Lecture 7:

SNMP Management: 
Organization and Information Models

Prof. Shervin Shirmohammadi
SITE, University of Ottawa
Overview

• SNMP is the most widely used network management protocol.
• When we say SNMP management, we are really referring to Internet management. (SNMP itself is a com protocol).
• SNMP management model:
  – Organization Model
    • Relationship between network element, agent, and manager
    • Hierarchical architecture
  – Information Model
    • Uses ASN.1 syntax
    • SMI (Structure of Management Information
    • MIB (Management Information Base)
  – Communication Model
    • Transfer syntax
    • SNMP over TCP/IP
    • Communication services addressed by messages
    • Security framework community-based model
  – Functional Model
    • Fault
    • Performance
    • Configuration
    • Accounting
    • Security
Managed LAN

NMS on subnet 192.168.252.1 manages the router and the hubs on subnet 172.16.46.1 across the backbone network.

Figure 4.1 A Managed LAN Network
Managed Hubs: System Information

• Information obtained querying the hubs

• Data truly reflects what is stored in the hub

Title: System Information: 172.16.46.2
Name or IP Address: 172.16.46.2

System Name:
System Description: 3Com LinkBuilder FMS, SW version: 3.02
System Contact:
System Location:
System Object ID: .iso.org.dod.internet.private.enterprises.43.1.8.5
System Up Time: (2475380437) 286 days, 12:03:24.37

Title: System Information: 172.16.46.3
Name or IP Address: 172.16.46.3

System Name:
System Description: 3Com LinkBuilder FMS, SW version: 3.12
System Contact:
System Location:
System Object ID: .iso.org.dod.internet.private.enterprises.43.1.8.5
System Up Time: (3146735182) 364 days, 12:55:51.82
Managed Router: System Information

Title: System Information: router1.gatech.edu
Name or IP Address: 172.16.252.1

System Name: router1.gatech.edu
System Description: Cisco Internetwork Operating System Software
  : IOS (tm) 7000 Software (C7000-JS-M), Version 11.2(6),RELEASE SOFTWARE (ge1)
  : Copyright (c) 1986-1997 by Cisco Systems, Inc.
  : Compiled Tue 06-May-97 19:11 by kuong

System Contact:
System Location:
System Object ID: iso.org.dod.internet.private.enterprises.cisco.ciscoProducts.cisco 7000
System Up Time: (315131795) 36 days, 11:21:57.95
Managed Hub: Port Addresses

- Information acquired by the NMS on hub interfaces
- Index refers to the interface on the hub
- Link address is the MAC address
- The second row data is a serial link

IP address: 172.16.46.2

<table>
<thead>
<tr>
<th>Index</th>
<th>Interface</th>
<th>IP address</th>
<th>Network Mask</th>
<th>Network Address</th>
<th>Link Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3Com</td>
<td>172.16.46.2</td>
<td>255.255.255.0</td>
<td>172.16.46.0</td>
<td>0x08004E07C25C</td>
</tr>
<tr>
<td>2</td>
<td>3Com</td>
<td>192.168.101.1</td>
<td>255.255.255.0</td>
<td>192.168.101.0</td>
<td>&lt;none&gt;</td>
</tr>
</tbody>
</table>
## Managed Router: Port Addresses

- Information acquired by NMS on the router interfaces
- Index refers to the interface on the router
- LEC is the LAN emulation card
- Ethernet 2/0 interface refers to the interface card 2 and port 0 in that card

