Lecture 7.2

Application Layer: SNMP

Dr. Vandana Kushwaha

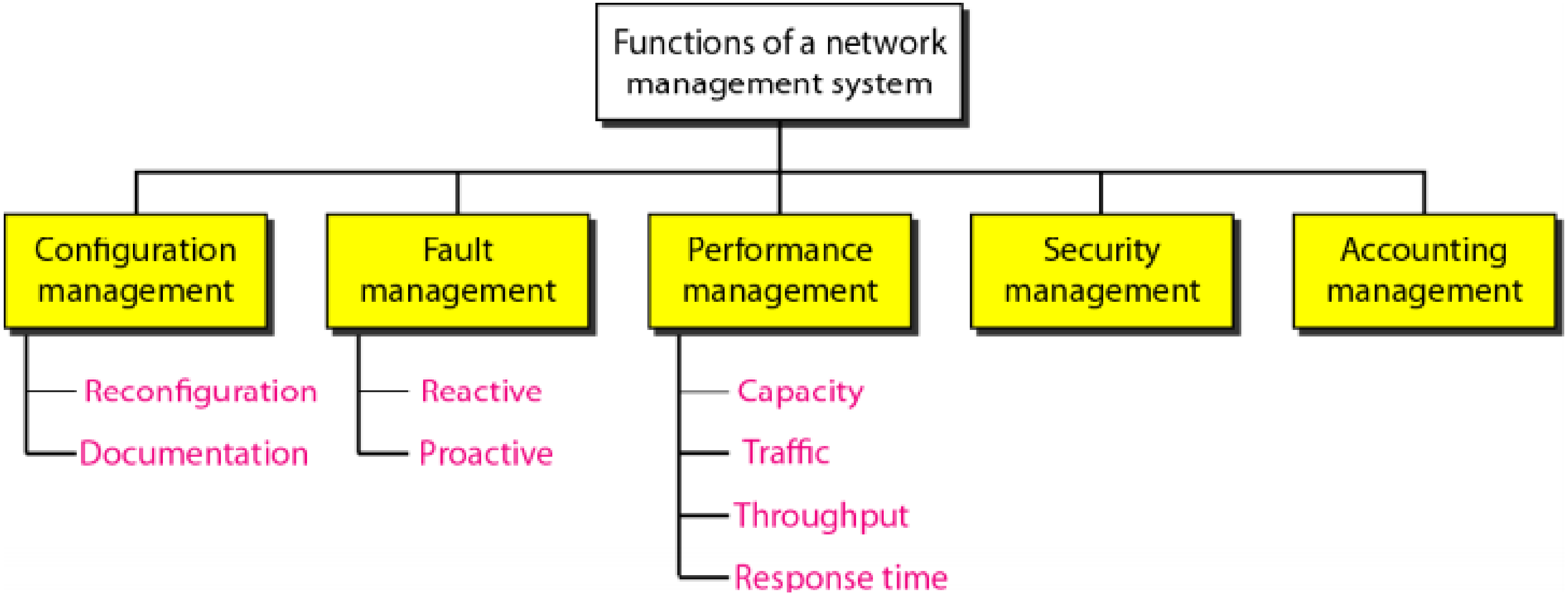
Department of Computer Science

Institute of Science, BHU, Varanasi

Network Management

- We can define **network management as monitoring, testing, configuring, and trouble shooting 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 **functions performed by a network management system** can be divided into **five broad categories**:
    1. ***Configuration management,***
    2. ***Fault management,***
    3. ***Performance management,***
    4. ***Security management,***
    5. ***Accounting management,***

Functions of a network management system



1.Configuration Management

- A **large network** is usually made up of **hundreds/thousands** of **entities** that are **physically** or **logically connected** to one another.
- 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** .
- **Configuration management** can be divided into **two subsystems**:
  - **Reconfiguration: Hardware** reconfiguration, **Software** reconfiguration and **User-account** reconfiguration
  - **Documentation: original network configuration** and each subsequent **change must be recorded** meticulously.

