., the device driver in an operating system) that can be configured to perform many different applications. The same device, for example, can be configured to act either as a router or as an end system node or both. Once it is decided how a device is to be used, the configuration manager can choose the appropriate software and set of attributes and values (e.g., a transport layer retransmission timer) for that device.

Configuration management is concerned with initializing a network and gracefully shutting down part or all of the network. It is also concerned with

maintaining, adding, and updating the relationships among components and the status of components themselves during network operation.

USER REQUIREMENTS Startup and shutdown operations on a network are the specific responsibilities of configuration management. It is often desirable for these operations on certain components to be performed unattended (e.g., starting up or shutting down a network interface unit). The network manager needs the capability to identify initially the components that comprise the network and to define the desired connectivity of these components. Those who regularly configure a network with the same or a similar set of resource attributes need ways to define and modify default attributes and to load these predefined sets of attributes into the specified network components. The network manager needs the capability to change the connectivity of network components when users' needs change. Reconfiguration of a network is often desired in response to performance evaluation or in support of network upgrade, fault recovery, or security checks.

Users often need to, or want to, be informed of the status of network resources and components. Therefore, when changes in configuration occur, users should be notified of these changes. Configuration reports can be generated either on some routine periodic basis or in response to a request for such a report. Before reconfiguration, users often want to inquire about the upcoming status of resources and their attributes.

Network managers usually want only authorized users (operators) to manage and control network operation (e.g., software distribution and updating).

Performance Management

OVERVIEW Modern data communications networks are composed of many and varied components, which must intercommunicate and share data and resources. In some cases, it is critical to the effectiveness of an application that the communication over the network be within certain performance limits. Performance management of a computer network comprises two broad functional categories—monitoring and controlling. Monitoring is the function that tracks activities on the network. The controlling function enables performance management to make adjustments to improve network performance. Some of the performance issues of concern to the network manager are as follows:

- What is the level of capacity utilization?
- Is there excessive traffic?
- Has throughput been reduced to unacceptable levels?
- Are there bottlenecks?
- Is response time increasing?

To deal with these concerns, the network manager must focus on some initial set of resources to be monitored to assess performance levels. This includes associating appropriate metrics and values with relevant network resources as indicators of different levels of performance. For example, what count of retransmissions on a transport connection is considered to be a performance problem requiring attention? Performance management, therefore, must monitor many resources to provide

information in determining network operating level. By collecting this information, analyzing it, and then using the resultant analysis as feedback to the prescribed set of values, the network manger can become more and more adept at recognizing situations indicative of present or impending performance degradation.

USER REQUIREMENTS Before using a network for a particular application, a user may want to know such things as the average and worst-case response times and the reliability of network services. Thus, performance must be known in sufficient detail to respond to specific user queries. End users expect network services to be managed in such a way as to afford their applications consistently good response time.

Network managers need performance statistics to help them plan, manage, and maintain large networks. Performance statistics can be used to recognize potential bottlenecks before they cause problems to end users. Appropriate corrective action can then be taken. This action can take the form of changing routing tables to balance or redistribute traffic load during times of peak use or when a bottleneck is identified by a rapidly growing load in one area. Over the long term, capacity planning based on such performance information can indicate the proper decisions to make, for example, with regard to expansion of lines in that area.

Security Management

OVERVIEW Security management is concerned with generating, distributing, and storing encryption keys. Passwords and other authorization or access control information must be maintained and distributed. Security management is also concerned with monitoring and controlling access to computer networks and access to all or part of the network management information obtained from the network nodes. Logs are an important security tool, and therefore security management is very much involved with the collection, storage, and examination of audit records and security logs, as well as with the enabling and disabling of these logging facilities.

USER REQUIREMENTS Security management provides facilities for protection of network resources and user information. Network security facilities should be available for authorized users only. Users want to know that the proper security policies are in force and effective and that the management of security facilities is itself secure.

20.2 NETWORK MANAGEMENT SYSTEMS

Architecture of a Network Management System

A **network management system** is a collection of tools for network monitoring and control that is integrated in the following senses:

- A single operator interface with a powerful but user-friendly set of commands for performing most or all network management tasks.
- A minimal amount of separate equipment. That is, most of the hardware and software required for network management is incorporated into the existing user equipment.

A network management system consists of incremental hardware and software additions implemented among existing network components. The software used in accomplishing the network management tasks resides in the host computers and communications processors (e.g., front-end processors, terminal cluster controllers, bridges, routers). A network management system is designed to view the entire network as a unified architecture, with addresses and labels assigned to each point and the specific attributes of each element and link known to the system. The active elements of the network provide regular feedback of status information to the network control center.

Figure 20.1 suggests the architecture of a network management system. Each network node contains a collection of software devoted to the network management task, referred to in the diagram as a network management entity (NME). Each NME performs the following tasks:

- Collect statistics on communications and network-related activities.
- Store statistics locally.
- Respond to commands from the network control center, including commands to
  1. Transmit collected statistics to the network control center.
  2. Change a parameter (e.g., a timer used in a transport protocol).
  3. Provide status information (e.g., parameter values, active links).
  4. Generate artificial traffic to perform a test.
- Send messages to the NCC when local conditions undergo a significant change.