<table>
<thead>
<tr>
<th>Index</th>
<th>Interface</th>
<th>IP address</th>
<th>Network Mask</th>
<th>Network Address</th>
<th>Link Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>LEC.1.0</td>
<td>192.168.3.1</td>
<td>255.255.255.0</td>
<td>192.168.3.0</td>
<td>0x00000C3920B4</td>
</tr>
<tr>
<td>25</td>
<td>LEC.3.9</td>
<td>192.168.252.1</td>
<td>255.255.255.0</td>
<td>192.168.252.0</td>
<td>0x00000C3920B4</td>
</tr>
<tr>
<td>13</td>
<td>Ethernet2/0</td>
<td>172.16..46.1</td>
<td>255.255.255.0</td>
<td>172.16..46.0</td>
<td>0x00000C3920AC</td>
</tr>
<tr>
<td>16</td>
<td>Ethernet2/3</td>
<td>172.16.49.1</td>
<td>255.255.255.0</td>
<td>172.16.49.0</td>
<td>0x00000C3920AF</td>
</tr>
<tr>
<td>17</td>
<td>Ethernet2/4</td>
<td>172.16.52.1</td>
<td>255.255.255.0</td>
<td>172.16.52.0</td>
<td>0x00000C3920B0</td>
</tr>
<tr>
<td>9</td>
<td>Ethernet1/2</td>
<td>172.16.55.1</td>
<td>255.255.255.0</td>
<td>172.16.55.0</td>
<td>0x00000C3920A6</td>
</tr>
<tr>
<td>2</td>
<td>Ethernet 0/1</td>
<td>172.16.56.1</td>
<td>255.255.255.0</td>
<td>172.16.56.0</td>
<td>0x00000C3920D</td>
</tr>
<tr>
<td>15</td>
<td>Ethernet2/2</td>
<td>172.16.57.1</td>
<td>255.255.255.0</td>
<td>172.16.57.0</td>
<td>0x00000C3920AE</td>
</tr>
<tr>
<td>8</td>
<td>Ethernet1/1</td>
<td>172.16.58.1</td>
<td>255.255.255.0</td>
<td>172.16.58.0</td>
<td>0x00000C3920A5</td>
</tr>
<tr>
<td>14</td>
<td>Ethernet2/1</td>
<td>172.16.60.1</td>
<td>255.255.255.0</td>
<td>172.16.60.0</td>
<td>0x00000C3920AD</td>
</tr>
</tbody>
</table>
Internet Management History

• 1970s: Advanced Research Project Agency Network (ARPANET) comes up with Internet control Message Protocol (ICMP)

• Internet Engineering Task Force (IETF)
  – 1990 SNMPv1
  – 1995 SNMPv2
  – 1998 SNMPv3

• Internet documents:
  – Request for Comments (RFC)
  – IETF STD Internet Standard
  – FYI For your information
SNMP Related RFCs

Figure 4.4 SNMP Document Evolution

Prof. Shervin Shirmohammadi
CEG 4395
SNMP Organization Model

(a) One Manager - One Agent Model
(b) Multiple Managers - One Agent Model

- **two-tier model**
- **three-tier model**
System Overview

Figure 4.9 SNMP Network Management Architecture

Prof. Shervin Shirmohammadi
CEG 4395

7-11
SNMP Messages

- **Get-Request**
  - Sent by manager requesting data from agent

- **Get-Next-Request**
  - Sent by manager requesting data on the next MO to the one specified

- **Set-Request**
  - Initializes or changes the value of network element

- **Get-Response**
  - Agent responds with data for get and set requests from the manager

- **Trap**
  - Alarm generated by an agent
SNMP Information Model

- Structure of Management Information (SMI)
  - RFC 1155
  - Specifies the defines managed objects
  - Uses ASN.1 in doing so
  - Managed Object
    - Scalar
    - Aggregate or tabular object

- Management Information Base (MIB)
  - RFC 1213
  - Defines the storage and relationship between managed objects

![Figure 4.10 Managed Object : Type and Instance](image)
Multiple Instances of MOs

- All 3 Com hubs of the same version have identical identifier: **same object type**.
- Differences in actual values stored: **various instances**; identified by different IP addresses.

---

**Figure 4.11 Managed Object : Type with Multiple Instances**
Names

• Uniquely defined by
  – **DESCRIPTOR**, and
  – **OBJECT IDENTIFIER**

\[
\text{internet OBJECT IDENTIFIER ::= \{iso(1) standard(3) dod(6) internet(1)\}}
\]
\[
\text{internet OBJECT IDENTIFIER ::= } \{1 3 6 1\}
\]
\[
\text{internet OBJECT IDENTIFIER ::= \{iso standard dod internet \}}
\]
\[
\text{internet OBJECT IDENTIFIER ::= \{ iso standard dod(6) internet(1) \}}
\]
\[
\text{internet OBJECT IDENTIFIER ::= \{ iso(1) standard(3) 6 1 \}}
\]
SNMP ASN.1 Data Type

Figure 4.15 SNMP ASN.1 Data Type
# Primitive Data Types

<table>
<thead>
<tr>
<th>Structure</th>
<th>Data Type</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primitive types</td>
<td>INTEGER</td>
<td>Subtype INTEGER (n1..nN)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Special case: Enumerated INTEGER type</td>
</tr>
<tr>
<td></td>
<td>OCTET STRING</td>
<td>8-bit bytes binary and textual data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Subtypes can be specified by either range or fixed</td>
</tr>
<tr>
<td></td>
<td>OBJECT IDENTIFIER</td>
<td>Object position in MIB</td>
</tr>
<tr>
<td></td>
<td>NULL</td>
<td>Placeholder</td>
</tr>
</tbody>
</table>

- **Enumerated**: special case of INTEGER data type

```plaintext
error-status INTEGER {
    noError(0)
    tooBig(1)
    genErr(5)
    authorizationError(16)
}
```
Defined or Application Data Type

- Based on previously defined data types.
- Defined data types are simple or base types.
- Gauge, in the example below, is based on INTEGER, but with special restriction.