2. Fault Management

- **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***
  - ***Proactive fault management***.
- A **reactive fault management** system is **responsible** for **detecting, isolating, correcting,** and **recording faults.**
- **Proactive fault management** tries to **prevent faults** from **occurring.**

Performance Management & Security Management

3. 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** by using some **measurable** quantity such as **capacity, traffic, throughput, or response time**.

4. Security Management

- **Security management** is responsible for **controlling access to the network** based on the **predefined policy.**

5. Accounting Management

- **Accounting management** is the **control 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.**
- 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-term** and **long-term planning** based on the **demand** for **network use.**

SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

- **Simple Network Management Protocol (SNMP)** is a widely used protocol designed to **facilitate** the **management** of **networked devices** from a **central location.**
- The **Simple Network Management Protocol (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** a **network.**
- **SNMP** uses the **services** of **UDP** on two **well-known ports, 161** and **162**.
- The well known **port 161** is used by the **server (agent),** and the well-known **port 162** is used by the **client (manager).**
- **Devices** that typically support **SNMP** include **routers, switches, servers, workstations, printers, modem racks, and more.**
- **SNMP** is an **application-level protocol** in which a **few manager stations control** a **set of agents.**

SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

- The **SNMP protocol** is designed at the **application level** so that it can **monitor devices made by different manufacturers** and **installed** on **different physical networks.**
- In other words, **SNMP** frees **management tasks** from both the **physical characteristics** of the **managed devices** and the underlying **networking technology.**
- It can be used in a **heterogeneous internet** made of **different LANs** and **WANs** connected by **routers** made by **different manufacturers.**

SNMP Architecture

The **SNMP architecture** is composed of **three major elements**:

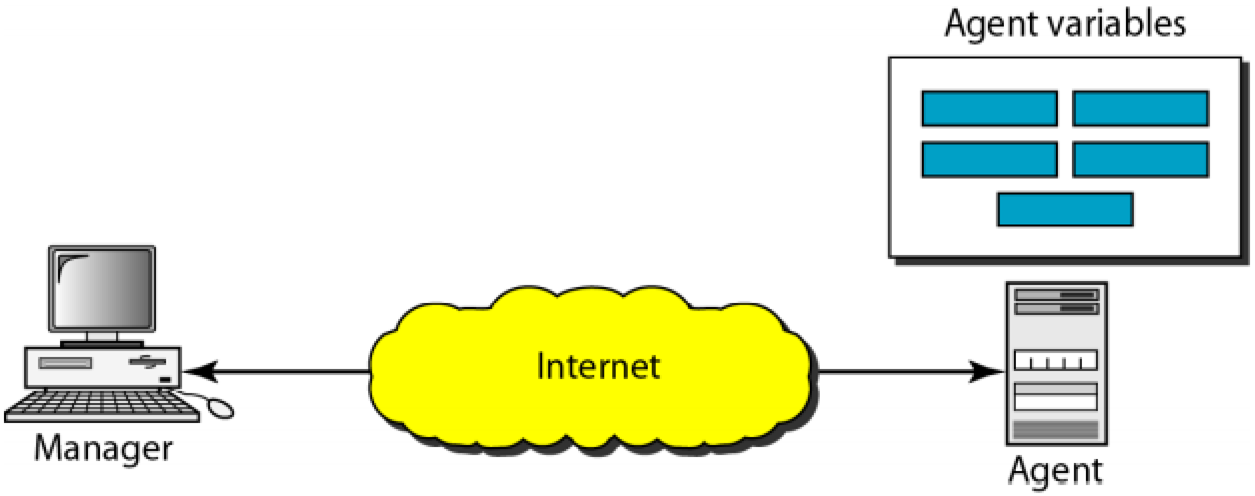
1. ***Managers***
2. ***Agents***
3. ***MIBs*** *(Management Information Base)*

- 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** and the **manager** has **access** to the **values** in the **database.**

SNMP Architecture

- For **example**, a **router(Agent)** 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.**
- **Manager** simply **sends** a **packet** to **force a 0** value in the **counter**.