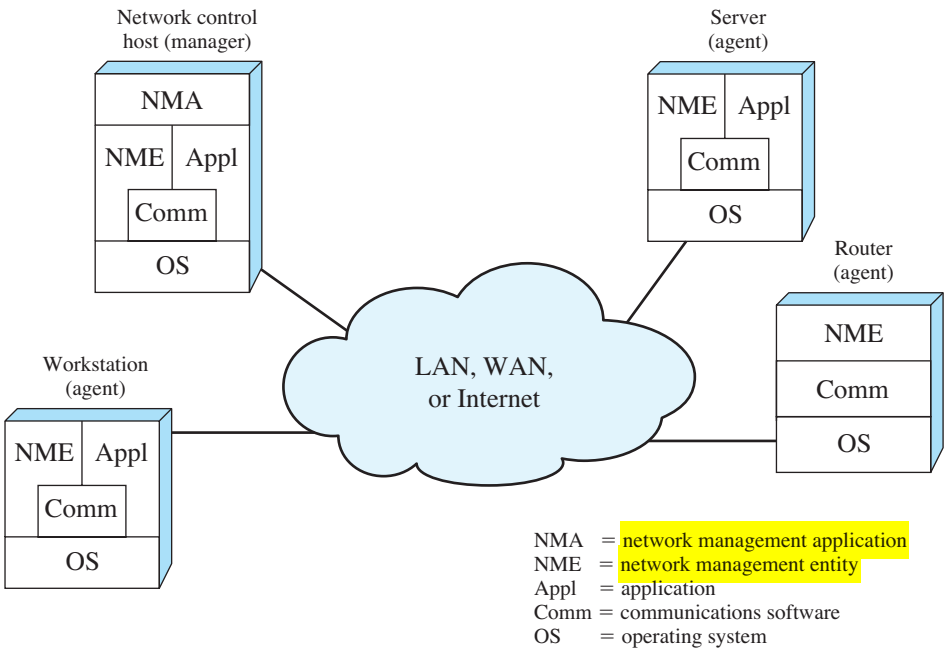

Figure 20.1 Elements of a Network Management System

At least one host in the network is designated as the network control host, or **manager**. In addition to the NME software, the network control host includes a collection of software called the network management application (NMA). The NMA includes an operator interface to allow an authorized user to manage the network. The NMA responds to user commands by displaying information and/or by issuing commands to NMEs throughout the network. This communication is carried out using an application-level network management protocol that employs the communications architecture in the same fashion as any other distributed application.

Each other node in the network that is part of the network management system includes a NME and, for purposes of network management, is referred to as an **agent**. Agents include end systems that support user applications as well as nodes that provide a communications service, such as front-end processors, cluster controllers, bridges, and routers.

As depicted in Figure 20.1, the network control host communicates with and controls the NMEs in other systems. For maintaining high availability of the network management function, two or more network control hosts are used. In normal operation, one of the centers is used for control, while the others are idle or simply collecting statistics. If the primary network control host fails, the backup system can be used.

20.3 SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

Simple Network Management Protocol Version 1 (SNMPv1)

SNMP was developed for use as a network management tool for networks and internetworks operating TCP/IP. It has since been expanded for use in all types of networking environments. The term *simple network management protocol (SNMP)* is actually used to refer to a collection of specifications for network management that include the protocol itself, the definition of a database, and associated concepts.

BASIC CONCEPTS The model of network management that is used for SNMP includes the following key elements:

- Management station, or manager
- Agent
- Management information base
- Network management protocol

The **management station** is typically a standalone device but may be a capability implemented on a shared system. In either case, the management station serves as the interface between the human network manager and the network management system. The management station will have, at minimum,

- A set of management applications for data analysis, fault recovery, and so on
- A user interface by which the network manager may monitor and control the network

- The capability of translating the network manager's requirements into the actual monitoring and control of remote elements in the network
- A database of network management information extracted from the databases of all the managed entities in the network

Only the last two elements are the subject of SNMP standardization.

The other active element in the network management system is the **management agent**. Key platforms, such as hosts, bridges, routers, and hubs, may be equipped with agent software so that they may be managed from a management station. The agent responds to requests for information from a management station, responds to requests for actions from the management station, and may from time to time provide the management station with important but unsolicited information.

To manage resources in the network, each resource is represented as an object. An object is, essentially, a data variable that represents one aspect of the managed agent. The collection of objects is referred to as a **management information base** (MIB). The MIB functions as a collection of access points at the agent for the management station. These objects are standardized across systems of a particular class (e.g., all bridges support the same management objects). A management station performs the monitoring function by retrieving the value of MIB objects. A management station can cause an action to take place at an agent or can change the configuration settings of an agent by modifying the value of specific variables.

The management station and agents are linked by a **network management protocol**. The protocol used for the management of TCP/IP networks is the Simple Network Management Protocol (SNMP). An enhanced version of SNMP, known as SNMPv2, is intended for both TCP/IP-and OSI-based networks. Each of these protocols includes the following key capabilities:

- **Get:** Enables the management station to retrieve the value of objects at the agent
- **Set:** Enables the management station to set the value of objects at the agent
- **Notify:** Enables an agent to send unsolicited notifications to the management station of significant events

