Nick Parlante

Networking 2

(Previous Handout)

Vendors: balkanize
TCP/IP Standard -- allow any-to-any networking
- Common language
- IP addr standard
- IP datagram standard
- TCP routing standard
- Built on top of local LAN technology

Ethernet LAN

Connect 2-50 computer near each other
Neat design: distributed, decentralized, unpredictable

Problem:Balkanized LANs

AKA "babel" problem
Many incompatible LANs -- each with its own naming scheme, packet format, etc.
How enable any-any networking?

Solution:Internet Standard

TCP/IP standard -- unifies
"standard" -- the spec is defined publicly and is freely implementable by anyone -- no permission or licensing is required.
A networking standard spoken by all the computers

IP Address

(see previous handout)
Standard namespace
Made of 4 bytes -- each in the range 0..255
e.g. 171.64.64.250
Roughly 4 billion addresses
Ranges of addresses are assigned to organizations -- e.g. Stanford has all of 171.64.*.*
Left "subnet" numbers = neighborhood
Right "host" numbers = computer in that neighborhood
The exact dividing point between the subnet and host parts of the IP addr varies.
IP addr depends on location within the Internet -- moving to a different location in the net may require a different IP addr

IP Packet

Standard packet format -- header field formats, etc.
"Datagram" is basically the same thing
"Best effort" delivery service of a single datagram (packet)

IP Routing

Routing standard for IP datagrams, hop by hop, across the Internet
IP routers carry on a background protocol to measure the current status of links and routers to determine the best current routes and adapt to changing network conditions.
(see the IP Routing example handout)

IP Internet Hierarchy

Roughly tree shaped
"Upstream" router, towards the Internet backbone

TCP / IP -- On Top of LAN

TCP / IP is implemented on top of the local LAN technology
The LAN is used to get the IP datagram one hop, to the next router.
The IP datagram is sent inside the LAN packet.
Each LAN hop may be a different technology may be different

IP Port Numbers

Each IP addr is logically divided into logical port numbers in the range 1-65535.
Traffic is addressed to a specific port number at the IP -- this allows a computer with one IP to be in multiple conversations at once. As we'll see, specific port numbers are reserved for specific purposes -- e.g. port 80 is used for HTTP traffic.
Each end of a connection has a source and destination port number as well as IP addr.

TCP -- End-to-End Control

Built on top of the basic IP best-effort datagram service.
End-end reliability -- reassemble and error check the packets at the destination. Re-send corrupted packets as needed.
Provide the illusion of a continuous, error-free byte stream from the sender to the recipient.
Flow control -- slow down the sender to a packet rate the receiver and the network in between can cope with. The receiver sends "ACK" acknowledgment packets back to the sender to signal which packets have been successfully received.

TCP "Virtual Circuit"

TCP -- Transport Control Protocol

Stream Oriented

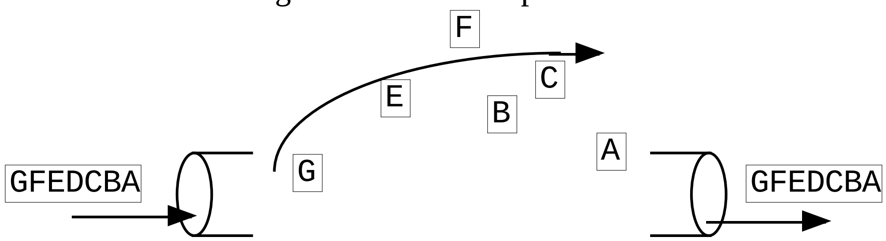
- The writer gets to write bytes to a stream -- like a file.
- The writer can just "print" to the stream as if they were writing to a file.
- TCP breaks the stream up into IP datagrams for transmission, but this detail is hidden from the writer.