SNMP Architecture



A PC running the **SNMP client** program

Router/ Switch/ Network Printer running the **SNM Server** program

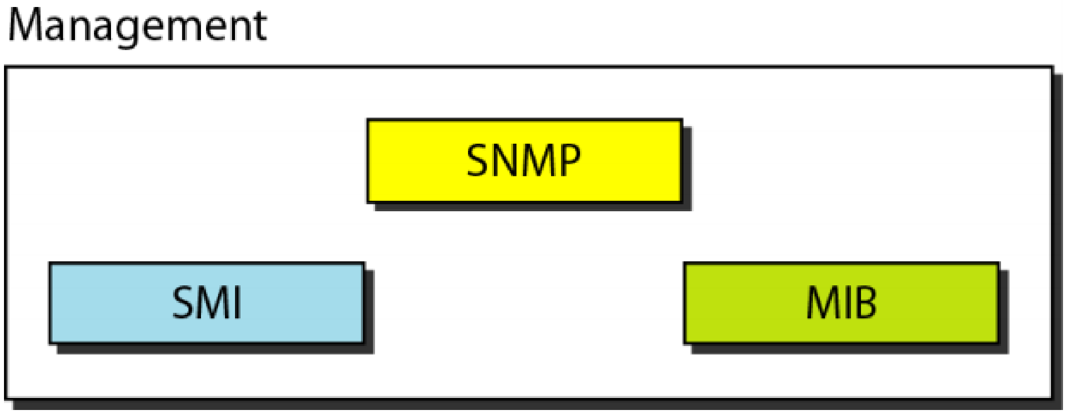
SNMP Architecture

- **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:**
  1. ***Structure of Management Information (SMI)***
  2. ***Management Information Base (MIB).***
- **Management** on the **Internet** is done through the **cooperation** of the **three protocols** **SNMP, SMI**, and **MIB,** as shown in **Figure** below:



Role of SNMP

- **SNMP** has some **very specific roles** in **network management.**
- It **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** based on the **responses** of **agents.**
- The **packets** **exchanged between a** **manager** **and an** **agent** contain the **object (variable) names** and their **status (values).**
- **SNMP** is responsible for **reading** and **changing** **these values**.

Roles of SMI

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

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

- **SMI** is a **guideline** for **SNMP**.

Roles of MIB

- For each **entity** to be **managed,** **MIB** must **define** the **number of objects**, **name them** according to the **rules defined** by **SMI,** and **associate a type to each named object .**
- **MIB creates** a **collection** of **named objects**, **their types**, and **their relationships to each other** in an entity to be managed.
- Each **agent** has its **own MIB**, which is **a collection** of all the **objects** that the **manager can manage.**
- The **objects** in **MIB** are **categorized** under **10** different **groups**: system, interface, address translation, ip, icmp, tcp, udp, bgp, transmission, and snmp.
- **SNMP** stores, changes, and interprets the values of **objects** already declared by **MIB** according to the **rules** defined by **SMI**.