In a traditional centralized network management scheme, one host in the configuration has the role of a network management station; there may be one or two other management stations in a backup role. The remainder of the devices on the network contain agent software and a MIB, to allow monitoring and control from the management station. As networks grow in size and traffic load, such a centralized system is unworkable. Too much burden is placed on the management station, and there is too much traffic, with reports from every single agent having to wend their way across the entire network to headquarters. In such circumstances, a decentralized, distributed approach works best (e.g., Figure 20.2). In a decentralized network management scheme, there may be multiple top-level management stations, which might be referred to as management servers. Each such server might directly manage a portion of the total pool of agents. However, for many of the agents, the management server delegates responsibility to an intermediate manager. The intermediate manager plays the role of manager to monitor and control the agents under

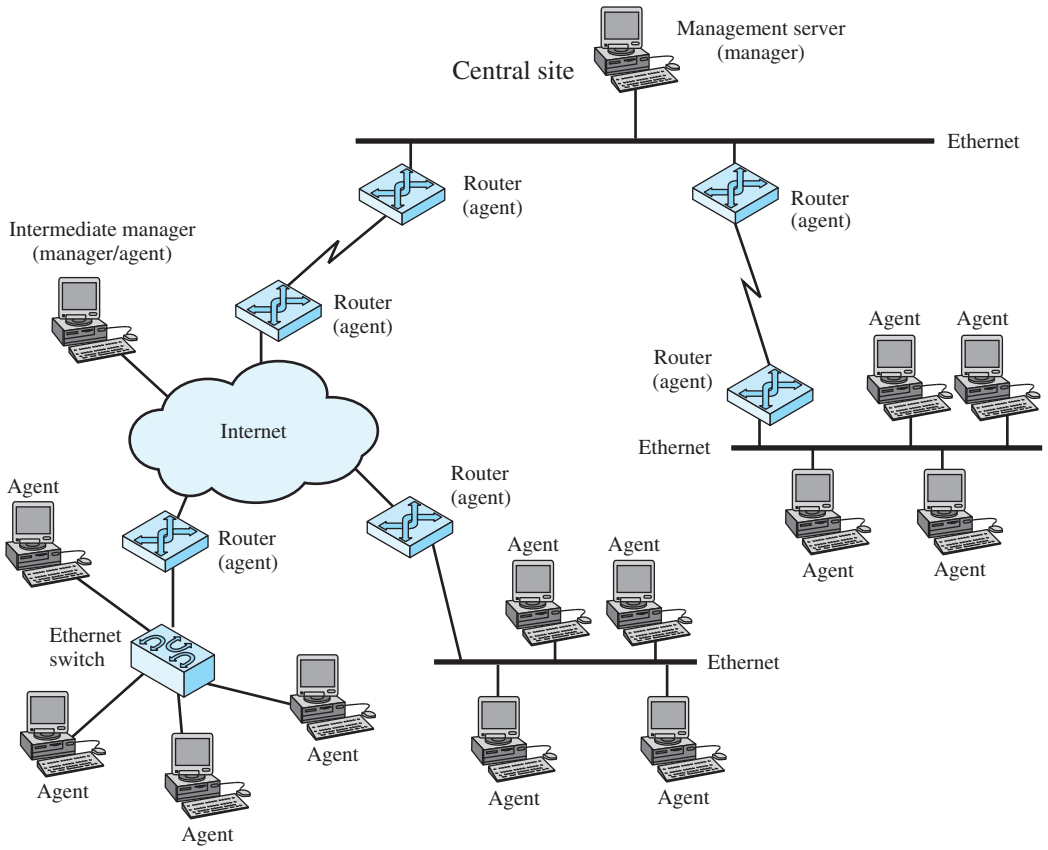

Figure 20.2 Example Distributed Network Management Configuration

its responsibility. It also plays an agent role to provide information and accept control from a higher-level management server. This type of architecture spreads the processing burden and reduces total network traffic.

NETWORK MANAGEMENT PROTOCOL ARCHITECTURE SNMP is an application-level protocol that is part of the TCP/IP protocol suite. It is intended to operate over the user datagram protocol (UDP). Figure 20.3 suggests the typical configuration of protocols for SNMPv1. For a standalone management station, a manager process controls access to a central MIB at the management station and provides an interface to the network manager. The manager process achieves network management by using SNMP, which is implemented on top of UDP, IP, and the relevant network-dependent protocols (e.g., Ethernet, ATM, frame relay).

Each agent must also implement SNMP, UDP, and IP. In addition, there is an agent process that interprets the SNMP messages and controls the agent's MIB. For an agent device that supports other applications, such as FTP, TCP as well as UDP is required. In Figure 20.3, the shaded portions depict the operational environment: that which is to be managed. The unshaded portions provide support to the network management function.