<table>
<thead>
<tr>
<th>Defined types</th>
<th>NetworkAddress</th>
<th>Not used</th>
</tr>
</thead>
<tbody>
<tr>
<td>IpAddress</td>
<td>Dotted decimal IP address</td>
<td></td>
</tr>
<tr>
<td>Counter</td>
<td>Wrap-around, non-negative integer, monotonically increasing, max $2^{32} - 1$</td>
<td></td>
</tr>
<tr>
<td>Gauge</td>
<td>Capped, non-negative integer, increase or decrease</td>
<td></td>
</tr>
<tr>
<td>TimeTicks</td>
<td>Non-negative integer in hundredths of second units</td>
<td></td>
</tr>
<tr>
<td>Opaque</td>
<td>Application-wide arbitrary ASN.1 syntax, double wrapped OCTET STRING</td>
<td></td>
</tr>
</tbody>
</table>
Constructor or Structured Data

- **SEQUENCE**, and **SEQUENCE OF**

<table>
<thead>
<tr>
<th>Object</th>
<th>OBJECT IDENTIFIER</th>
<th>ObjectSyntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ipAdEntAddr</td>
<td>{ipAddrEntry 1}</td>
<td>IpAddress</td>
</tr>
<tr>
<td>2 ipAdEntIfIndex</td>
<td>{ipAddrEntry 2}</td>
<td>INTEGER</td>
</tr>
<tr>
<td>3 ipAdEntNetMask</td>
<td>{ipAddrEntry 3}</td>
<td>IpAddress</td>
</tr>
<tr>
<td>4 ipAdEntBcastAddr</td>
<td>{ipAddrEntry 4}</td>
<td>INTEGER</td>
</tr>
<tr>
<td>5 ipAdEntReasmMaxSize</td>
<td>{ipAddrEntry 5}</td>
<td>INTEGER</td>
</tr>
<tr>
<td>6 ipAddrEntry</td>
<td>{ipAddrTable 1}</td>
<td>SEQUENCE</td>
</tr>
</tbody>
</table>

List: IpAddrEntry ::= 

```plaintext
  SEQUENCE {
    ipAdEntAddr   IpAddress
    ipAdEntIfIndex INTEGER
    ipAdEntNetMask IpAddress
    ipAdEntBcastAddr INTEGER
    ipAdEntReasmMaxSize INTEGER (0..65535)
  }
```

Managed Object IpAddrEntry as a list

<table>
<thead>
<tr>
<th>Object Name</th>
<th>OBJECT IDENTIFIER</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 ipAddrTable</td>
<td>{ip 20}</td>
<td>SEQUENCE OF</td>
</tr>
</tbody>
</table>

Table: IpAddrTable ::= 

```plaintext
  SEQUENCE OF   IpAddrEntry
```

Managed Object IpAddrTable as a table
Encoding

- Basic Encoding Rules (BER)
  - uses Tag, Length, and Value (TLV)

<table>
<thead>
<tr>
<th>Type</th>
<th>Length</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class (7-8th bits)</td>
<td>P/C (6th bit)</td>
<td>Tag Number (1-5th bits)</td>
</tr>
</tbody>
</table>

- SNMP Data Types and Tags

<table>
<thead>
<tr>
<th>Type</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECT IDENTIFIER</td>
<td>UNIVERSAL 6</td>
</tr>
<tr>
<td>SEQUENCE</td>
<td>UNIVERSAL 16</td>
</tr>
<tr>
<td>IpAddress</td>
<td>APPLICATION 0</td>
</tr>
<tr>
<td>Counter</td>
<td>APPLICATION 1</td>
</tr>
<tr>
<td>Gauge</td>
<td>APPLICATION 2</td>
</tr>
<tr>
<td>TimeTicks</td>
<td>APPLICATION 3</td>
</tr>
<tr>
<td>Opaque</td>
<td>APPLICATION 4</td>
</tr>
</tbody>
</table>
Managed Object: Structure

- RFC 1155 requires managed objects to have *textual names*, *syntax*, *definition*, *access*, and *status*.

