Internet, HTTP and DNS
Examples
The Internet and an internet

internet :: involves the *interconnection* of multiple networks into a single large networks.

the “Internet” :: refers to the successor to ARPANET.

**IP (the Internet Protocol)** :: provides *connectionless* transfer of packets across an internet.
An internet

G = gateway

Copyright ©2000 The McGraw Hill Companies

Leon-Garcia & Widjaja: Communication Networks

Figure 1.18

Networks: HTTP and DNS 3
IP

• Currently provides *best-effort service*
  – packets may be lost (i.e., IP is unreliable).

• General design philosophy
  – Keep internal operations simple by relegating complex functions to the edge of the subnet.
  – IP can operate over any network
  – allow IP to scale!!!

  – *The end-to-end mechanisms are responsible for recovery of packet losses and congestion control.*
**IP**

- Uses *hierarchical address space* with location information embedded in the structure.

- **IP address is usually expressed in** *dotted-decimal notation* (*e.g.*, 128.100.11.56).
Internet

- Provides a *name space* to refer to machines connected to the Internet (e.g. chablis.cs.wpi.edu).
- The name space is hierarchical, but is only administrative and not used in network routing operations.
- DNS (Domain Name Service) provides automatic translation of names to addresses.
Applications and Layered Architectures

- In the 1970’s vendor companies (IBM and DEC) developed *proprietary networks* with the common feature of grouping communication functions into related and manageable sets called *layers*.

*network architecture* :: a set of *protocols* that specify how every layer is to function.

[LG&W pp.43-49]
Layering Advantages

• Simplified the design process.

• Led to flexibility in modifying and developing the network.

• Accommodates incremental changes more readily.
Layering Examples

Client/server relationship ::

– Server process waits for incoming requests by listening to a port.
– Client process makes requests as required.
– Server process provides responses to these requests.
– The server process usually runs in the background as a daemon (e.g. httpd is the server daemon for HTTP).
HTTP Example

- HTTP (HyperText Transfer Protocol) specifies rules by which the client and the server interact so as to retrieve a document.
- The protocol assumes the client and the server can exchange messages directly.
- The client software needs to set up a two-way connection prior to the HTTP request.
HTTP client/server interaction

HTTP client

Request

Response

HTTP server

Figure 2.1

Leon-Garcia & Widjaja: Communication Networks

Networks: HTTP and DNS
Networks: HTTP and DNS

HTTP
SMTP
DNS
RTP

TCP
UDP

IP

Network Interface 1
Network Interface 2
Network Interface 3

Networks: HTTP and DNS
The user clicks on a link to indicate which document is to be retrieved.

The browser must determine the address that contains the document. It does this by sending a query to its local name server.

Once the address is known the browser establishes a connection to the specified machine, usually a TCP connection. In order for the connection to be successful, the specified machine must be ready to accept TCP connections.

The browser runs a client version of HTTP, which issues a request specifying both the name of the document and the possible document formats it can handle.

The machine that contains the requested document runs a server version of HTTP. It reacts to the HTTP request by sending an HTTP response which contains the desired document in the appropriate format.

The TCP connection is then closed and the user may view the document.

Retrieving a Web Page
HTTP client

HTTP server

Ephemeral

Port #

TCP

GET 80, #

#, 80 STATUS

Port 80

Networks: HTTP and DNS