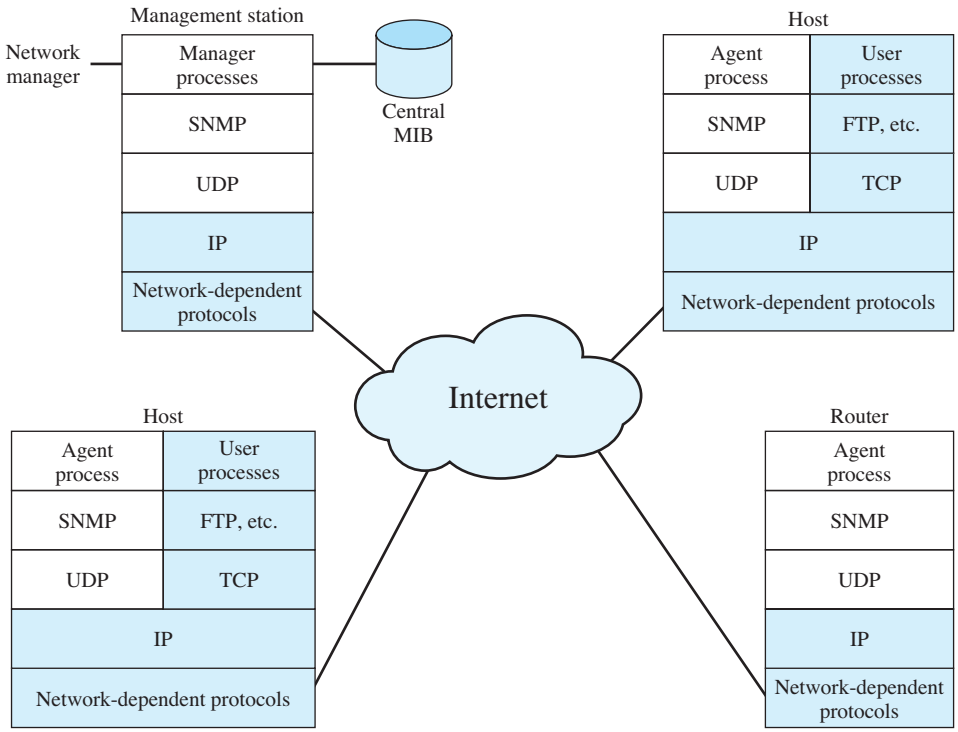

Figure 20.3 SNMPv1 Configuration

Figure 20.4 provides a somewhat closer look at the protocol context of SNMP. From a management station, three types of SNMP messages are issued on behalf of management applications: GetRequest, GetNextRequest, and SetRequest. The first two are two variations of the get function. All three messages are acknowledged by the agent in the form of a GetResponse message, which is passed up to the management application. In addition, an agent may issue a Trap message in response to an event that affects the MIB and the underlying managed resources. Management requests are sent to UDP port 161, while the agent sends traps to UDP port 162.

Because SNMP relies on UDP, which is a connectionless protocol, SNMP is itself connectionless. No ongoing connections are maintained between a management station and its agents. Instead, each exchange is a separate transaction between a management station and an agent.

Simple Network Management Protocol Version 2 (SNMPv2)

In August of 1988, the specification for SNMP was issued and rapidly became the dominant network management standard. A number of vendors offer standalone network management workstations based on SNMP, and most vendors of bridges, routers, workstations, and PCs offer SNMP agent packages that allow their products to be managed by an SNMP management station.

As the name suggests, SNMP is a simple tool for network management. It defines a limited, easily implemented management information base (MIB) of scalar

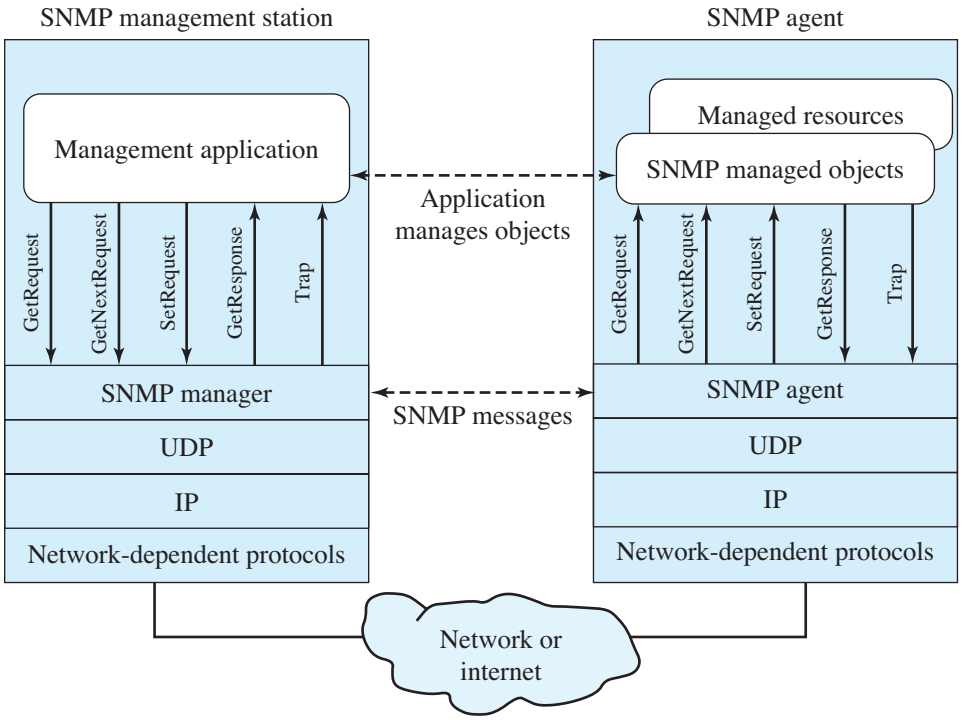

Figure 20.4 The Role of SNMPv1

variables and two-dimensional tables, and it defines a streamlined protocol to enable a manager to get and set MIB variables and to enable an agent to issue unsolicited notifications, called *traps*. This simplicity is the strength of SNMP. SNMP is easily implemented and consumes modest processor and network resources. Also, both the protocol and the MIB structures are sufficiently straightforward that it is not difficult to achieve interoperability among management stations and agent software from a mix of vendors.