Number the packets as they go out

Include a checksum with each packet

Error check and re-assemble the packets at the destination

- Re-build the original stream and present it to the reader



Sending end of TCP virtual circuit

Packets routed through the Internet slightly randomly

Receiving end error checks the packets and re-assembles them into the original stream

Errors

Corrupted packets

Missing packets

Duplicated packets

Out-of-order Packets

TCP Solution

- TCP detects all of these cases and fixes them on its own

TCP "ACK" Strategy

ACK

- TCP uses "acknowledgment" traffic back from the destination to tell the destination which packets have been received correctly.

Re-Send

- The sender can re-send a packet that did not get through

"In flight" window

- The sender will only send so many packets out before some of them are acknowledged. This is the number of "in flight" packets allowed at one time. In TCP/IP terminology this is known as the "window size".

Flow-Control

- The flow of ACK packets also serve to slow the sender down to a rate that the recipient is capable of accepting. If the sender is putting out packets faster than the recipient (or a router on the way) can process, it's a real problem. Error detection is the obvious function of TCP, but actually flow control or (or "flow optimization") is just as important.

TCP Stream Service

Stream

TCP gives the appearance of a stream all the way from the sender to the recipient. It hides the underlying datagram, routing, ack, re-assembly details -- it just looks like stream-in and stream-out to the client.

Stream = FIFO

The writer writes a linear sequence, and the reader will see that same linear sequence.

Bursty Timing

However, TCP cannot hide the irregular "bursty" timing of the traffic.

Irregular Cadence

The bytes may **look** like a stream, but the cadence (timing) that they arrive will be irregular.

TCP 3-Way Handshake

How a TCP stream connection is set up

Each party sends a SYN (request) which is acknowledged (ACK) by the other end. Each party picks a random sequence number (like an id number), that is used by both parties to identify the conversation.

1. Client sends a SYN to the server, including the sequence number the client would like to use
2. Server ACK's the client SYN, and sends its own SYN and sequence number
3. Client ACKs the server SYN (The ACK can be piggybacked on data traffic the client is sending to the server)

Latency

Notice it takes 3 trips minimum for the client to send anything to the server. Therefore, in TCP, we'll have at least 3x single trip latency. The packets involved are tiny -- the latency of the system dominates.

Client Abstraction of TCP/IP

Ignoring Implementation Details

Think about how the Internet looks to a simple computer connected to it -- not thinking about **how** the routers etc. work, but just what they **accomplish**.

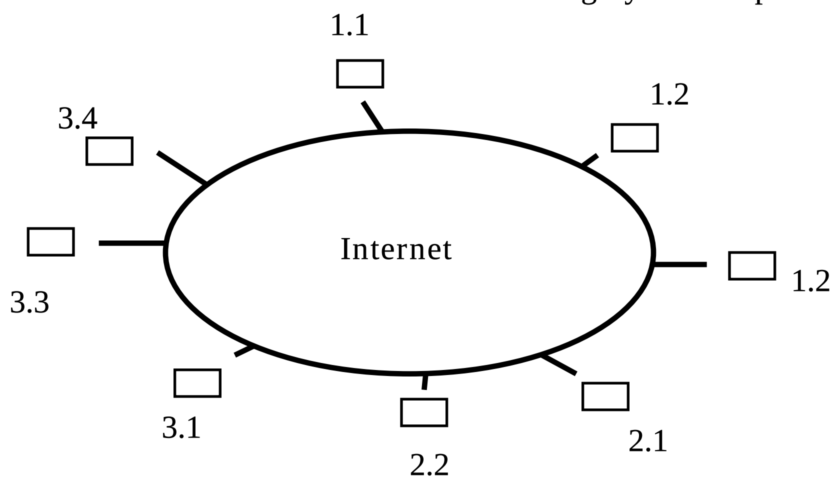
Phone System

IP Addr

The TCP/IP Internet is like a phone system the connects all the computers. Each computer has a phone-number. Any computer can place a call to any other computer. Each computer has its own IP addr

TCP Stream

Through TCP, the sender can write a stream of bytes on their end, and TCP will re-create that stream for reading by the recipient.



Future Topics

Neat aspects of TCP/IP put off until later...
How does dialup PPP work?
How does wireless 802.11 work?
How does NAT work?