COMPSCI 314 S1 C

BOOTP, DHCP,

MIBs and SNMP

BOOTP: RFC 951

- ARP finds a host's IP address, but ARP is a link-layer (layer 2) protocol, it can't be routed
- BOOTP (the Bootstrap Protocol) is an application protocol using UDP transport:
  - Requesting host sends a BOOTP request packet containing the client's MAC address to port 67 at the IP broadcast address
  - BOOTP server looks up the client MAC address in a database, then sends response back to client port 68
  - Server may add client to its ARP cache, otherwise it broadcasts the reply

BOOTP relay

- Administrator can maintain database of MAC addresses and their IP addresses
- Simple networks may use a separate BOOTP server for each subnet (e.g. City and Tamaki)
- Larger networks can run a single server, and configure routers to relay BOOTP datagrams
  - Router receives a BOOTP request, forwards it to a specific BOOTP server
  - Router receives BOOTP response, sends it to requesting client

DHCP: RFC 2131

- An extension of BOOTP to carry other data besides client IP address, e.g. netmask, default gateway, nameserver
- Includes ability to allocate addresses *dynamically*
  - Administrator maintains list of permanently-allocated IP addresses for fixed hosts, and range of addreses for other hosts
  - Client is leased (allocated) an address from the range for a specified lease time. Leases may be extended or revoked
- DHCP (and BOOTP) check for clients using same IP address

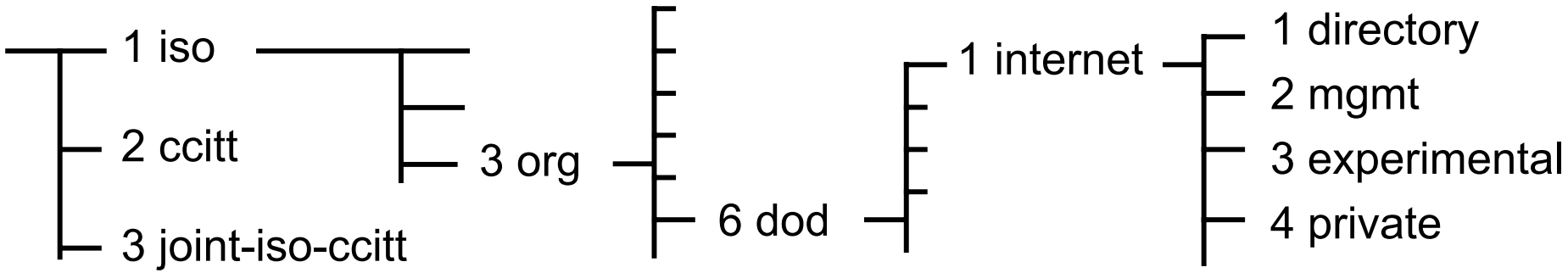
Network Management

- 'Management' (IETF-style) means
  - Configuring devices
  - Monitoring device operation
  - Changing device configurations
  - Keeping track of configurations as they change
- Simple (late 1980s) view
  - Each device keeps configuration in a 'Management Information Database' (MIB)
  - SNMP allows a manager to interact with a device via its MIB
- Current (2000s) view
  - Need to manage large sets of devices as a coherent whole

Management Information Base

- Management of a network involves reading and setting many network-related values, as we evaluate the network performance and adjust its operation
- The management values are held in an extensive database using a structure agreed between Internet and OSI network management, known as the *Management Information Base* or *MIB*
- We can monitor a device's behaviour by watching values in the MIB, e.g. bytes in/out of an interface
- The MIB is a hierarchical structure, with a name identified by the sequence of node indices as the tree is traversed from the root to the node

