ECE 435 – Network Engineering
Lecture 17

Vince Weaver

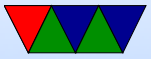
http://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

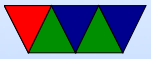
7 November 2016

Announcements

- HW#6 graded, HW#8 on Wednesday
- Plumbers wrapup
  Not much network related, though they were having problems with a SYN flood attack somehow
- Project topics?
- Examples: create a sockets program (game, chat, etc) benchmark speed, try to get best bandwith/latency?

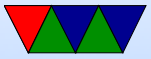


triangulate positio based on wireless router signals?
out of a Pi? USB connector?
set up a mini network/router?
web-proxy that does something funny (upside down?)
physical layer, decode packets at the scope level (hard)
decode packets on fly, packet sniffer, wired or wireless
security: only do these things on isolated networks, not *actual* ones. Try to break wireless passwords, try to spoof IP address, writing a SYN flood tool



HW#7 Wrapup

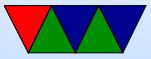
- How did it go?
- Puzzling only got 10-18Mbits/s out of it. It's a gigabit switch and the Pi should be able to saturate a 100Mbit connection (my tests got around 94MBits/s). Hmm.



HW#6

1. IPv6 addressed

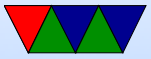
(a) 2607:f8b0:4009:0801:0000:0000:0000:200e
    yes, google
(b) 2607:f8b0:4009:801::200e
    yes, shortened
(c) 2607:f8b0::4009:801::200e
    no, cannot have multiple ::, ambiguous
(d) 123.45.67.189
    no, ipv4



2. ARP

3. BEGIN IPv6 PACKET HEADER

```
0x000e:  6002 2618 :  6 = IPv6
00 = traffic class
2618 = flow label
0x0012:  0031 =       payload length, 49
0x0014:  11 =         next header = 0x11, UDP
0x0015:  40 =         hop limit 0x40, 64
0x0016:  2610 0048 0100 08da 0230 18ff feab 1c3
source address
0x0026:  2001 4860 4860 0000 0000 0000 0000 884
```

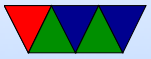


```
destination address

BEGIN UDP HEADER
0x0036:  e239 = source port (decimal)
0x0038:  0035 = destination port (53)
0x003a:  0031 = length (49)
0x003c:  9c0e = checksum
END UDP HEADER
```

4. What type, port 53 UDP = dns

5. netstat: also ss tool



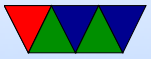
unix

tcp, udp etc

6. traceroute, to portland nox.org, northern crossroads (new england schools) i2, internet2 ams.nl probably amsterdam lod.uk london janet is british academic network 7-¿8 across ocean 80ms = ?? speed of light 80e-3s 3e8m/s = 24000km? 5500km = 1/4 speed of light?

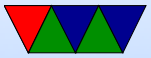
7. traceroute6

   different hops? IP6 different? random chance



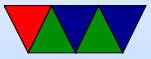
hop 5-¿6

washington? internet 2? abilene was the predecessor to internet2 fra.de frankfurt germany probably not france latency 133ms rather than 106ms

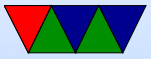


TCP Timer Management

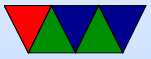
- What should the timer value be? Too short, send extra packets, too long and takes long time to notice lost packets.
- On the fly measures round trip time. When send segment, start timer, updates.
- Connection Timer – send SYN. If no response in time, reset
- Retransmission Timer – retransmit data if no ACK



- Delayed ACK timer – if send a packet, tag an ACK along if timer expires and no outgoing data, have to send stanadlone ACK
- Persist Timer – solve deadlock where window was 0, so waiting, and missed the update that said window was open again
- Keepalive Timer – if connection idle for a long time, sends probe to make sure still up
- FIN_WAIT_2 Timer – avoid waiting in this state forever if other side cashes

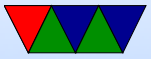


- TIME_WAIT_TIMER – used in TIME_WAIT to give other side time to finish before CLOSE

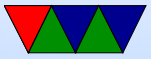


When Things Go Wrong

- Data loss – after retransmit timeout, will notce and retransmit
  If packets just taking a long time, could end up always retransmitting. Data gets in but huge waste of bandwidth. see later.
- ACK loss
- out-of-sequence
- ECN explicit congestion notification – uses extra bits in

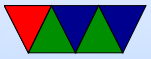


reserved flags, routers initially had trouble with this.



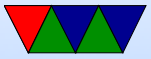
Making things faster

- Offload engines
- CPU speed can be more important than network speed



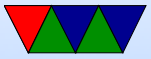
Sockets Programming, Again

- `AF_UNIX`
- `AF_INET`
- Can have lots of other types not TCP/IP
- `SOCK_STREAM` tcp
- `SOCK_DGRAM` udp



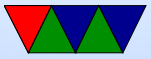
Creating Raw Frames

- Why would you want to?
  ping, traceroute
  also to spoof, ping-of-death, other questionable purposes
- `AF_PACKET`, `SOCK_DGRAM`, `SOCK_RAW` complete control of ethernet header
- Setting promiscuous mode
- Packet filtering, eBPF



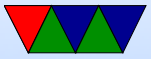
Proposed Replacements

- T/TCP
- SCTP – stream control transmission protocol. More complex
- Whatever google is up to

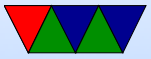


Application Layer

- What do you do with the internet?
- Applications
  - FTP (1971)
  - e-mail (1972)
  - telnet (1982)
  - usenet (1971) NNTP (1986)
  - WWW (1989)
- Utilities, DNS (1987)
- Remote Desktop/X11 forwarding

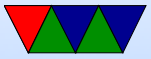


- Gaming (MUDs, etc)
- VOIP (skype, etc)
- Instant Messaging (AIM, ICQ, etc)
- File sharing



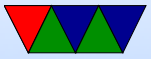
Internet Applications

- Often Client/Server
- Server “daemon”
- Listens on port – IANA “well-known” ports 0-1023, Registered ports: 1024-49151, Dynamic/Private 49152-65535
- Start at boot time? Old days inetd, these days systemd



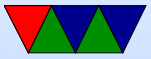
Server Types

- Concurrent – handle multiple connections at time (forks or threads)
- Iterative – handles one connection at a time, rest wait on queue
- Iterative Connectionless – common+trivial, short lived
- Iterative Connection – high latency
- Concurrent Connectionless – when need fast turnaround,



low latency DNS, NFS

- Concurrent Connection – widely used. WWW.



Protocols

- What type of protocol should talk?
- Fixed-length binary?
- Free-form ASCII text?
- 7-bit ASCII vs Unicode?