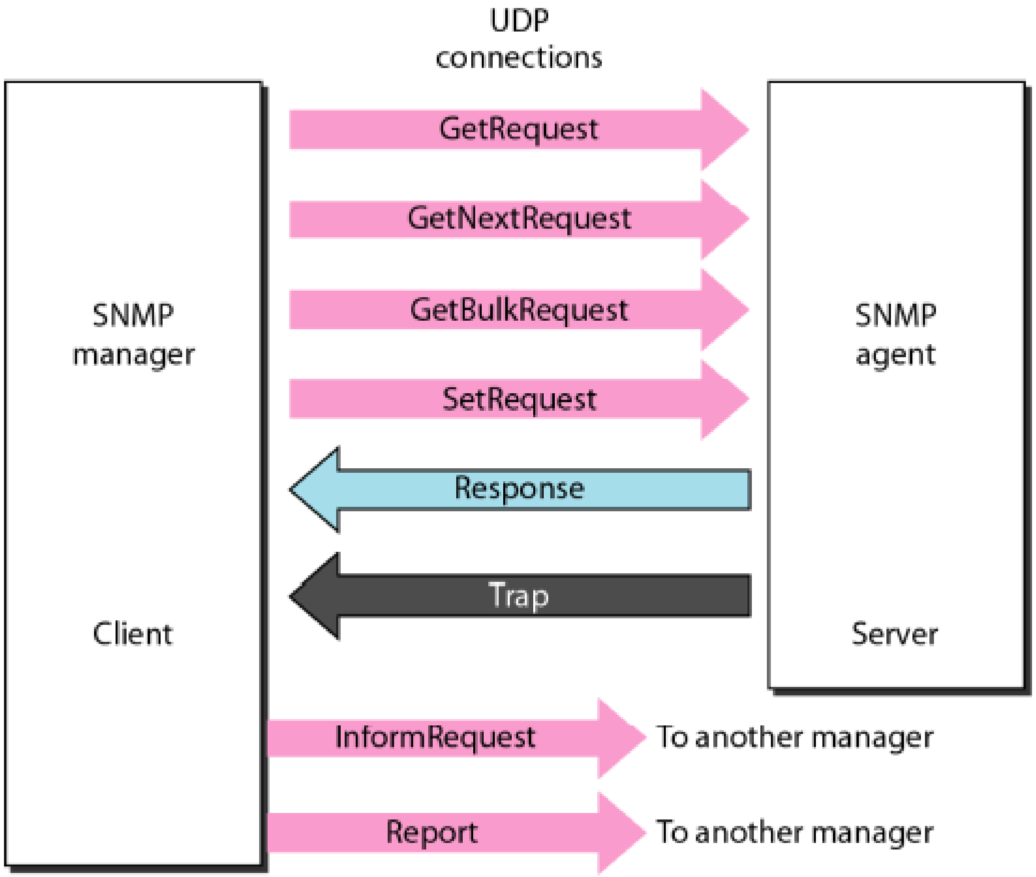
Network Management Analogy

- We can compare the task of **Network Management** to the task of **writing a program.**
- **Both** tasks **need rules**.
- In **network management** this is handled by **SMI.**
- Both tasks need **variable declarations**.
- In **network management** this is handled by **MIB.**
- Both tasks have **actions performed** by statements.
- In **network management** this is handled by **SNMP.**

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.**
- **MIB** is responsible for **finding** the **object** that **holds the number of the 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.

SNMP Message Exchange



SNMP Message Exchange

- **GET Request**
- **Manager → Agent**
- Used to **retrieve** the value of a variable from the MIB.
- **Example:**

  Manager asks: “What is the CPU usage?”

- **GET-BULK (SNMPv2 only)**
- **Manager → Agent**
- Efficiently retrieves **large blocks of data** (e.g., routing tables).

SNMP Message Exchange

- **SET Request**
- **Manager → Agent**
- Changes the value of a MIB object.
- **Example:** Set an **interface** administratively **down/up.**

- **RESPONSE**
- **Agent → Manager**
- Sent in reply to:
  - GET
  - GET-NEXT
  - GET-BULK
  - SET
- Contains the requested values or error status.

SNMP Message Exchange

- **TRAP**
- **Agent → Manager**
- Sent automatically when something important happens:
  - Sent when a specific event occurs
  - Manager does not acknowledge TRAPs
  - **Example**: link down, device reboot

- **INFORM (SNMPv2/3)**
- **Agent → Manager**
- Similar to trap, but **requires acknowledgment**.
- Reliable version of **Trap.**