MIB

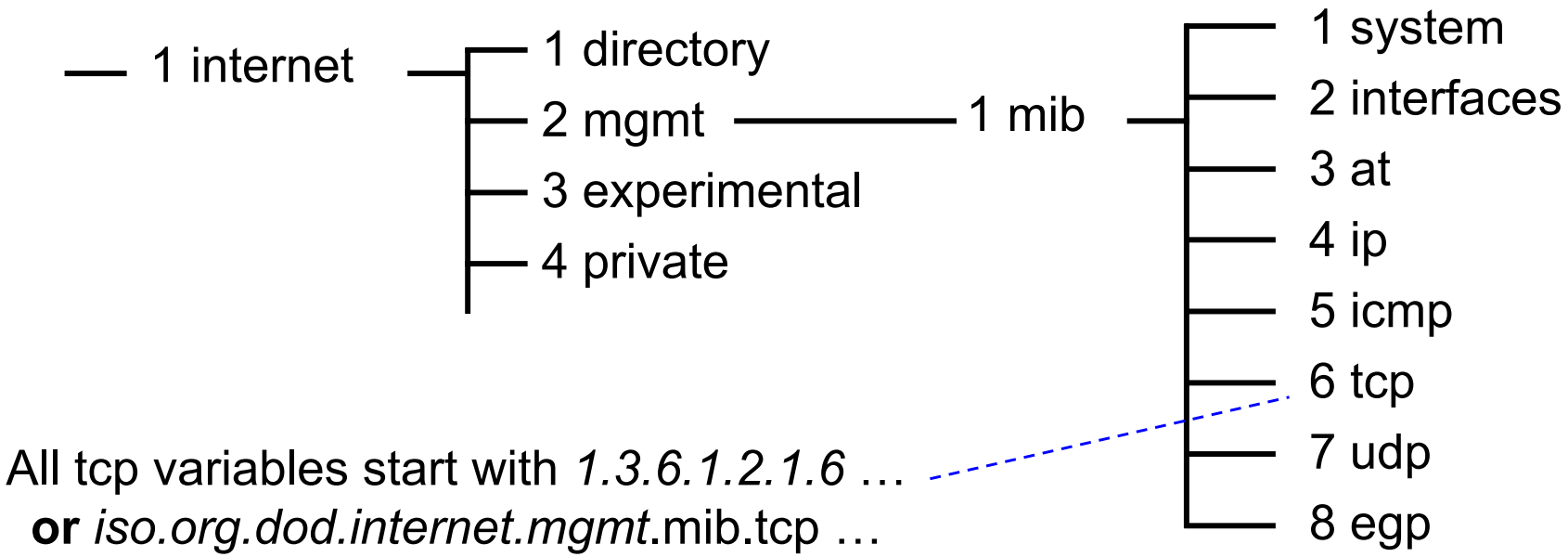


All internet Management variables start with
1.3.6.1.2. … **or** *iso.org.dod.internet.mgmt*

First levels of Management Information Base tree

- There are three basic trees, from an unnamed root node.
- The one of interest is ‘internet,’ which is managed by ‘dod’ (the Department of Defense), which is an ‘org’ (Organisation) known to the International Standards Organisation ‘iso’

MIB (2)



Namespace for variables within Internet MIB-I (RFC 1156)

- MIB-I is 'for Network Management of TCP/IP-based internets'
- It's *interface* variables are very useful, e.g.
  `ifInOctets` and **`ifOutOctets`**

Simple Network Management Protocol (SNMP)

The SNMP protocol provides the user interface to the MIB and is in some respects an extension of the MIB.

Based on a request/response, or fetch/store paradigm, in which the manager may request values for variables or supply values; actions may occur as a side-effect of writing values.

Command	Meaning
get-request	Fetch a value from a specific variable
get-next-request	Fetch the value following that fetched by earlier get-request
set-request	Store a value in a specific variable
get-response	Reply to a get operation
trap	Message triggered by an event

SNMP (2)

- SNMP operations are atomic, meaning that all of the actions of a message must occur, or none will occur; any error causes the whole request to be abandoned
- The problem of scanning through tables is solved by the *get-next-request* which has an identifier for a known item in a table, but triggers a *get- response* corresponding to the next table entry. A *get-next-request* to the table itself returns the first entry
- SNMP is usually implemented as a datagram protocol, using UDP. SNMP response packets serve as Acks to set- or get- requests
- Each command is given a 32-bit sequence number which is returned in replies
- Replies can be associated with requests, or repeated commands ignored

Details from here on are not examinable, but principles should be known.

Fortunately, SNMP is supported by a suite of public-domain routines which do most of the message formatting and interpretation for the user, so that even people working at the detailed level seldom need to know all the gory details.