With its widespread use, the deficiencies of SNMP became increasingly apparent; these include both functional deficiencies and a lack of a security facility. As a result, an enhanced version, known as SNMPv2, was issued (RFCs 1901, 1905 through 1909, and 2578 through 2580). SNMPv2 quickly gained support, and a number of vendors announced products within months of the issuance of the standard.

THE ELEMENTS OF SNMPV2 As with SNMPv1, SNMPv2 provides a framework on which network management applications can be built. Those applications, such as fault management, performance monitoring, and accounting, are outside the scope of the standard.

SNMPv2 provides the infrastructure for network management. Figure 20.5 is an example of a configuration that illustrates that infrastructure.

The essence of SNMPv2 is a protocol that is used to exchange management information. Each "player" in the network management system maintains a local database of information relevant to network management, known as the MIB. The

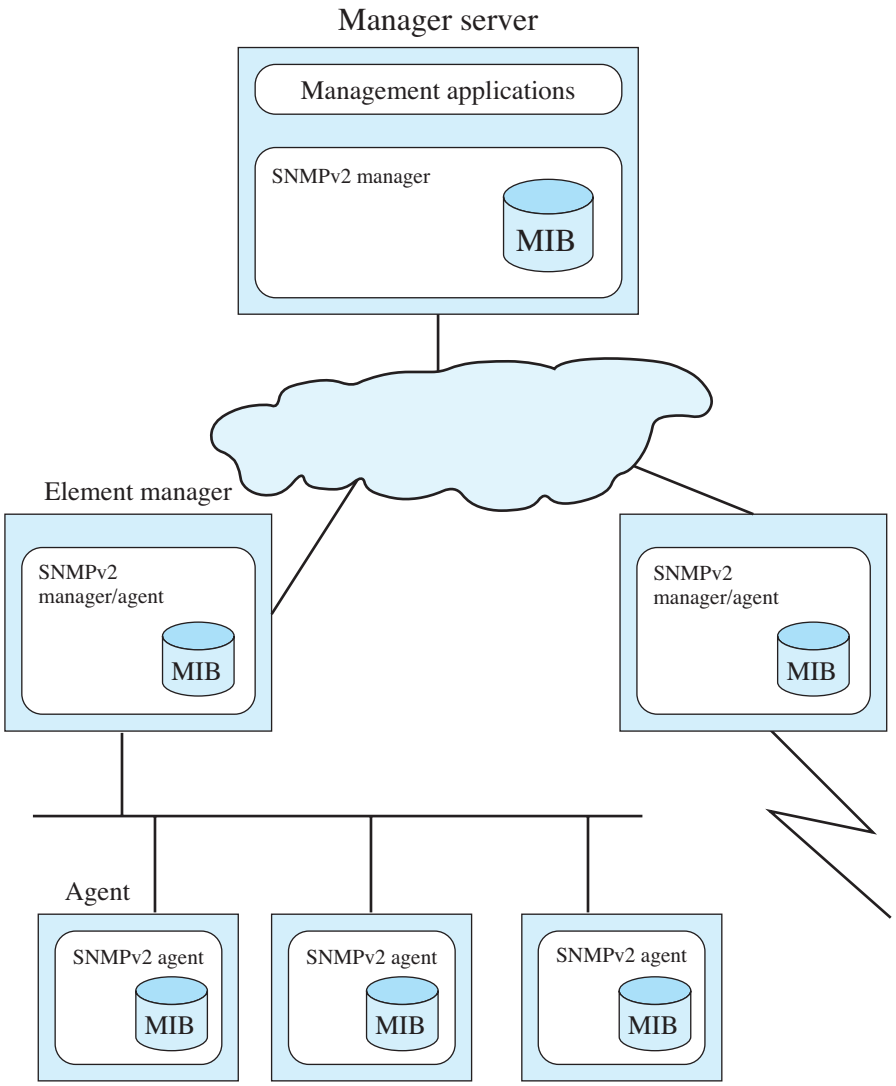

Figure 20.5 SNMPv2-Managed Configuration

SNMPv2 standard defines the structure of this information and the allowable data types; this definition is known as the structure of management information (SMI). We can think of this as the language for defining management information. The standard also supplies a number of MIBs that are generally useful for network management.[1] In addition, new MIBs may be defined by vendors and user groups.

[1]There is a slight fuzziness about the term *MIB*. In its singular form, the term *MIB* can be used to refer to the entire database of management information at a manager or an agent. It can also be used in singular or plural form to refer to a specific defined collection of management information that is part of an overall MIB. Thus, the SNMPv2 standard includes the definition of several MIBs and incorporates, by reference, MIBs defined in SNMPv1.

At least one system in the configuration must be responsible for network management. It is here that any network management applications are hosted. There may be more than one of these management stations, to provide redundancy or simply to split up the duties in a large network. Most other systems act in the role of agent. An agent collects information locally and stores it for later access by a manager. The information includes data about the system itself and may also include traffic information for the network or networks to which the agent attaches.

SNMPv2 supports either a highly centralized network management strategy or a distributed one. In the latter case, some systems operate both in the role of manager and of agent. In its agent role, such a system will accept commands from a superior management system. Some of those commands relate to the local MIB at the agent. Other commands require the agent to act as a proxy for remote devices. In this case, the proxy agent assumes the role of manager to access information at a remote agent, and then assumes the role of an agent to pass that information on to a superior manager.

