ECE 435 – Network Engineering
Lecture 12

Vince Weaver

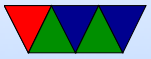
http://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

9 March 2021

Announcements

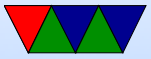
- HW#6 will be posted soon (not due for 2 weeks)
- Midterm on the 16th
- Project will be posted



HW#5 Review

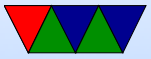
- Header length was the most trouble, top 4 bits of nibble (0x8)
  can sanity check with size.
- Decode the flags
- Timestamp not necessarily actual times, used for more advanced congestion
- Data is ASCII, handy thing to recognize

```
0x0022:  bda5 ____________________ Source port (48549)
0x0024:  0050 ____________________ Destination port (80)
0x0026:  cdc4 6a49 _______________ Sequence Number
0x002a:  3c7b 6ca5 _______________ Acknowledgement Number
```



```
0x002e:  80 ____________________ 1000 header length = 8*4=32
0x002f:  18 ____________________ 11000 ACK+PSH
0x0030:  00e5 __________________ Window Size = 229
0x0032:  79f4 __________________ Checksum = 0x79f4
0x0034:  0000 __________________ Urgent = ?
0x0036:  01       _Option: NOP (padding)
0x0037:  01        _Option: NOP (padding)
0x0038:  080a      _Option: Timestamp, 10 bytes
0x003a:  0104 3e58 _Timstamp TSval
0x003e:  34a8 7bc3 _Timestamp TSecr Echo Reply
```

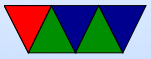
- It's a web request
- Size: 0x46 = 70 bytes, 4/70 = 5.7%
- 3-way handshake SYN/SYN+ACK/ACK
- Sends hi / ack / sends back HI / ack. Note PSH sent



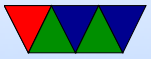
so that it doesn't wait and piggyback

- Closing connection. FIN/ACK+FIN/ACK
- Network connections
  - CLOSE-WAIT: received a FIN and ACKed it, waiting to close

    Only a few, https and imap
  - ESTAB: established, a few ssh, https, imap connections
  - SYN-RECV: way too many, SYN flood
  - TIME-WAIT: connection closed, waiting a bit before re-using port

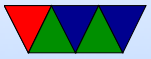


  - UNCONN – UDP listening. 789? ipp, mdns (multicast DNS, bonjour, can find names on network w/o running DNS), lsof -i udp:789, rpcbind
  - LISTEN – listening. Can see ipp (CUPS printing), netbios/microsoft, apparently have SAMBA running,
- Synflood, by default Linux uses SYN cookies to defend against this



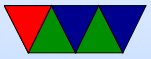
Internetworking

- Metcalf: networks value is the square of the nodes
- Joining networks together of different types
- Might have to convert packets at boundaries
- Or tunnel
- What if packets too big for size limit?
  - Fragmentation (difficult)
  - Path MTU (Maximum Transmit Unit) discovery



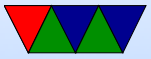
The Internet Protocol v4

- RFC791
- Network of “autonomous systems” interconnected
- Transport layer takes data and breaks into dataframes of up to 64kB. Sent through Internet (possibly broken up) and when get to other side reconstructed by network layer and passed up to transport layer.
- Global and unique address.
- Need hierarchical structure to locate IP address globally



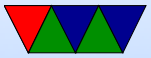
IPv4 Addresses

- Each IP address is 32-bits and has network address and host ID
- Can write many ways: decimal, hex, (all equivalent) but most common is dotted decimal (i.e. `12.34.56.78`)
- Unique to *interface* not necessarily to *host*.



Who Hands these Out?