SNMP(3)

- The GetRequest-PDU for example contains a 32-bit *requestID* which is essentially a sequence number of the request to match requests and responses, error indicators and a list of objects for which values are wanted.
- The messages tend to be complex. The next slide contains the lines —

```
 A0     1C      02      04      05     AE      56      02
getreq len=28 INTEGER len=4 |------- request id --------|
```

A0	identifies a Get-Request command
1C	length of the command
02	an integer follows (known to be the request ID)
04	the length of the integer
05 AE 56 02	the 4 bytes of the integer value

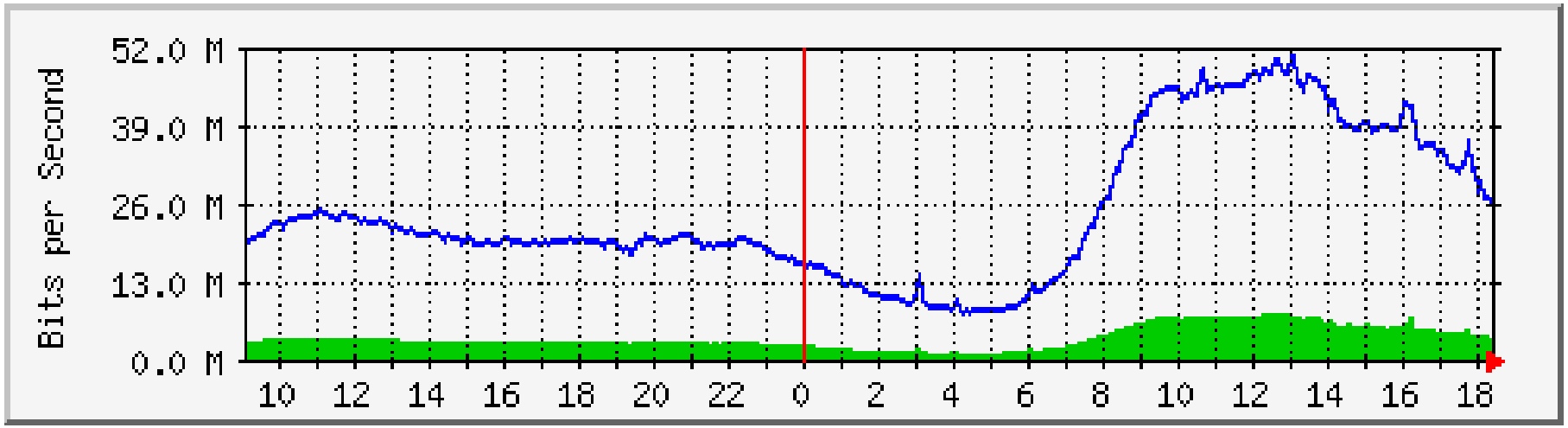
SNMP(4)

```
  30        29       02          01     00
SEQUENCE len=41  INTEGER     len=1  vers=0

  04        06      70  75   62  6C   69   63
string    len=6     p   u    b   l    i    c

  A0        1C        02       04        05  AE  56  02
getreq    len=28  INTEGER len=4   |-- request id --|

  02        01        00    02        01        00
INTEGER   len=1     00  INTEGER   len=1 err.index

  30        0E        30          0C      06      08
SEQUENCE len=14  SEQUENCE len=12 objectid len=8

  2B        06    01    02    01    01    01   00
  1.3        6     1     2     1     1     1    0

  05        00
 null     len=0
```

Example of SNMP message – get-request for sysDescr (1.3.6.1.2.1.1.1)

Monitoring a link with MRTG

MRTG reads SNMP variables and plots them on web pages, for example:



Blue trace for **`ifInOctets`**, green for **`ifOutOctets`**

[plot from http://sysadmin.oreilly.com/news/slb_0301.html]

NeTraMet: a Network Traffic Meter

A system of:

- *Meters*, to collect measurements on the attached networks
- *Meter readers*, to collect data from meters (meters & readers may be many to many)
- *Managers*, to coordinate and process data from readers
- Communication (configuration and reading) is by SNMP