Simple Network Management Protocol - SNMP v1, ASN, MIB, BER
<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td>Computer Networks - Network Management Architectures &amp; Applications</td>
</tr>
<tr>
<td>Week 2</td>
<td>Network Management Standards Architectures &amp; Applications</td>
</tr>
<tr>
<td>Week 3</td>
<td>Simple Network Management Protocol - SNMP v1, ASN, MIB, BER</td>
</tr>
<tr>
<td>Week 4</td>
<td>Network Management Functions - <strong>Fault</strong></td>
</tr>
<tr>
<td>Week 5</td>
<td>Simple Network Management Protocol - SNMP v2 - <strong>Configuration</strong></td>
</tr>
<tr>
<td>Week 6</td>
<td>Network Management Functions - <strong>Accounting</strong></td>
</tr>
<tr>
<td>Week 7</td>
<td>Simple Network Management Protocol - SNMP v3 - <strong>Performance</strong></td>
</tr>
<tr>
<td>Week 8</td>
<td>Network Management Functions - <strong>Security</strong></td>
</tr>
<tr>
<td>Week 9</td>
<td>Midterm</td>
</tr>
<tr>
<td>Week 10</td>
<td>Remote Network Monitoring RMON 1, <strong>SLA</strong></td>
</tr>
<tr>
<td>Week 11</td>
<td>Remote Network Monitoring RMON 2</td>
</tr>
<tr>
<td>Week 12</td>
<td>Management Tools, Systems and Applications</td>
</tr>
<tr>
<td>Week 13</td>
<td>NM Project Presentations</td>
</tr>
<tr>
<td>Week 14</td>
<td>NM Project Presentations</td>
</tr>
<tr>
<td>Week 15</td>
<td>NM Project Presentations</td>
</tr>
</tbody>
</table>
Outline

• Simple Network Management Protocol (SNMP)
• Structure of Management Information (SMI)
• Management Information Bases (MIBs)
• BER (Basic Encoding Rules)
• Related Tools
• Conclusion
SNMP overview: 4 key parts

• Management information base (MIB):
  – distributed information store of network management data

• Structure of Management Information (SMI):
  – data definition language for MIB objects

• SNMP protocol
  – convey manager<->managed object info, commands

• security, administration capabilities
  – major addition in SNMPv3
Network Management Standards

• NM Standards
  – ISO - OSI
    • Common Management Information Protocol (CMIP)
    • Common Management Information Service (CMIS)
  – IETF - Internet
    • Simple Network Management Protocol (SNMP)
  – ITU-T
    • Telecommunications Management Network (TMN)
NM Standards (cont.)

- Differences

<table>
<thead>
<tr>
<th>ISO</th>
<th>IETF</th>
<th>TMN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management should be powerful</td>
<td>Management should be simple</td>
<td>Define management architecture only</td>
</tr>
<tr>
<td>Object Oriented Approach</td>
<td>Variable Oriented Approach</td>
<td>Using OSI protocol (CMIP &amp; CMIS)</td>
</tr>
<tr>
<td>Reliable underlying transport</td>
<td>Unreliable transport mechanisms</td>
<td>Management Information is exchanged out-of-band</td>
</tr>
</tbody>
</table>
History

COPIED:
• Manager-Agent concept
• MIBS
• ASN.1
• TERMINOLOGY

starting as temporary solution
many ideas were copied from OSI

Management framework

OSI introduced new ideas
Real object orientation

OSI Protocol Copied
OSI NM Architecture and Model

Network Management

- Organization Model
- Information Model
- Communication Model
- Functional Model

Manager
Agent
Object
Structure of Management Information (SMI)
Management Information Base (MIB)
Protocol Data Unit (PDU)
Configuration Management (CM)
Fault Management (FM)
Performance Management (PM)
Security Management (SM)
Accounting Management (AM)
SNMP Architecture and Model

- **Organization Model**
  - Same as OSI model

- **Information Model**
  - Same as OSI model

- **Communication Model**
  - Message less complex than OSI
  - Transfer structure (PDU)

- **Functional Model**
  - Application functions
    - Operations
    - Administration
    - Security
TMN Architecture