All of these exchanges take place using the SNMPv2 protocol, which is a simple request/response type of protocol. Typically, SNMPv2 is implemented on top of the user datagram protocol (UDP), which is part of the TCP/IP suite. Because SNMPv2 exchanges are in the nature of discrete request-response pairs, an ongoing reliable connection is not required.

Structure of Management Information The structure of management information (SMI) defines the general framework within which a MIB can be defined and constructed. The SMI identifies the data types that can be used in the MIB, and how resources within the MIB are represented and named. The philosophy behind SMI is to encourage simplicity and extensibility within the MIB. Thus, the MIB can store only simple data types: scalars and two-dimensional arrays of scalars, called tables. The SMI does not support the creation or retrieval of complex data structures. This philosophy is in contrast to that used with OSI systems management, which provides for complex data structures and retrieval modes to support greater functionality. SMI avoids complex data types and structures to simplify the task of implementation and to enhance interoperability. MIBs will inevitably contain vendor-created data types and, unless tight restrictions are placed on the definition of such data types, interoperability will suffer.

There are three key elements in the SMI specification. At the lowest level, the SMI specifies the data types that may be stored. Then the SMI specifies a formal technique for defining objects and tables of objects. Finally, the SMI provides a scheme for associating a unique identifier with each actual object in a system, so that data at an agent can be referenced by a manager.

Table 20.2 shows the data types that are allowed by the SMI. This is a fairly restricted set of types. For example, real numbers are not supported. However, it is rich enough to support most network management requirements.

Protocol Operation The heart of the SNMPv2 framework is the protocol itself. The protocol provides a straightforward, basic mechanism for the exchange of management information between manager and agent.

The basic unit of exchange is the message, which consists of an outer message wrapper and an inner protocol data unit (PDU). The outer message header deals with security and is discussed later in this section.

Table 20.2 Allowable Data Types in SNMPv2

Data Type	Description
INTEGER	Integers in the range of -2^{31} to $2^{31} - 1$.
UInteger32	Integers in the range of 0 to $2^{32} - 1$.
Counter32	A nonnegative integer that may be incremented modulo 2^{32}.
Counter64	A nonnegative integer that may be incremented modulo 2^{64}.
Gauge32	A nonnegative integer that may increase or decrease, but shall not exceed a maximum value. The maximum value can not be greater than $2^{32} - 1$.
TimeTicks	A nonnegative integer that represents the time, modulo 2^{32}, in hundredths of a second.
OCTET STRING	Octet strings for arbitrary binary or textual data; may be limited to 255 octets.
IpAddress	A 32-bit internet address.
Opaque	An arbitrary bit field.
BIT STRING	An enumeration of named bits.
OBJECT IDENTIFIER	Administratively assigned name to object or other standardized element. Value is a sequence of up to 128 nonnegative integers.

Seven types of PDUs may be carried in an SNMP message. The general formats for these are illustrated informally in Figure 20.6 Several fields are common to a number of PDUs. The Request-ID field is an integer assigned such that each outstanding request can be uniquely identified. This enables a manager to correlate incoming responses with outstanding requests. It also enables an agent to cope with duplicate PDUs generated by an unreliable transport service. The Variable-Bindings field contains a list of object identifiers; depending on the PDU, the list may also include a value for each object.

The GetRequest-PDU, issued by a manager, includes a list of one or more object names for which values are requested. If the get operation is successful, then the

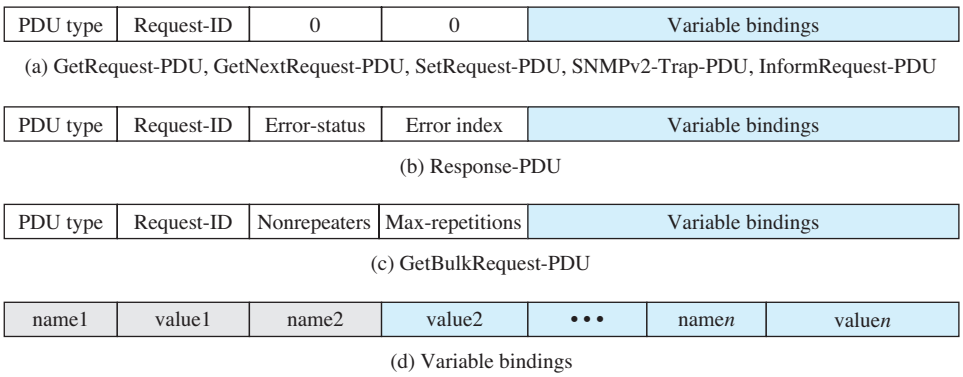

Figure 20.6 SNMPv2 PDU Format

responding agent will send a Response-PDU. The variable-bindings list will contain the identifier and value of all retrieved objects. For any variables that are not in the relevant MIB view, its identifier and an error code are returned in the variable-bindings list. Thus, SNMPv2 permits partial responses to a GetRequest, which is a significant improvement over SNMP. In SNMP, if one or more of the variables in a GetRequest is not supported, the agent returns an error message with a status of no-SuchName. To cope with such an error, the SNMP manager must either return no values to the requesting application, or it must include an algorithm that responds to an error by removing the missing variables, resending the request, and then sending a partial result to the application.

