Network Management

- introduction
- Internet SNMP: Simple Network Management Protocol
- required reading: section 7.3 in text, KR Ch 8

Network Management: Introduction

- network consists of many heterogeneous, multi-vendor resources: routers, bridges, hosts, terminal servers, modems, links, interfaces
- goal of network management:
  - identification and correction of hardware/software failure of malfunction
  - performance monitoring and tuning
Network Management issues

Approaches to network management must:
- scale well: large number of entities to manage
- not interfere with normal operating (low overhead)
- operate under stress: most important when network under stress

Issues:
- which resources will be managed
- how to name/describe managed resources

standards:
- Internet: SNMP: Simple Network Management Protocol
- OSI CMIP: Common Management Information Protocol
**Managing and Managed Entities**

**Managing entity:**
- has "big picture" view of network
- set of application-level programs
  controlling/managing network
  - with human intervention
  - with rule-based AI (expert) system assistance
- communicates with managed entities to:
  - query (poll) status (e.g., link states, routing tables, number of packets dropped)
  - have managed entities make changes: e.g., change turn a link off

**Managed Entity:**
- application-level process located at each resource site to communicate with network manager and do its bidding
  - responds to queries from manager
  - notify manager of significant events (e.g., link down)
SNMP

- managing entity resides in network management station (NMS)
- managed entity called SNMP agent
- MIB: Management Information Base
  - logical store of information for network management
  - locally maintained by SNMP agent
  - queried and modified by NMS
  - 175 "objects" organized into 10 groups: system, interfaces, address translation, IP, ICMP, TCP, UDP, EGP, transmission, SNMP

UDP-related MIB variables

<table>
<thead>
<tr>
<th>name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>udpInDatagrams</td>
<td># UDP dg's delivered to processes</td>
</tr>
<tr>
<td>udpNoPorts</td>
<td># UDP dg's with no receiving applic.</td>
</tr>
<tr>
<td>udpInErrors</td>
<td># other UDP errors (e.g., checksums)</td>
</tr>
<tr>
<td>udpOutDatagrams</td>
<td># UDP dg's sent</td>
</tr>
<tr>
<td>udpTable</td>
<td>table of IP (interface) addresses and ports for which system will receive UDP dg's. e.g., port 520 for RIP routing msgs</td>
</tr>
<tr>
<td>ifIndex</td>
<td>index of interface</td>
</tr>
<tr>
<td>ifDescr</td>
<td>textual description of interface</td>
</tr>
<tr>
<td>ifType</td>
<td>interface type (e.g., 7 for IEEE 802.3)</td>
</tr>
</tbody>
</table>
## MIB Variables (cont)

<table>
<thead>
<tr>
<th>name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ifMTU</td>
<td>maximum packet size</td>
</tr>
<tr>
<td>ifSpeed</td>
<td>speed in bits/sec</td>
</tr>
<tr>
<td>ifPhysAddress</td>
<td>physical address (e.g., 802.* address)</td>
</tr>
<tr>
<td>ifOperStatus</td>
<td>1 if up, 2 if down, 3 if testing</td>
</tr>
<tr>
<td>ifInErrors</td>
<td># incoming pkts discarded due to errors</td>
</tr>
<tr>
<td>ifInDiscards</td>
<td># incoming pkts discarded due to buffer overflow</td>
</tr>
<tr>
<td>ifInUcastPkts</td>
<td># incoming unicast pkts received</td>
</tr>
<tr>
<td>ifOutQLen</td>
<td># pkts in outbound queue</td>
</tr>
</tbody>
</table>

## Referencing MIB variables

### Recall (?) ASN.1 OBJECT IDENTIFIER type:

- provides structured, ISO-standard method for naming objects
- nameable objects include protocols and MIB variables
e.g., 1.3.6.1.2.1.7.1 specifies number of udp packets delivered to user processes:

![SNMP Diagram]

**SNMP Protocol**

- communication between managing entity and managed entity via UDP ports 161, 162
  - aside: why not TCP!
- SNMP protocol has 5 message types:
  - **get-request**: fetch value of one or more MIB variables
  - **get-next-request**: for looping through variables and tables
  - **set-request**: tell agent to set value of MIB variable to specified value
  - **get-response**: used by agent to return value to manager
  - **trap**: used by agent to notify manager of "event"
Trap packets:

<table>
<thead>
<tr>
<th>trap name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cold start</td>
<td>SNMP agent initializing self</td>
</tr>
<tr>
<td>warm start</td>
<td>SNMP agent reinitializing</td>
</tr>
<tr>
<td>link up</td>
<td>interface changed from down to up state</td>
</tr>
<tr>
<td>link down</td>
<td>interface changed from up to down state</td>
</tr>
<tr>
<td>authentication failure</td>
<td>SNMP pkt received from unknown manager</td>
</tr>
</tbody>
</table>

Network Management: Summary

- emerging standard for naming, access, modification for network resources and data
- network management tools: focus on display and management of large amounts of data
- how to manage network
  - manage/avoid failures
  - manage performance

Still much more art than science!
- wide open, important research field
- FYI reading:
  - Nov. 1993 issue of IEEE Network magazine