• Address management of telecommunication networks
• Based on OSI model
• Superstructure on OSI network
• Addresses network, service, and business management
Other Groups

- **DMTF (Distributed Management Task Force)**
  - WBEM: Web Based Enterprise Management
  - CIM: Common Information Model
- **TM-Forum (Tele-Management Forum)**
  - Originally based on OSI
- **OMG (Object Management Group)**
  - Looking at CORBA for management
- **IEEE (Institute of Electrical and Electronics Engineers)**
  - Addresses LAN and MAN management
  - Adopts OSI standards
Simple Network Management Protocol

SNMP
More on SNMP

- Different from ICMP (Internet Control Message Protocol) **Ping** (Packet Internet Grouper)
- **Remote** and **local** management
- Including servers, workstations, routers, switches and other managed devices
- Advantages of using SNMP
  - Standardized
  - Universally supported
  - Allows distributed management access
  - Lightweight protocol
Standard

- **SMI (Structure of Management Information)**
  - Rules specifying the format used to define objects managed that the SNMP protocol accesses
    - SMIv1: RFC 1155
    - SMIv2: RFC 2578

- **MIBs (Management Information Bases)**
  - Tell what management information exists
    - MIB-I: RFC 1156
    - MIB-II: RFC 1213

- **SNMP (Simple Network Management Protocol)**
  - Define how information is exchanged between NMS and Agent
    - SNMPv1: RFC 1157
    - SNMPv2: RFC 1901, 1905, 1906
    - SNMPv3: RFC 2571, 2572, 2573, 2574, 2575
SNMP Versions

- **SNMPv1**
  - Proposed in 1989
  - SNMP information include “version number”, “Community Name”, and “PDU”

- **SNMPv2**
  - Proposed in 1993, Revised in 1995
  - An upgrade to SNMPv1
  - Add functional enhancements to SNMPv1 (ex: GetBulk-PDU)

- **SNMPv3**
  - Proposed in 1998
  - Define Authentication, Security, and Access Control for SNMP security mechanism
  - Used with the functionality provided by SNMPv2
SNMP protocol

Two ways to convey MIB info, commands:

- **request/response mode**
  - **request** → **response**
  - **agent** → **data**
  - **Managed device**

- **trap mode**
  - **trap msg**
  - **agent** → **data**
  - **Managed device**
Principle Operation

Transport SNMP
v1: unreliable
v2/v3: reliable alternatives

Manager

Agents

MIB

variable

table

GET/SET

TRAPS

poll

poll

poll
### SNMP protocol: message types

<table>
<thead>
<tr>
<th>Message type</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetRequest</td>
<td>Mgr-to-agent: “get me data” (instance,next in list, block)</td>
</tr>
<tr>
<td>GetNextRequest</td>
<td></td>
</tr>
<tr>
<td>GetBulkRequest</td>
<td></td>
</tr>
<tr>
<td>InformRequest</td>
<td>Mgr-to-Mgr: here’s MIB value</td>
</tr>
<tr>
<td>SetRequest</td>
<td>Mgr-to-agent: set MIB value</td>
</tr>
<tr>
<td>Response</td>
<td>Agent-to-mgr: value, response to Request</td>
</tr>
<tr>
<td>Trap</td>
<td>Agent-to-mgr: inform manager of exceptional event</td>
</tr>
</tbody>
</table>
SNMP protocol: message formats
## SNMP protocol: message types

<table>
<thead>
<tr>
<th>Message type</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetRequest</td>
<td>Mgr-to-agent: “get me data” (instance, next in list, block)</td>
</tr>
<tr>
<td>GetNextRequest</td>
<td></td>
</tr>
<tr>
<td>GetBulkRequest</td>
<td></td>
</tr>
<tr>
<td>InformRequest</td>
<td>Mgr-to-Mgr: here’s MIB value</td>
</tr>
<tr>
<td>SetRequest</td>
<td>Mgr-to-agent: set MIB value</td>
</tr>
<tr>
<td>Response</td>
<td>Agent-to-mgr: value, response to Request</td>
</tr>
<tr>
<td>Trap</td>
<td>Agent-to-mgr: inform manager of exceptional event</td>
</tr>
</tbody>
</table>
Principle Operation (cont.)

