Internetworking

- There exist many LAN and WAN (Wide Area Networking) techniques
- Providing universal service requires interconnecting networks with different technologies
  - The telephone is an example of universal service!
- Internetworking means connecting multiple networks of dissimilar technologies
  - Requires special hardware between networks and special software in attached computers

Repeaters

- A repeater is a hardware device used to extend a LAN.
- It connects two cable segments, amplifying and sending all electrical signals from one cable to the other.
  - Including collisions!
- Pairs of computers on the extended LAN can communicate with each other
  - They do not know whether a repeater separates them
Repeaters (cont’d)

Figure 11.2 A repeater R connecting two Ethernet segments. The repeater connects directly to the cable.

Bridges

- A bridge is a hardware device used to extend a LAN.
- A bridge forwards complete, correct frames (packets) from one segment to another
  - Does not forward interference or collisions!
- Pairs of computers on the extended LAN can communicate with each other
  - They do not know whether a bridge separates them
**Routers**

- A router is a hardware component that interconnects networks
- A router has interfaces on multiple networks

**Routers (cont’d)**

- Networks can use different technologies
- Router forwards packets between networks
  - Uses packet address to decide which network to send it to
- Transforms packets as necessary to meet standards for each network
- An internetwork is composed of arbitrarily many networks interconnected by routers
Routers (cont’d)

- Note that routers can have more than two interfaces!

Virtual Networks

- Internetworking software builds a single, seamless **virtual network** out of multiple physical networks
  - Universal addressing scheme
  - Universal service
- All details of physical networks are hidden from users and application programs
Virtual Networks (cont’d)

TCP/IP

- **TCP/IP** is the most widely used internetworking protocol suite
  - Initially funded through ARPA
  - Picked up by NSF
  - Used in the Internet
- Other internetworking protocols exist but are less used
  - Example: AppleTalk
Host and Routers

- A **host computer** (or just **host**) is any system attached to an internetwork that runs applications
  - May be a supercomputer or a toaster!
- TCP/IP allows any pair of hosts on an internetwork to communicate directly
- Differently from routers, hosts typically have only one interface and don’t forward packets

Universal Addressing

- One key aspect of virtual networks is **single, uniform address format**
- Sending hosts put destination internetworking address in the packet
- Destination addresses can be interpreted by any intermediate router
- Router examine address and forward packet on to the destination
**IP Addresses**

- Each computer on the Internet (host machine) has a unique IP address
  - Sometimes the IP address is defined “dynamically” at the time a machine connects
- The IP address is different from the “physical” or “MAC” address
  - The “physical address” is the address of a computer (actually, of a NIC) in the LAN
    - It is only know within the LAN
  - The IP address is a universal address
  - When a packet arrives in a LAN, there needs to be a conversion from IP to MAC address (“address resolution”)

**IP Addresses (cont’d)**

- An IP address is represented by a binary number with 32 bits
  - Meaning that there are around 4 billion addresses
  - Often IP addresses are represented in “dotted decimal”, such as **128.114.144.4**
    - Each group of numbers can go from 0 to 255
IP Addresses (cont’d)

• We’ll now look at the organization of numerical IP addresses
• Each IP address is divided into a prefix and a suffix
  – Prefix identifies network to which computers are attached
  – Suffix identifies computers within that network
• Remember that two computers in a network can communicate without a router in between

Networks and Host Numbers

• Every network in a TCP/IP internet is assigned a unique network number
• Each host on a specific network is assigned a host address that is unique within that network
• Host’s IP address is the combination of the network number (prefix) and host address (suffix)
• Assignment of network numbers must be coordinated globally; assignment of host addresses can be managed locally
**IP Address Format**

- IP address are 32 bits long
- There are different classes of addresses, corresponding to different subdivisions of the 32 bits into prefix and suffix
  - Some address classes have large prefix, small suffix
    - Many such networks, few hosts per network
  - Other address classes have small prefix, large suffix
    - Few such networks, many hosts per network

---

**IP Address Format (cont’d)**

- How can we recognize to which class an IP address belongs to?
  - Look at the first 4 bits!