**Figure 4.17 Specifications for System Description**
Managed Object: Macro

OBJECT-TYPE MACRO ::= 
BEGIN
   TYPE NOTATION ::= “SYNTAX” type(TYPE ObjectSyntax)
   “ACCESS” Access
   “STATUS” Status
   VALUE NOTATION ::= value(VALUE ObjectName)

   Access ::= “read-only” | “write-only” | “not-accessible”
   Status ::= “mandatory” | “optional” | “obsolete”

END

Figure 4.18(a) OBJECT-TYPE Macro [RFC 1155]

sysDescr OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
   “A textual description of the entity. This value should
   include the full name and version identification of the
   system’s hardware type, software operating-system, and
   networking software. It is mandatory that this only contain
   printable ASCII characters.”
::= {system 1 }

Figure 4.18(b) Scalar or Single Instance Macro: sysDescr

[ RFC 1213]
Aggregate Object

- A group of objects
- Also called tabular objects
- Can be represented by a table with
  - Columns of objects
  - Rows of instances
- Example: IP address table
- Consists of objects:
  - IP address
  - Interface
  - Subnet mask (which subnet this address belongs to)
  - Broadcast address (value of l.s.b. in IP broadcast address)
  - Largest IP datagram that can be assembled
- Multiple instances of these objects associated with the node
Example

ipAddrEntry { ipAddrTable 1 }

SYNTAX  IpAddrEntry ::= SEQUENCE {
    ipAdEntAddr   IpAddress,
    ipAdEntIfIndex INTEGER,
    ipAdEntNetMask IpAddress,
    ipAdEntBcastAddr INTEGER,
    ipAdEntReasmMaxSize INTEGER (0..65535)
}

ACCESS  not-accessible
STATUS   mandatory
DESCRIPTION "The addressing information for one of this entity's IP addresses."
INDEX   { ipAdEntAddr }

ipAddrTable {ip 20}

SYNTAX  SEQUENCE OF IpAddrEntry
ACCESS  not-accessible
STATUS   mandatory
DESCRIPTION "The table of addressing information relevant to this entity's IP addresses."
EXAMPLE (...)  

ipAdEntAddr OBJECT-TYPE
SYNTAX  IpAddress
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"The IP address to which this entry's addressing information pertains."
::= { ipAddrEntry 1 }

ipAdEntReasmMaxSize OBJECT-TYPE
SYNTAX  INTEGER (0..65535)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"The size of the largest IP datagram which this entity can re-assemble from incoming IP fragmented datagrams received on this interface."
::= { ipAddrEntry 5 }
Tabular Representation of Aggregate Object

- The objects TABLE T and ENTRY E are objects that are logical objects. They define the grouping and are not accessible.
- Columnar objects are objects that represent the attributes and hence are accessible.
- Each instance of E is a row of columnar objects 1 through 5.
- Multiple instances of E are represented by multiple rows.

Figure 4.22(a) Multiple Instance Managed Object
Multiple Instances

Figure 4.22(b) Example of 5 Columnar Object with 4 Instances (rows)
Multiple Instances Example

ipAddrTable {1.3.6.1.2.1.4.20}
  ipAddrEntry (1)
    ipAdEntAddr (1)
    ipAdEntIfIndex (2)
    ipAdEntNetMask (3)
    ipAdEntBcastAddr (4)
    ipAdEntReasmMaxSize (5)

Columnar object ID of ipAdEntBcastAddr is (1.3.6.1.2.1.4.20.1.4):

iso  org  dod  internet  mgmt  mib  ip  ipAddrTable  ipAddrEntry  ipAdEntBcastAddr
  1  3  6  1  2  1  4  20  1  4

Figure 4.23(a) Columnar objects under ipAddrEntry

Figure 4.23(b) Object instances of ipAddrTable (1.3.6.1.2.1.4.20)

<table>
<thead>
<tr>
<th>Row</th>
<th>ipAdEntAddr</th>
<th>ipAdEntIfIndex</th>
<th>IpAdEntNetMask</th>
<th>IpAdEntBcastAddr</th>
<th>IpAdEntReasmMaxSize</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>123.45.2.1</td>
<td>1</td>
<td>255.255.255.0</td>
<td>0</td>
<td>12000</td>
</tr>
<tr>
<td>2</td>
<td>123.45.3.4</td>
<td>3</td>
<td>255.255.0.0</td>
<td>1</td>
<td>12000</td>
</tr>
<tr>
<td>3</td>
<td>165.8.9.25</td>
<td>2</td>
<td>255.255.255.0</td>
<td>0</td>
<td>10000</td>
</tr>
<tr>
<td>4</td>
<td>9.96.8.138</td>
<td>4</td>
<td>255.255.255.0</td>
<td>0</td>
<td>15000</td>
</tr>
</tbody>
</table>