SNMP Manager Station

Manager Application

Get-Request
GetNext-Request
Set-Request
Get-Response
Trap

UDP
IP
DLC / MAC
PHY

SNMP Managed Agent

Managed Resource

Managed Objects

MIB

Get-Request
GetNext-Request
Set-Request
Get-Response
Trap

UDP
IP
DLC / MAC
PHY

Port 161

Port 162
SNMP Command Tools

• snmptest
• snmpget
• snmpgetnext
• snmpset
• snmptrap
• snmpwalk
• snmpnetstat
Simple Network Management Protocol (SNMP)

Network Management Software (Manager)

Management Information Base (MIB)

Simple Network Management Protocol (SNMP)

Command (Get, Set, etc.)

Response

Trap

Management Information Base (MIB)

RMON Probe
SNMP Command - Protocol Data Units (PDUs)

SNMPv1
- get
- getnext
- set
- getresponse
- trap

SNMPv2
- get
- getnext
- set
- getresponse
- trap
- getbulk
- notification
- inform
- report

SNMPv3
- get
- getnext
- set
- getresponse
- trap
- getbulk
- notification
- inform
- report
SNMP Community

- Manager & Agent: trust relationship upon community
- The basic security of SNMP
- Used to identify the group of nodes
- Define where an SNMP message is destined for
- Only when two applications in the same community group can communicate with each other
- Default community name
  - public
  - private
- Each community name is assigned an access mode
  - read-only (default for “public” community)
  - read-write (default for “private” community)
Extension to SNMP management architecture – RMON, Proxy Agents
RMON

- RMON, Remote MONitoring
- RMON agents or probe has MIBs
  - v1: RFC 1757
  - v2: RFC 2021
- Monitor the data flowing on the remote network using probe or RMON agents
- An extension of the network manager’s operation
- Reduces the amount of information required to be transmitted to NMS (data collection as offline mode)
RMON (cont.)

<table>
<thead>
<tr>
<th>Rmon Groups (1.3.6.1.2.1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistic Group (1)</td>
</tr>
<tr>
<td>Traffic Matrix Group (6)</td>
</tr>
<tr>
<td>History Group (2)</td>
</tr>
<tr>
<td>Filter Group (7)</td>
</tr>
<tr>
<td>Alarms Group (3)</td>
</tr>
<tr>
<td>Capture Group (8)</td>
</tr>
<tr>
<td>Hosts Group (4)</td>
</tr>
<tr>
<td>Events Group (9)</td>
</tr>
<tr>
<td>Host Top N Group (5)</td>
</tr>
</tbody>
</table>

Network Management Station

WAN circuit

RMON Agent/Probe

RMON-MIBs

Agent

MIB
Proxy Agents

- Proxy Agents extend the capabilities of SNMP by allowing it to
  - Manage a device that cannot support an SNMP agent
  - Manage a device that supports a non-SNMP management agent
  - Allow a non-SNMP management system to access an SNMP agent
  - Provide firewall-type security to other SNMP agents (UDP packet filtering)
  - Translate between different formats of SNMP messages (v1 and v2)
  - Consolidate multiple managed nodes into a single network address (also to provide a single trap destination)
SNMP security and administration

• encryption: DES-encrypt SNMP message

• authentication: compute, send MIC(m,k): compute hash (MIC) over message (m), secret shared key (k)

• protection against playback: use nonce

• view-based access control
  – SNMP entity maintains database of access rights, policies for various users
  – database itself accessible as managed object!
SMI, ASN.1, BER, OID, MIB, …, etc.

SMI & MIBS
**Language of SNMP**

- **SMI (Structure of Management Information)**
  - Specifies the format used for defining managed objects that are accessed via the SNMP protocol

- **ASN.1 (Abstract Syntax Notation One)**
  - Used to define the format of SNMP messages and managed objects (MIB modules) using an unambiguous data description format

- **BER (Basic Encoding Rules)**
  - Used to encode the SNMP messages into a format suitable for transmission across a network
A real-life presentation problem:
Presentation problem: potential solutions

1. Sender learns receiver’s format. Sender translates into receiver’s format. Sender sends.
   – real-world analogy?
   – pros and cons?