The GetNextRequest-PDU also is issued by a manager and includes a list of one or more objects. In this case, for each object named in the Variable-Bindings field, a value is to be returned for the object that is next in lexicographic order, which is equivalent to saying next in the MIB in terms of its position in the tree structure of object identifiers. As with the GetRequest-PDU, the agent will return values for as many variables as possible. One of the strengths of the GetNextRequest-PDU is that it enables a manager entity to discover the structure of a MIB view dynamically. This is useful if the manager does not know a priori the set of objects that are supported by an agent or that are in a particular MIB view.

One of the major enhancements provided in SNMPv2 is the GetBulkRequest PDU. The purpose of this PDU is to minimize the number of protocol exchanges required to retrieve a large amount of management information. The GetBulkRequest PDU allows an SNMPv2 manager to request that the response include as many requested variables as possible given the constraints on message size.

The SetRequest-PDU is issued by a manager to request that the values of one or more objects be altered. The receiving SNMPv2 entity responds with a Response-PDU containing the same Request-ID. The SetRequest operation is atomic: Either all of the variables are updated or none are. If the responding entity is able to set values for all of the variables listed in the incoming variable-bindings list, then the Response-PDU includes the Variable-Bindings field, with a value supplied for each variable. If at least one of the variable values cannot be supplied, then no values are returned, and no values are updated. In the latter case, the error-status code indicates the reason for the failure, and the error-index field indicates the variable in the Variable-Bindings list that caused the failure.

The SNMPv2-Trap-PDU is generated and transmitted by an SNMPv2 entity acting in an agent role when an unusual event occurs. It is used to provide the management station with an asynchronous notification of some significant event. The variable-bindings list is used to contain the information associated with the trap message. Unlike the GetRequest, GetNextRequest, GetBulkRequest, SetRequest, and InformRequest-PDUs, the SNMPv2-Trap-PDU does not elicit a response from the receiving entity; it is an unconfirmed message.

The InformRequest-PDU is sent by an SNMPv2 entity acting in a manager role, on behalf of an application, to another SNMPv2 entity acting in a manager role, to provide management information to an application using the latter entity. As with the SNMPv2-Trap-PDU, the Variable-Bindings field is used to convey the associated information. The manager receiving an InformRequest acknowledges receipt with a Response-PDU.

For both the SNMPv2-Trap and the InformRequest, various conditions can be defined that indicate when the notification is generated; the information to be sent is also specified.

Simple Network Management Protocol Version 3 (SNMPv3)

Many of the functional deficiencies of SNMP were addressed in SNMPv2. To correct the security deficiencies of SNMPv1/v2, SNMPv3 was issued as a set of Proposed Standards in January 1998 (currently RFCs 3410 through 3415). This set of documents does not provide a complete SNMP capability but rather defines an overall SNMP architecture and a set of security capabilities. These are intended to be used with the existing SNMPv2 or with SNMPv1.

SNMPv3 provides three important services: authentication, privacy, and access control. The first two are part of the User-Based Security Model (USM), and the last is defined in the View-Based Access Control Model (VACM). Security services are governed by the identity of the user requesting the service; this identity is expressed as a principal, which may be an individual or an application or a group of individuals or applications.

The authentication mechanism in USM assures that a received message was transmitted by the principal whose identifier appears as the source in the message header. This mechanism also assures that the message has not been altered in transit and has not been artificially delayed or replayed. The sending principal provides authentication by including a message authentication code with the SNMP message it is sending. This code is a function of the contents of the message, the identity of the sending and receiving parties, the time of transmission, and a secret key that should be known only to sender and receiver. The secret key must be set up outside of USM as a configuration function. That is, the configuration manager or network manager is responsible for distributing secret keys to be loaded into the databases of the various SNMP managers and agents. This can be done manually or using some form of secure data transfer outside of USM. When the receiving principal gets the message, it uses the same secret key to calculate the message authentication code once again. If the receiver's version of the code matches the value appended to the incoming message, then the receiver knows that the message can only have originated from the authorized manager and that the message was not altered in transit. The shared secret key between sending and receiving parties must be preconfigured. The actual authentication code used is known as HMAC, which is an Internet-standard authentication mechanism.

The privacy facility of USM enables managers and agents to encrypt messages. Again, manager principal and agent principal must share a secret key. In this case, if the two are configured to use the privacy facility, all traffic between them is encrypted using the Data Encryption Standard (DES). The sending principal encrypts the message using the DES algorithm and its secret key and sends the message to the receiving principal, which decrypts it using the DES algorithm and the same secret key.

The access control facility makes it possible to configure agents to provide different levels of access to the agent's MIB to different managers. An agent principal can restrict access to its MIB for a particular manager principal in two ways. First, it

can restrict access to a certain portion of its MIB. For example, an agent may restrict most manager parties to viewing performance-related statistics and only allow a single designated manager principal to view and update configuration parameters. Second, the agent can limit the operations that a manager can use on that portion of the MIB. For example, a particular manager principal could be limited to read-only access to a portion of an agent's MIB. The access control policy to be used by an agent for each manager must be preconfigured and essentially consists of a table that detail the access privileges of the various authorized managers.

APPLICATION NOTE

How Much Management?

With today's communication software and hardware, there is a tremendous amount of management capability built into each device. In addition, you can install protocols for the express purpose of more management. As an example, an organization may decide to run the Simple Network Management Protocol or Cisco Discovery Protocol in order to solicit information and control network devices. While it is clear the some level of administration is vital, it is easy to get carried away and spend the day "supervising" devices that are working just fine.

