ICS 351: Today's plan

- introduction to the HTTP protocol
- introduction to the TCP protocol
- introduction to the IP protocol
- introduction to the Ethernet protocol
- introduction to the 802.11/WiFi protocol

Introduction to the HTTP protocol

- HyperText Transfer Protocol
- data always sent over TCP/IP
- server is usually on port 80, sometimes on other ports (8080, or 443/4433 for secure HTTP)
- protocol format includes a header and an optional body

HTTP example request

- example request:

  GET /~esb/ HTTP/1.1

  Host: www2.hawaii.edu

  Accept: */*

  Connection: close

- in-class exercise: what does this request tell us?
- HTTP data is always encoded using ASCII characters, not binary (e.g. the content length is a decimal number)
  - no longer true in HTTP/2

HTTP example reply

- example reply (beginning only):

  HTTP/1.1 200 OK

  Date: Wed, 03 Sep 2008 04:33:31 GMT

  Server: Apache

  Last-Modified: Sat, 10 May 2008 08:06:17 GMT

  ETag: "de109-170-48255779"

  Accept-Ranges: bytes

  Content-Length: 368

  Connection: close

  Content-Type: text/html

- in-class exercise: what does this reply tell us?

HTTP/2

- HTTP/2 is a new version of HTTP/1.1
- major differences include:
  - headers are compressed and encoded in binary
  - one TCP connection can carry several independent streams of data
    - so responses can be reordered compared to requests
  - new **push** mode for server to send unsolicited content to client
- other than the above, HTTP/2 is intended to be functionally equivalent to HTTP/1.1

Introduction to the TCP protocol

- Transmission Control Protocol
- TCP is designed to run over packet-oriented protocols, such as IP, that don't guarantee to deliver all their packets
- TCP provides the abstraction of a stream of bytes sent reliably end-to-end
- TCP also provides different ports so different server applications can be reached on the same machine (UDP also provides ports)

Introduction to the TCP header

- the TCP header is always encoded in binary big-endian format, and includes:
  - a source and a destination port number,
  - a sequence number for the data carried in the packet,
  - an acknowledgement number

TCP connection setup

- TCP sends three packets to establish the connection: the SYN (client to server), SYN-ACK (server to client), and ACK (client to server)
- tcpdump of the start of a TCP connection to www.hawaii.edu (edited for clarity):

% tcpdump host www.hawaii.edu

18:50:07.566744 IP zero.ics.hawaii.edu.52718 > www.hawaii.edu:

S 1100198413:1100198413(0) win 5840 <mss 1460,sackOK,timestamp 568120014 0,nop,wscale 5>

18:50:07.567633 IP www.hawaii.edu > zero.ics.hawaii.edu.52718:

S 3441676781:3441676781(0) ack 1100198414 win 24616 <nop,nop,timestamp 433923892 568120014,nop,wscale 0,nop,nop,sackOK,mss 1460>

18:50:07.567690 IP zero.ics.hawaii.edu.52718 > www.hawaii.edu:

. ack 1 win 183

18:50:07.568045 IP zero.ics.hawaii.edu.52718 > www.hawaii.edu:

P 1:554(553) ack 1 win 183

Introduction to the IP protocol

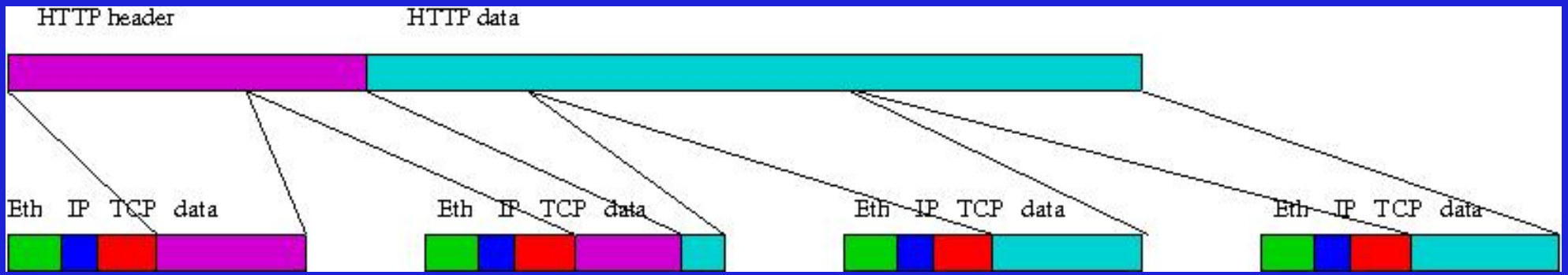
- IP adds a header to the TCP header + payload
- this header includes the source and destination IP addresses, e.g. 123.171.25.102 (IPv4) or fe80::202:2ab:0e15:3223 (IPv6)
- IP then consults its routing table (even on a host) to determine the **next hop** and the **interface to use to reach the next hop**
- the packet is sent to that **next hop** using whatever mechanism is appropriate for the network connected to that **interface**, e.g. Ethernet, WiFi, PPP, etc

Introduction to the Ethernet protocol

- Ethernet adds a header to what comes from IP
- this header includes the source and destination Ethernet (hardware, MAC) addresses, e.g. 01:ab:0e:15:32:23
- an automatically constructed table, the Address Resolution Protocol table (ARP table), is used to determine
  - the Ethernet/MAC address of the next hop, given
  - the IP address of the next hop
- Ethernet performs Medium Access Control (MAC): it has to determine when nobody else is sending on the same medium so it is OK to send a packet

HTTP over TCP over IP over Ethernet

- HTTP always runs over TCP, TCP always runs over IP
- IP can run over many different local area technologies, including PPP, Ethernet, WiFi, etc.
- the TCP protocol splits the application stream so it is carried by successive IP packets (and retransmitting when necessary)
- the HTTP request header is at the start of the TCP connection, the HTTP reply header is at the start of the TCP data sent in the opposite direction



Introduction to the 802.11/WiFi protocol

- 802.11 is in some ways very similar to Ethernet (802.3): the header is added to each IP packet, and it contains the source and destination address
- 802.11 is different in the way MAC is done:
  - a transmitter cannot tell whether its message is colliding with another message
  - so instead, transmitter and receiver may exchange short packets to tell everyone in range to keep quiet for the duration of the data packet
  - the short packets: Request To Send (RTS), Clear To Send (CTS)
  - the data packet may be acknowledged with a short ACK packet
- 802.11 also provides different forms of security: WEP (very weak), or WPA and WPA2 (much better)