2. Sender sends. Receiver learns sender’s format. Receiver translate into receiver-local format
   – real-world-analogy
   – pros and cons?

   – real-world analogy?
   – pros and cons?
Solving the presentation problem

1. Translate local-host format to host-independent format
2. Transmit data in host-independent format
3. Translate host-independent format to remote-host format
ASN.1: Abstract Syntax Notation 1

• ISO standard X.680
  – used extensively in Internet
  – like eating vegetables, knowing this “good for you”!

• defined data types, object constructors
  – like SMI

• BER: Basic Encoding Rules
  – specify how ASN.1-defined data objects to be transmitted
  – each transmitted object has Type, Length, Value (TLV) encoding
**TLV Encoding**

**Idea:** transmitted data is self-identifying

- **T:** data type, one of ASN.1-defined types
- **L:** length of data in bytes
- **V:** value of data, encoded according to ASN.1 standard

<table>
<thead>
<tr>
<th>Tag Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Boolean</td>
</tr>
<tr>
<td>2</td>
<td>Integer</td>
</tr>
<tr>
<td>3</td>
<td>Bitstring</td>
</tr>
<tr>
<td>4</td>
<td>Octet string</td>
</tr>
<tr>
<td>5</td>
<td>Null</td>
</tr>
<tr>
<td>6</td>
<td>Object Identifier</td>
</tr>
<tr>
<td>9</td>
<td>Real</td>
</tr>
</tbody>
</table>
TLV encoding: example

Value, 259
Length, 2 bytes
Type=2, integer

Value, 5 octets (chars)
Length, 5 bytes
Type=4, octet string
ASN.1

• Addresses both syntax and semantics
• Two type of syntax
  – **Abstract syntax**: set of rules that specify data type and structure for information storage
  – **Transfer syntax**: set of rules for communicating information between systems
• Makes application layer protocols independent of lower layer protocols
• Can generate machine-readable code: Basic Encoding Rules (BER) is used in management modules
BER

- CCITT X.209 specifies the Basic Encoding Rules
- The relationship between ASN.1 and BER parallels that of source code and machine code
- All SNMP messages are converted / serialized from ASN.1 notation into smaller, binary data (BER)
SMI  Structure of Management Information

• Denotes:
  – How MIB variables in an MIB are related to one another
  – How variables are formatted
  – Information to obtain the standardization of the MIB

• Purpose: to make the definition of new MIBs easier
  – Help to guide MIB designer
  – Define the syntax
  – Allow tools to be built
SMI (cont.)

• SMI defines for a managed object
  – Syntax
  – Semantics
  – Other information (ex: status)

• Definition of the high-level structure of the internet branch
  \texttt{(iso(1).org(3).dod(6).internet(1))} of the MIB naming tree

• The subset of the ASN.1 language that is used in MIBs
SMI: data definition language

**Purpose:** syntax, semantics of management data well-defined, unambiguous

- base data types:
  - straightforward, boring
- **OBJECT-TYPE**
  - data type, status, semantics of managed object
- **MODULE-IDENTITY**
  - groups related objects into MIB module

**Basic Data Types**

- INTEGER
- Integer32
- Unsigned32
- OCTET STRING
- OBJECT IDENTIFIED
- IPaddress
- Counter32
- Counter64
- Guage32
- Time Ticks
- Opaque
SMI: Object, module examples

**OBJECT-TYPE:** ipInDelivers

```plaintext
ipInDelivers OBJECT TYPE
SYNTAX     Counter32
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
   “The total number of input
dataagrams successfully
delivered to IP user-
protocols (including ICMP)”
 ::= { ip 9}
```

**MODULE-IDENTITY:** ipMIB

```plaintext
ipMIB MODULE-IDENTITY
LAST-UPDATED “941101000Z”
ORGANIZATION “IETF SNPv2
Working Group”
CONTACT-INFO
   “ Keith McCloghrie
       ……”
DESCRIPTION
   “The MIB module for managing IP
and ICMP implementations, but
excluding their management of
IP routes.”
REVISION “019331000Z”
```

