cis3.2 — electronic commerce — 6 feb 2006 — lecture # 4

today

topics:

- the internet
- internet addressess
- domain names
- clients and servers
- ports and sockets
- network concepts

reading:

- today: Ince chapter 2 (sections 3.3 through 3.7; note that we are skipping section 4)
- for next week: Ince chapter 3

the internet

- history of the internet
  - ARPAnet (circa 1971): used "NCP"
  - TCP (1974): hardware independent, open
  - internet was standardized in September 1981
- structure
  - internet layers:

telnet	SNMP	*OSI model*
FTP	TFTP	*upper*
user applications	user applications	*layers*
TCP	UDP	*lower*
IP	ICMP	*layers*

  - telnet: lets users log on remotely
  - FTP: file transfer protocol
  - SNMP: simple network management protocol (monitors network for problems)
  - TFTP: trivial file transfer protocol (fast file xfer, lacks security)
  - other example (user) applications:
    - ∗ SMTP: simple mail transfer protocol (transfers mail msgs from one computer to another)
    - ∗ kerberos: security protocol (allows transfer of highly confidential data)
    - ∗ DNS: domain name system (enables symbolic instead of numeric host naming)
    - ∗ NFS: network file system (allows sharing of files between computers)
  - TCP: can re-transmit if errors
  - UDP: no error checking, fast messaging
  - IP: i.e., moving data via TCP or UDP
  - ICMP = internet control message protocol (checks status of computers with other network devices)

internet addresses

- *IP address* = Internet Protocol address
- every computer on the internet has a unique address
- *dotted quad notation* = four numbers separated by dots (.); e.g., 146.245.250.131 (which is the address of the CIS dept web server...)
- address can be stored in 32 bits; there are four formats, depending on the size of the network (i.e., the size of each of the numbers in the dotted quad notation is defined by the format...)
- "subnetting" is a standard (defined in 1985) to divide a large network into a number of smaller networks (this is what a *router* does)
- example:

	network prefix	subnet number	host number
IP address 130.5.5.25 →	130.5.	.5	.25

domain names

- provides a more convenient way to address a computer on the internet than the numeric IP address
- structured hierarchically (see tree structure figure 2.7)
- example: `www.sci.brooklyn.cuny.edu`
- common *domain names*: `com`, `edu`, `gov`, `uk` and other country-based domains
- *name server*: maintains correspondance between numeric IP address and domain names;
- DNS = internet domain name system = group of domain name servers

clients and servers

- *server*: computer on a network which carries out some service for another computer
- *server*: the other computer for whom the server is carrying out the service...
- types of servers: domain name server, web server, email server, game server, etc.

open systems

- "a system whose architecture is not a secret"
- prime example: UNIX, LINUX — runs on many hardware platforms; LINUX is free
- *protocol*
  - rules for how computers communicate with each other
  - TCP: transmission control protocol (computer ↔ computer)
  - UDP: user datagram protocol
  - IP: internet protocol
  - HTTP: hypertext transfer protocol (computer ↔ browser)
- *distributed system*
  - multiple computers are distributed geographically
  - example: banking system

ports and sockets

- *port*
  - network conduit on a computer through which a connection to/from other computers is made
  - certain numbers are commonly associated with certain services (see table 2.2); e.g., port 80 → http
  - allows two-way communication
  - NOT a hardware concept (NOT like "USB port" or "printer port")
  - users can define their own network ports and use them for user-specific applications
- *socket*:
  - a network connection implemented in software
  - i.e., a program has to open a "socket" on a computer (using an IP address and a port number) to reach another program on another computer

network concepts

- network topologies
  - bus (linear model; inexpensive to run cables, but not robust to node failure)
  - ring (example: IBM token ring)
  - star (book calls it "hub"; can be expensive to run cables, but robust to node failure)
- layered models: OSI (below) is the classic model

Open System Interconnection (OSI) reference model

- also called the 7-layer model:
  1. application layer (displays data, communicates with lower layers via presentation layer)
  2. presentation layer (link between app and lower layers; converts application layer data to forms understandable by other layers, and back; translates the "meaning" of the bits)
  3. session layer (exchange of data between applications – "dialog" – and synchronization between applications)
  4. transport layer (transfer of data through network; effects flow control; provides some error recovery)
  5. network layer (physical routing of data from one computer to another; facilitates sender finding receiver)
  6. data link layer (manages transmissions of low-level data; detects and corrects transmission errors)
  7. physical layer (sends electronic signals, or "bits" – 0's and 1's) (usually linked to above)
- Here's a phrase to remember the layers: "**A**ll **p**eople **s**eem **t**o **n**eed **d**ata **p**rocessing." where the first letter of each word refers to a layer (above).

- information formats
  - "information units" are passed from one layer to another; "headers" are added as information passes from upper to lower layer
  - terminology is defined below (it is often not used clearly or precisely):

        - *frame* = information unit whose source and destination are the *data link* layer
        - *packet* = information unit whose source and destination are the *network* layer
        - *segment* = information unit whose source and destination are the *transport* layer
        - *message* = information unit whose source and destination are the *application*, *presentation* or *session* layers
        - *datagram* = information unit in a "connectionless" network (see below)

connection-oriented vs connectionless networks

- "connection-oriented"
    - like a telephone
    - first a connection must be established, then data is transferred, finally the connection is closed
    - e.g., TCP
    - monitors for lost packets and re-sends if necessary
    - more overhead than connectionless, but more reliable
- "connectionless"
    - like sending a letter in the mail:
    - there is no guarantee that the recipient ever gets the letter;
    - e.g., UDP
    - less overhead than connection-oriented, and less reliable