<table>
<thead>
<tr>
<th>Class</th>
<th>Prefix</th>
<th>Suffix</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>110</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>1111</td>
<td>reserved for future use</td>
</tr>
</tbody>
</table>
**IP Address Format (cont’d)**

- Class A, B and C are **primary classes**
  - Used for ordinary addressing
- Class D is used for **multicast**, which is a limited form of **broadcast**
  - Internet hosts join a **multicast group**
  - Packets are delivered to all members of the group
  - Routers manage delivery of single packets from source to all members of multicast group
  - Example: **mbone**
- Class E is reserved

**IP Addresses (cont’d)**

- Another way to determine the address class is by looking at the first group of numbers in the dotted decimal notation

<table>
<thead>
<tr>
<th>Class</th>
<th>Range of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0 through 127</td>
</tr>
<tr>
<td>B</td>
<td>128 through 191</td>
</tr>
<tr>
<td>C</td>
<td>192 through 223</td>
</tr>
<tr>
<td>D</td>
<td>224 through 239</td>
</tr>
<tr>
<td>E</td>
<td>240 through 255</td>
</tr>
</tbody>
</table>
Networks and Hosts in Each Class

<table>
<thead>
<tr>
<th>Class</th>
<th>Bits In Prefix</th>
<th>Maximum Number of Networks</th>
<th>Bits In Suffix</th>
<th>Maximum Number Of Hosts Per Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>7</td>
<td>128</td>
<td>24</td>
<td>16777216</td>
</tr>
<tr>
<td>B</td>
<td>14</td>
<td>16384</td>
<td>15</td>
<td>65536</td>
</tr>
<tr>
<td>C</td>
<td>21</td>
<td>2097152</td>
<td>8</td>
<td>256</td>
</tr>
</tbody>
</table>

Understanding IP Addresses

- Remember: the first 3 digits determine the class of the address
- Depending on the class of an address, we can find out its prefix and its suffix
  - If Class A: ppp.sss.sss.sss (with 0 ≤ ppp ≤ 127)
  - If Class B: ppp.ppp.sss.sss (with 128 ≤ ppp ≤ 191)
  - If Class C: ppp.ppp.ppp.sss (with 192 ≤ ppp ≤ 223)
- Examples:
  - 10.0.0.37 (class A)
  - 128.10.0.1 (class B)
  - 192.5.48.3 (class C)
Example: A Private Internet

DNS

- Numbers are not easy to remember
- The Domain Name System (DNS) maps number to names (DNS addresses)
  - Sometimes alternate names (alias) are used
- Each DNS address is formed by a host name followed by a domain name
  - Host_name.domain_name
DNS (cont’d)

- The domain name is formed by the institutional site name and the Top-Level Domain name (TLD)
  - So the entire address is in the form
    Host_name. Ist_site_name. TLD_name
- Examples:
  - sundance.ucsc.edu
  - soe.ucsc.edu (alias for sundance.ucsc.edu)
  - italia.cse.ucsc.edu
  - helios.jpl.nasa.gov

TLD

- TLD names identify organization types or geographical locations
- Examples:

<table>
<thead>
<tr>
<th>TLD</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.com</td>
<td>Commercial org.</td>
</tr>
<tr>
<td>.edu</td>
<td>Educational site in US</td>
</tr>
<tr>
<td>.gov</td>
<td>Government site in US</td>
</tr>
<tr>
<td>.mil</td>
<td>Military site in US</td>
</tr>
<tr>
<td>.net</td>
<td>Network site</td>
</tr>
<tr>
<td>.org</td>
<td>Nonprofit organization</td>
</tr>
<tr>
<td>.au</td>
<td>Australia</td>
</tr>
<tr>
<td>.ca</td>
<td>Canada</td>
</tr>
<tr>
<td>.fr</td>
<td>France</td>
</tr>
<tr>
<td>.de</td>
<td>Germany</td>
</tr>
<tr>
<td>.uk</td>
<td>Great Britain</td>
</tr>
<tr>
<td>.it</td>
<td>Italy</td>
</tr>
<tr>
<td>.es</td>
<td>Spain</td>
</tr>
<tr>
<td>.ac.uk</td>
<td></td>
</tr>
<tr>
<td>.edu.au</td>
<td></td>
</tr>
</tbody>
</table>