.......... 
```plaintext
 ::= {mib-2 48}
```
SMI Object Tree

Root-Node

ccitt(0)  iso(1)  joint(2)

org(3)
dod(6)

internet(1)
dictionary(1)  mgmt(2)  experimental(3)  private(4)

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 }
Summary

SNMP message command

BER

MIB

SMI

Managed Object

ASN.1
SMIv2

• SMIv2 improves SMIv1
• “snmpV2” branch additional
  – 1.3.6.1.6.3.1.1
  – iso.org.dod.internet.snmpV2.snmpModules.snmpMIB.snmpMIBObjects
• Other new object type definition
Object Definition Form

<name> OBJECT-TYPE
  SYNTAX <datatype>
  UnitsParts <Optional, See below>
  MAX-ACCESS < See below >
  STATUS < See below >
  DESCRIPTION
    “Textual description describing this particular managed object.”
  AUGMENTS { <name of table> }
  ::= { <Unique OID that defines this object> }

<table>
<thead>
<tr>
<th>Object Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnitsParts</td>
<td>time unit's description (ex. seconds, milliseconds, etc.)</td>
</tr>
<tr>
<td>MAX-ACCESS</td>
<td>read-only, read-write, read-create, not-accessible, and accessible-for-notify</td>
</tr>
<tr>
<td>STATUS</td>
<td>current(mandatory), obsolete, and deprecated</td>
</tr>
<tr>
<td>AUGMENTS</td>
<td>increase one or more columns for expand the table</td>
</tr>
</tbody>
</table>
### Object Type Definition

#### OBJECT-TYPE:

<table>
<thead>
<tr>
<th>SYNTAX</th>
<th>INTEGER</th>
<th>OCTET STRING</th>
<th>OBJECT IDENTIFIER</th>
<th>BITS</th>
<th>IpAddress</th>
<th>Integer32</th>
<th>Counter32</th>
<th>Counter64</th>
<th>Gauge32</th>
<th>TimeTicks</th>
<th>Opaque</th>
<th><strong>New Type</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-ACCESS</td>
<td>read-only</td>
<td>read-write</td>
<td>read-create</td>
<td>accessible-for-notify</td>
<td>not-accessible</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATUS</td>
<td>current</td>
<td>deprecated</td>
<td>obsolete</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>&quot;&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SNMP MIB

MIB module specified via SMI

MODULE-IDENTITY
(100 standardized MIBs, more vendor-specific)

MODULE

OBJECT TYPE:

OBJECT TYPE:

OBJECT TYPE:

objects specified via SMI

OBJECT-TYPE construct
# MIB example: UDP module

<table>
<thead>
<tr>
<th>Object ID</th>
<th>Name</th>
<th>Type</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.6.1.2.1.7.1</td>
<td>UDPInDatagrams</td>
<td>Counter32</td>
<td>total # datagrams delivered at this node</td>
</tr>
<tr>
<td>1.3.6.1.2.1.7.2</td>
<td>UDPNoPorts</td>
<td>Counter32</td>
<td># underdeliverable datagrams no app at port1</td>
</tr>
<tr>
<td>1.3.6.1.2.1.7.3</td>
<td>UDInErrors</td>
<td>Counter32</td>
<td># undeliverable datagrams all other reasons</td>
</tr>
<tr>
<td>1.3.6.1.2.1.7.4</td>
<td>UDPOutDatagrams</td>
<td>Counter32</td>
<td># datagrams sent</td>
</tr>
<tr>
<td>1.3.6.1.2.1.7.5</td>
<td>udpTable</td>
<td>SEQUENCE</td>
<td>one entry for each port in use by app, gives port # and IP address</td>
</tr>
</tbody>
</table>
SNMP Naming

**question:** how to name every possible standard object (protocol, data, more..) in every possible network standard??

**answer:** *ISO Object Identifier tree:*

– hierarchical naming of all objects
– each branchpoint has name, number

```
1.3.6.1.2.1.7.1
ISO
US DoD
Internet
udpInDatagrams
UDP
MIB2
management
```
Check out www.alvestrand.no/harald/objectid/top.html
MIB-II

• MIB-II is the most important management group of SNMP
• Each device (which supports SNMP) supports MIB-II as well
• RFC1213-MIB defined the branch of mib-2’s OIDs
• Defined base on SMIv1
• mib-2 defined as
  – iso.org.dod.internet.mgmt.1
  – 1.3.6.1.2.1
MIB-II (cont.)