This ROW is not part of the object
Multiple Instances Example (…)

<table>
<thead>
<tr>
<th>Columnar Object</th>
<th>Row # in (b)</th>
<th>Object Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipAdEntAddr 1.3.6.1.2.1.4.20.1.1</td>
<td>2</td>
<td>{1.3.6.1.2.1.4.20.1.1.123.45.3.4}</td>
</tr>
<tr>
<td>ipAdEntIfIndex 1.3.6.1.2.1.4.20.1.2</td>
<td>3</td>
<td>{1.3.6.1.2.1.4.20.1.2.165.8.9.25}</td>
</tr>
<tr>
<td>ipAdEntBcastAddr 1.3.6.1.2.1.4.20.1.4</td>
<td>1</td>
<td>{1.3.6.1.2.1.4.20.1.4.123.45.2.1}</td>
</tr>
<tr>
<td>IpAdEntReasmMaxSize 1.3.6.1.2.1.4.20.1.5</td>
<td>4</td>
<td>{1.3.6.1.2.1.4.20.1.5.9.96.8.138}</td>
</tr>
</tbody>
</table>

**Figure 4.23(c) Object Id for specific instance**
SMI Definition from RFC1155

• **EXPORTS** identifies the objects that any other module could import

RFC1155-SMI DEFINITIONS ::= BEGIN

EXPORTS -- EVERYTHING
    internet, directory, mgmt, experimental, private, enterprises,
    OBJECT-TYPE, ObjectName, ObjectSyntax, SimpleSyntax,
    ApplicationSyntax, NetworkAddress, IpAddress, Counter, Gauge,
    TimeTicks, Opaque;

    -- the path to the root

internet OBJECT IDENTIFIER ::= { iso org(3) dod(6) 1 }

directory OBJECT IDENTIFIER ::= { internet 1 }
mgmt OBJECT IDENTIFIER ::= { internet 2 }
experimental OBJECT IDENTIFIER ::= { internet 3 }
private OBJECT IDENTIFIER ::= { internet 4 }

enterprises OBJECT IDENTIFIER ::= { private 1 }
SMI Definition from RFC1155 (…)

-- definition of object types

OBJECT-TYPE MACRO ::= BEGIN
  TYPE NOTATION ::= "SYNTAX" type (TYPE ObjectSyntax)
  "ACCESS" Access
  "STATUS" Status
  VALUE NOTATION ::= value (VALUE ObjectName)
  Access ::= "read-only" | "read-write" | "write-only" "not-accessible"
  Status ::= "mandatory" | "optional" | "obsolete"
END

-- names of objects in the MIB

ObjectName ::= OBJECT IDENTIFIER

-- syntax of objects in the MIB

ObjectSyntax ::= CHOICE {
  simple
    SimpleSyntax,
  application-wide
    ApplicationSyntax}
SMI Definition from RFC1155 (...)

SimpleSyntax ::=  
   CHOICE {          
      number          
         INTEGER,     
      string          
         OCTET STRING, 
      object          
         OBJECT IDENTIFIER,  
      empty          
         NULL        
   }

ApplicationSyntax ::=  
   CHOICE {         
      address          
         NetworkAddress, 
      counter          
         Counter,      
      gauge            
         Gauge,       
      ticks            
         TimeTicks,    
      arbitrary        
         Opaque

   -- other application-wide types, as they are defined, will be added here
   }

Prof. Shervin Shirmohammadi  CEG 4395
SMI Definition from RFC1155 (...)

-- application-wide types

NetworkAddress ::= 
    CHOICE {
      internet
        IpAddress
    }

IpAddress ::= 
    [APPLICATION 0] -- in network-byte order
      IMPLICIT OCTET STRING (SIZE (4))

Counter ::= 
    [APPLICATION 1]
      IMPLICIT INTEGER (0..4294967295)

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

TimeTicks ::= 
    [APPLICATION 3]
      IMPLICIT INTEGER (0..4294967295)

Opaque ::= 
    [APPLICATION 4] -- arbitrary ASN.1 value,
      IMPLICIT OCTET STRING -- "double-wrapped"

END