There are literally dozens of management "items." The following is a brief list of the information that can be reported from most networking devices:

- Track statistics for the network
- Keep track of status of machinery
- Enable/disable ports
- Graphically display information (ports)
- Maintain security (users, login, blocking traffic from unknown device)
- Provide resilience
- Handle messaging/polling
- Implement software upgrades
- Handle traffic/topology control
- Configure IP address
- Set to defaults
- Monitor other devices
- Send messages (to monitor)
- View faults

In addition to being able to determine a great deal regarding the network, this information can be sent to the administrator via e-mail, SNMP messages, pagers, and cell phones. A common mistake is to set the management parameters and alarm limits prior to understanding the normal operation of the network. Baselining is the information you gather

regarding the standard values for network behavior. It is the inexperienced manager that decides to be notified of any and all network events. As an example, we might set an alarm to e-mail us when the number of frames transmitted in error on the wireless network exceeds 100 when the normal level of retransmissions is 1000. This results in a lot of automatically generated e-mail.

With small business and home networks, the amount of management required is usually very small. However, there are instances where a certain amount of configuration and notification may be desired. These instances include initial setup, security alerts, and network-specific items such as billing or authorization. At home we may have additional requirements, such as parental monitoring for children using the Web, spyware/antivirus programs, and software configuration for browsers. In either case, the level of involvement is usually small and there is not much call for expensive servers or days spent configuring the management system.

As networks grow, decisions need to be made regarding the appropriate level of sophistication of the Network Management System (NMS). Remember, the time you spend managing the network is time away from other tasks. The NMS can also degrade network performance. Earlier in this book, the concepts of in-band and out-of-band control were introduced. Most enterprise devices are initially configured via a console or serial link. This is a separate physical pathway for the management traffic and is called out-of-band management. Once the administrator starts using telnet, ssh, http, or SNMP to run the network, the messages compete with standard data traffic. This is called inband management. There are some instances where the topology or configuration can effectively disable a portion of the network because of management-inspired messaging. For example, a suspect link can be told to mirror all of its traffic to another location for analysis. Once this is done, the link receiving the new information can be oversubscribed if there are other processes running. Lastly, network failures may cause the connection to the management devices to be severed.

Management of computing resources and a network can be costly and time consuming. Keys to running successful management schemes include appreciating the normal operating conditions of the network, implementing only those management items that are required, understanding the conditions that will result in alarms, knowing the effect of implementing your NMS, and only configuring alerts for items you really want to know about.

20.4 RECOMMENDED READING AND WEB SITES

[STAL99] provides a comprehensive and detailed examination of SNMP, SNMPv2, and SNMPv3; the book also provides an overview of network management technology. One of the few textbooks on the subject of network management is [SUBR00].

STAL99 Stallings, W. *SNMP, SNMPv2, SNMPv3, and RMON 1 and 2.* Reading, MA: Addison-Wesley, 1999.

SUBR00 Subranamian, M. *Network Management: Principles and Practice.* Reading, MA: Addison-Wesley, 2000.

Recommended Web site:

- **Simple Web Site:** Maintained by the University of Twente. It is a good source of information on SNMP, including pointers to many public-domain implementations and lists of books and articles.

20.5 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

Key Terms

accounting management	management station	Simple Network Management Protocol (SNMP)
agent	manager	Structure of Management Information (SMI)
configuration and name management	network management	
fault	network management protocol	
fault management	network management system	
management information base (MIB)	performance management	
	security management	

Review Questions

20.1 List and briefly define the key areas that comprise network management.

20.2 Define *fault* as it applies to network management.

20.3 List two ways in which a network management system may be characterized as integrated.

20.4 List and briefly define the key elements of SNMP.

20.5 What functions are provided by SNMP?

20.6 What lower-layer protocol encapsulates SNMP messages?

20.7 Describe two different interpretations of the term *MIB*.

20.8 What are the differences among SNMPv1, SNMPv2, and SNMPv3?

Problems

20.1 Because SNMP uses two different port numbers (UDP ports 161 and 162), a single system can easily run both a manager and an agent. What would happen if the same port number were used for both?

20.2 The original (version 1) specification of SNMP has the following definition of a new type:

Gauge :: = [APPLICATION 2] IMPLICIT INTEGER (0..4294967295)

The standard includes the following explanation of the semantics of this type:

> This application-wide type represents a non-negative integer, which may increase or decrease, but which latches at a maximum value. This standard specifies a maximum value of $2^{32}-1$ (4294967295 decimal) for gauges.

Unfortunately, the word *latch* is not defined, and this has resulted in two different interpretations. The SNMPv2 standard cleared up the ambiguity with the following definition:

> The value of a Gauge has its maximum value whenever the information being modeled is greater than or equal to that maximum value; if the information being modeled subsequently decreases below the maximum value, the Gauge also decreases.

a. What is the alternative interpretation?

b. Discuss the pros and cons of the two interpretations.

20.3 One of the first steps in configuring a device to be managed is to give it an IP address. Why?

20.4 Many network administrators use the ping program as a primary management tool.

a. Why would you ping a network device?

b. Why would you ping yourself?

20.5 We have seen that SNMP uses UDP as its transport protocol. Why was UDP chosen over TCP?

20.6 What is the disadvantage of having the network management system operate at the application layer?