- mib-2(1) (1.3.6.1.2.1)
  - system(1) (1.3.6.1.2.1.1) 7 sysDescr, sysName, sysUptime, …
  - interfaces(2) (1.3.6.1.2.1.2) 23 ifInError, ifOutError, …
  - at(3) (1.3.6.1.2.1.3) 3 (Address Translation), atPhysAddress, …
  - ip(4) (1.3.6.1.2.1.4) 42 ipRoutingTable, ipNetToMediaTable, …
  - icmp(5) (1.3.6.1.2.1.5) 26 icmpInEchos, icmpOutEchos, …
  - tcp(6) (1.3.6.1.2.1.6) 19 tcpConnTable, …
  - udp(7) (1.3.6.1.2.1.7) 6 udpTable, …
  - egp(8) (1.3.6.1.2.1.8) 20 egpNeighAs, …
  - transmission(10) (1.3.6.1.2.1.10) 0 non
  - snmp(11) (1.3.6.1.2.1.11) 30 snmpInTraps, …
SNMP Object Model

• SNMP Object Model
  – The MIB database schema
  – Defines objects (parameters) about which information is stored for each managed device

• SNMP System Objects
  – System name
  – System description
  – System contact person
  – System uptime (since last reboot)
SNMP Object Model

• SNMP IP Objects
  – Forwarding (for routers). Yes if forwarding (routing), No if not
  – Subnet mask
  – Default time to live
  – Traffic statistics
  – Number of discards because of resource limitations
SNMP Object Model

• SNMP IP Objects (Continued)
  – Number of discards because could not find route
  – Number of rows in routing table
  – Rows discarded because of lack of space
  – Individual row data in the routing table
SNMP Object Model

• SNMP TCP Objects
  – Maximum / minimum retransmission time
  – Maximum number of TCP connections allowed
  – Opens / failed connections / resets
  – Segments sent
  – Segments retransmitted
  – Errors in incoming segments
  – No open port errors
  – Data on individual connections (sockets, states)
SNMP Object Model

• SNMP UDP Objects
  – Error: no application on requested port
  – Traffic statistics

• SNMP ICMP Objects
  – Number of errors of various types
SNMP Object Model

- SNMP Interface Objects (One per Port)
  - Type (e.g., 69 is 100Base-FX; 71 is 802.11)
  - Status: up / down / testing
  - Speed
  - MTU (maximum transmission unit—the maximum packet size)
  - Traffic statistics: octets, unicast / broadcast / multicast packets
  - Errors: discards, unknown protocols, etc.
SNMP Agents, NMS, Trend Analysis Tool, ..., etc

RELATE TOOLS
Relate Tools

• SNMP Agents
  – AdventNet SNMP Agent
  – Microsoft
  – …

• NMS
  – HP OpenView
  – Net-SNMP
  – Nagios
  – OpenNMS
  – NINO
  – …

• Trend Analysis Tool
  – MRTG (Multi Router Traffic Grapher)
  – RRDtool
  – …

• Others
  – ping, ipconfig / ifconfig, arp, netstat, tracert / traceroute, …, etc.
Network Management Tools

- SNMP command tools
- MIB Walk
- MIB Browser
- snmipsniff
SNMP Browser

- Command: snmpwalk host community [variable name]
- Uses Get Next Command
- Presents MIB Tree
Conclusion, Future Works, and References

CONCLUSION
Conclusion

- SNMP become the most popular Network Management System for INTERNET world
- Current SNMP used
  - SNMPv1
  - SNMPv2c
  - SNMPv3
- Current SMI used
  - SMIv2
- Current MIB used
  - MIB-II
Conclusion (cont.)

• network management
  – extremely important: 80% of network “cost”
  – ASN.1 for data description
  – SNMP protocol as a tool for conveying information
• Network management: more art than science
  – what to measure/monitor
  – how to respond to failures?
  – alarm correlation/filtering?