Countries define their own internal hierarchy (e.g., .ac.uk, .edu.au)
DNS (cont’d)

- Organizations can create any internal DNS hierarchy
- Authority for creating new subdomains within a domain name is delegated to each domain
  - Administration of ucsc.edu has authority to create cse.ucsc.edu and needs not contact any central naming authority

URL

- **URL (Uniform Resource Locator)**: it is the “address” of a web page
- A complete URL consists of 3 parts
  - Protocol identifier
    - e.g., http (hypertext transfer protocol)
  - Host name
  - Path name (location of file in the server)
- Examples:
  - http://www.soe.ucsc.edu/classes/cmpe080n/Fall03/index.html
  - http://128.114.48.30/classes/cmpe080n/fall03/index.html
  - http://robotics.jpl.nasa.gov/groups/mvts/homepage.html
URL (cont’d)

- Often times one does not need to type the path name (the server will expand it)
  - E.g.: \texttt{http://www.cse.ucsc.edu} is expanded into \texttt{http://www.cse.ucsc.edu/index.html}

- If the protocol identifier is missing, the resolver will add the \texttt{http} identifier
  - E.g., \texttt{www.cse.ucsc.edu} is expanded into \texttt{http://www.cse.ucsc.edu/}
  - Do we always need to start the host name with \texttt{www}? No!
  - But often times the server will give an alias name beginning with \texttt{www}
    - E.g. \texttt{www.soe.ucsc.edu} is an alias for \texttt{web01.cse.ucsc.edu}

Other Protocols

- \textbf{File Transfer Protocol (FTP)}
  - To exchange files with other computers
    - E.g. \texttt{ftp://ftp.wu-ftpd.org/}

- \textbf{Local Files}
  - To access file in the local computer
    - E.g. \texttt{file://pathname}

- \textbf{Telnet}
  - To remotely operate a UNIX machine
    - E.g. \texttt{telnet://hostname}
Name Resolution

• “Resolving a name” means mapping from the DNS name to the IP address
• A **client** computer calls a **DNS server** for name resolution
  – DNS request contains name to be resolved
  – DNS reply contains IP address for name in request

Example of DNS Hierarchy

For example, consider the URLs:
http://peanut.candy.foobar.com
http://almond.candy.foobar.com
http://walnut.candy.foobar.com
http://soap.foobar.com
DNS and Client-Server Computing

- DNS names are managed by a hierarchy of DNS servers
  - Hierarchy is related to DNS domain hierarchy
  - Each server manages a set of DNS addresses
  - A DNS server is responsible for converting
- Root server at top of tree knows about next level servers
- Next level servers, in turn, know about lower level servers

Example of DNS Hierarchy
Example of DSN Hierarchy (cont’d)

Using DNS Servers

- Each DNS server is the **authoritative server** for the names it manages
  - If request contains name managed by receiving server, that server replies directly
  - Otherwise, request is forwarded to the appropriate authoritative server
- DNS request is originally sent to **root server**, which points at next server to use
  - Eventually, the authoritative server for the DNS name in the request is located and IP address is returned
Choosing DNS Server Architecture

- Small organizations can use a single server
  - Easy to administer
  - Inexpensive
- Large organizations often use multiple servers
  - Reliability through redundancy
  - Improved response time through load sharing

IP addresses: how to get one?

- The network IP numbers are assigned by the Network Information Center
- How does host get its IP address in the network?
  Two possibilities:
  - 1: Hard-coded by system administrator in a file inside the host
    - Dynamically get address: “plug-and-play”
**DHCP**

- DHCP allows a computer to join a new network and automatically obtain an IP address. The network administrator establishes a pool of addresses for DHCP to assign.
- When a computer boots, it broadcasts a **DHCP request** to which a server sends a **DHCP reply**.
- DHCP allows non-mobile computers that run server software to be assigned a **permanent address** (won’t change when the computer reboots).
  - The permanent address actually needs to be renegotiated after a certain period of time.