Week 8: Network Management

This is the last seminar of this unit. You have the least amount of pages to read this time: not so simple but (I hope it will be) enjoyable. Network management is a big subject that requires a whole course to cover. In this seminar, we will only cover an overview of the network management, entities involved, and standards. A reader gets a sense of what goes on while managing a network, and learns which tools and mechanisms are used to aid the system manager. Sorry if this subject feels so dry, because this how it is.

As we all know, the Internet is becoming increasingly an everyday necessity for many people, including people living in developing countries. Large number people already depend on the Internet in their daily life. Thus, like electricity, many cannot afford to lose their Internet connection for a lengthy period of time. Luckily, this need has been recognized so early and tools have been devised to predict errors before they happen; and when they do happen, procedures for fast recovery (which automatic in some cases) are also provided.

Several scenarios for what could go wrong, and what a manager can do to handle these problems are listed in the book. For example: Failure of an interface card, host failure, overloaded links, router misconfiguration, deterioration in quality of service, and network intrusion. Accordingly, the manager should detect the problem before it happens and take care of it, or at least suggest a solution.

Network management is best defined by the following long statement, which is taken from the text: *"Network management includes the deployment, integration, and coordination of the hardware, software, and human element to monitor, test, poll, configure analyze, evaluate, and control the network and element resources to meet the real-time, operational performance, and quality of service requirements at a reasonable cost".* Detailed explanation of this sentence is the subject of this seminar.

Networks typically have one manager located at one place, and a large number of network components (hosts, routers, bridges, links, and etc) scattered around a wide geographic area. To effectively manage this network, every component must periodically report its status to the manager. To report status, components have running **agents** (hardware and software) that are in touch with the **managing entity** located at manager's host (see Figure 9.2 in the textbook). Clearly, management agents should share protocols and standards with the managing entity to report status. There is a set of standards available for this purpose, notably the **SNMP** that will be studied later in this seminar.

The **Internet network-management framework** defines the working environment of the process of network management. Managing entity and agents share a set of rules to coordinate the management process. First, managing entity and agents agree on the parameters and qualities to monitor. Second, they agree on the control messages that the managing entity can send to the agent. Third, they agree on the format of exchanged information. Finally, they share a common protocol that governs the communication between them.

Management Information Base, **MIB,** is the set of network management objects that can be queried for status or sent a control message to perform an action at the agent's side. For example, it is possible to query certain MIB objects about number IP datagrams discarded, the number of Ethernet collisions, routing paths, and etc. There are a large number of standard MIB objects, in addition to many more non-standard (vendor-specific) ones. Related MIB objects are grouped in **MIB modules.**

The language that defines data format between managing entity and the agents is called **SMI.** SMI has a list of simple (basic) predefined data types (see Table 9.1 in the textbook), and complex types (Object type, and module entity). Basic data types are similar to the ones used in programming languages; however they are specialized for effective storage and use. Complex

types are aggregate types that have specific fields specialized to report status, as defined by MIB. A MIB module, on the other hand, is a collection of related MIB objects (see text's examples). The IETF standard body has worked on standardizing MIB modules associated with routers, hosts, and other network equipment. The product of this standard is the famous SNMP protocol. SNMP was created in modular form to facilitate future integration with ISO standards, an integration that never happened.

It might be useful to learn the naming and numbering conventions of ISO network management standards. Figure 9.3 in the textbook exhibits object identifier tree. In this tree every element is associated with a name and a number. Standard numeric identifiers are formed by following branches in this tree. For example, the standard (1.3.6.1.2.1) refer to the MIB-2 of the object tree (start at ISO (1), then move in the tree to ISO identified organization (3), and US DoD (6), …, MIB-2 (1)).

The Simple Network-Management Protocol, **SNMP**, is used to convey MIB information among managing entities and agents executing on behalf of these entities. Table 9.4 in the textbook lists SNMPv2 protocol data units (PDU) types. For example, the `GetRequest, GetNextRequest, and GetBulkRequest` PDUs are sent by managing entities to agents requesting status information. These requests differ in the size and the order of requested information. Agents respond with the `Response` PDU carrying the requested information. Another example is the `SetRequest`, which is initiated by managing entities to set value of one or more MIB object instance; agents respond with '`noError`' Error Status. The `SNMPv2-Trap` is an asynchronous PDU that is initiated by agents without a managing entity request. Clearly, this is an emergency PDU to attract the attention of the managing entity to an exceptional event (interface failure, for example).

SNMPv3 is an improved SNMPv2, with security features. Remember that SNMP has the power to set MIB object values. Consequently, an intruder-injected SNMP `SetRequest` PDUs could reck havoc in the network. SNMPv3 engine and applications are illustrated in Figure 9.5 in the textbook. The managing entity has command generator, notification receiver and proxy forwarder. Meanwhile, the agent has the similar blocks in the second line of or Figure 9.5 in the textbook. To provide security, PDUs generated by SNMPv3 application are passed through a SNMP engine, which adds a security header to the PDU. SNMPv3 security includes: encryption, authentication, protection against playback, and access control:

- Encryption is done using DES algorithm that we studied earlier.
- Authentication is done with a slight modification to the method we learned in last seminar. A sender attaches a secret key (other than the public or private encryption keys) to the message, encrypts the message together with the key, and then sends it to the receiver. The receiver is thus sure that the message is authentic because no one else is supposed to know the secret key other than the real partner.
- Protection against play back is done through adding a new number to every new message, that is, similar messages at different times are attached with different numbers. Relying on the expected number, managing entities and agents can determine whether a message is a playback of an older message, or not.
- SNMPv3 provides tools for access control, which limits network management information available to be queried or set by a user.

ASN.1 is an ISO originated standard that deals, among other things, with the data presentation translation between different machines (hardware). Different computer architectures adopt different internal data representation (big-endian and little-endian formats, for example). Hence, data should be put in a commonly known format before being moved from one machine to another. ANS.1 (so is SMI) have rules to describe data in a way that is interpretable by receivers. Consider the example of Figure 9.9 in the textbook, the sender wants to send the character stream "smith" and the number 259. Observe in the figure that there are two numbers preceding

the string “smith”. The first number, 4, represents the type of data to follow, which is Octet String according to table 9.5 in the textbook. The second number that precedes the data stream is 5, which represents the number of characters in the stream. When the string smith is through, the second data stream, the number 259 (encoded) follows in the same manner with two numbers ahead indicating: a) the type 2 for Integer and b) 2 for two bytes. The rules used in this encoding are called **Basic Encoding Rules (BER).**

Reading assignment

Read chapter 9 all sections.