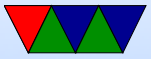
- ICANN and various regional authorities Internet Corporation for Assigned Names and Numbers Internet Assigned Numbers Authority (IANA)
- Regional Internet Registrars
  - AfriNIC (Africa)
  - ARIN (N America),
  - APNIC (Asia-pacific)
  - LACNIC (latin america),
  - RIPE NCC (Europe and rest)



Subnets

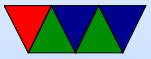
- Number of hosts available can be larger than possible
- Divide network into subnets
- All hosts on subnet have the same prefix (left bits)
- Use subnet mask indicating the leftmost bits to use as subnet
- Can look like 255.255.255.0 meaning only bottom 8 bits are for host

  Alternately can write this as 192.168.8.0/24 (24 is number of leading binary 1s in mask)



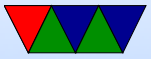
Classful IP Routing (No Longer Used)

- Routers just shifted right for A, B, and C class. Looked up A and B in table, C in hash table to find where to send
- Has a routing entry for each Class A (128), an entry for each class B (16k). Class C (2 million) a bit much, so hash table.
- Why so simple? In 80s memory and processors were expensive!
- Original classful addressing scheme (not used since 1993o



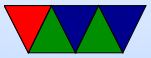
)

- Class A: 8 bit network (high bit 0) (24 bits of hosts) 0.0.0.0 to 127.255.255.255
- Class B: 16 bit network, (high bits 10) 128.0.0.0 to 191.255.255.255
- Class C: 24 bit network (high bits 110) 192.0.0.0 to 223.255.255.255
- Class D: multicast (high bits 1110) 224.0.0.0 to 239.255.255.255
- Class E: reserved (high bits 1111) 240.0.0.0 to 255.255.255.255

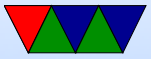


Reserved IP Ranges

- Private Networks
  - 10.0.0.0/8 private network (RFC 1918)
  - 172.16.0.0/12 private network (RFC 1918)
  - 192.168.0.0/16 Private Network (RFC 1918)
- Loopback
  - 127.0.0.0/8 loopback (RFC 6890)
- 0.0.0.0/8 reserved for current network (RFC 6890)
- 100.64.0.0/10 shared address space (RFC 6598)
- 169.254.0.0/16 link-local (RFC 3927)

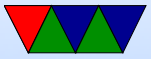


- 192.0.0.0/24 IETF (RFC 6890)
- 192.0.2.0/24 test (RFC 5737)
- 192.88.99.0/24 IPv6 to IPv4 relay (RFC 3068)
- 224.0.0.0/4 IP Multicast (class D) (RFC 5771)
- 240.0.0.0/4 Reserved (class E) (RFC 1700)
- 255.255.255.255 Broadcast (RFC 919)



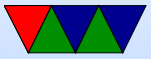
Other IPv4 Conventions

- .0 represents a subnet
- .1 is often (but not always) a router
- it all host bits 1, broadcast for that subnet
- 255.255.255.255 is broadcast for device that doesn't know own IP yet (DHCP)

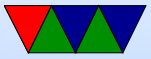


Classless Inter-Domain Routing (CIDR)

- Running out (have run out) of network addresses
- For many groups, Class-A too big, Class-C too small (three bears problem?)
- Merge neighboring class-C together
- RFC 1519
- Scalability problem: each network takes up space in routing table
- Solution, group neighboring class Cs together
- With CIDR bit more complex.

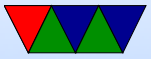


- Triplet with IP address, subnet mask, outgoing line.
- In theory has to scan all. If multiple matches, one with longest mask is used.
- There are algorithms to make this go faster.



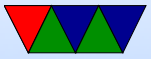
Local IP Routing

- If on same subnet, send packet directly to destination
- Otherwise, send on outgoing. See Linux route command. Often a "default router" 0.0.0.0/0. If doesn't match any other, sent out over default route
- Algorithm: if to same host, skip network. If to same subnet, deliver directly (Ethernet, etc) otherwise, send to default router
- If multiple network interfaces: If to this machine, deliver it, If to directly connected subnet, directly deliver, else



deliver to next hop router

- How do we know if on network? If ((hostIP XOR destip)&subnetmask)==0
- If local, how do we map IP to MAC? We'll see that in a minute.
- Due to CIDR, longest prefix matching. If match both a /21 and /24 then 24 is the one to send to as it's the longest.
- Data structures. Hashes? Trie?
  - Linux: two level hashing
  - BSD - trie (prefix tree)



Linux/UNIX routing setup

- "route" command
- `route add default gateway` sets default gateway (router) for packets leaving the local network
- also set up local subnets you are on, those packets don't need a router
- more complicated if you are configuring your Linux box to *be* a router