

ECE 435 – Network Engineering

Lecture 15

Vince Weaver

`https://web.eece.maine.edu/~vweaver`

`vincent.weaver@maine.edu`

12:30pm, Barrows 125

7 March 2024

Announcements

- Don't forget HW#7 (will extend to Monday)
- Midterms not graded yet



RFC791 Post-it-Note

Internet Protocol Datagram		RFC791 Version <input type="checkbox"/> If other than version 4, attach form RFC 2460.
<input type="checkbox"/> Source	<input type="checkbox"/> Destination	
Type of Service <input type="checkbox"/> high reliability <input type="checkbox"/> high throughput <input type="checkbox"/> low delay	Precedence <input type="checkbox"/> Routine <input type="checkbox"/> Priority <input type="checkbox"/> Immediate <input type="checkbox"/> Flash <input type="checkbox"/> Flash Override <input type="checkbox"/> CRITIC/ECP <input type="checkbox"/> Internetwork Control <input type="checkbox"/> Network Control	Fragmentation Offset <i>Transport layer use only</i> <input type="checkbox"/> more to follow <input type="checkbox"/> do not fragment <input type="checkbox"/> this bit intentionally left blank Identifier _____
Protocol <input type="checkbox"/> TCP <input type="checkbox"/> UDP <input type="checkbox"/> Other _____	Length Header Length <input type="checkbox"/> <input type="checkbox"/>	
Time to Live <input type="checkbox"/>	Options <input type="checkbox"/> <i>Do not write in this space.</i>	Data <i>Print legibly and press hard. You are making up to 255 copies.</i> _____ _____ _____ _____ _____ _____
Header Checksum <input type="checkbox"/>		

for more info, check IPv4 specifications at <http://www.ietf.org/rfc/rfc0791.txt>



HW#6 Review – TCP Header

```
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 _Timestamp TSval
0x003e:  34a8 7bc3 _Timestamp TSecr Echo Reply
```



- Header offset/length was the most trouble, top 4 bits of nibble (0x8) multiplied by 32 can sanity check with size.
- Decode the flags (ACK and PSH)
- Timestamp not necessarily actual times, used for more advanced congestion
- Data is ASCII, handy thing to recognize
- People getting offset, mostly confused by reserved/flags. Count bits. Note things, like checksum is mandatory on TCP so can't be 0.
- It's a web request



- Size: $0x46 = 70$ bytes, $4/70 = 5.7\%$
trouble counting bytes vs nybbles
“useful data” issue this year



HW#6 Review – TCP Connections

- 3-way handshake SYN/SYN+ACK/ACK
note also other things in packet, window scale, sackOK (selective), TS, val, ecr (timestamp: value, echo-reply)
- Sends hi / ack / sends back HI / ack.
Note PSH sent so that it doesn't wait and piggyback
Why is PSH sent? Most(?) TCP stacks when you do a write() will set PSH on the last packet containing data from the write.
- Closing connection. FIN/ACK+FIN/ACK



HW#6 Review – Noticing Congestion

- Timeout
- Multiple duplicate ACKs
- Note: not multiple timeouts
- ECN can notice congestion, but in this case it happens before packets start getting lost (otherwise you'd never get the packets with the ECN info)



HW#6 Review – Security

- 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
Could a lot of legitimate ssh connections cause this?
Unlikely. Have to stop handshake mid-way Why attack



- ssh? Have to try a socket someone is listening on
- 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



Remember Project Ideas Due

- Send e-mail with topic and group members, March 26th (Tues)
- Can work in groups
- Do something interesting network related.
- Can use any operating system and written in any language (asm, C, python, C++, Java, etc.)
- Coding, benchmarking
- Past projects: network games, firewall config, network attached storage, mesh networks, websockets, physical



layer, security

- Will be a final writeup, and then a 10 minute presentation and demo in front of the class during last week of classes.



Hierarchical Routing

- Would you want to have all routers in network on flat network?
Routing table would be a bit complex
- Split into a hierarchy
- Network made up of Autonomous Systems (AS)



Autonomous System (AS)

- A network under control of one group, with one routing policy
- Inside an AS, interior routing, between is exterior routing.
- How to get ASN (number for AS)? Similar to getting IP addresses.
- Usually you need to be a large enough group and be able to get some network connectivity
- Then you need to convince your ISP to add a route to your AS



Autonomous System Numbers

- Traditionally were 16-bit numbers, but ran out. In 2007 expanded to 32-bit. X.Y (dotted decimal). Old 16-bit are 0.X
- Can look up, UMaine is AS557
<https://bgpview.io/asn/557>



Routing

- Systems under same command (same ISP) use intra-domain routing protocol, or interior gateway protocol (IGP)
- Border routers connect to border routers of others
- Inter-domain routing, EGP (exterior gateway protocol)

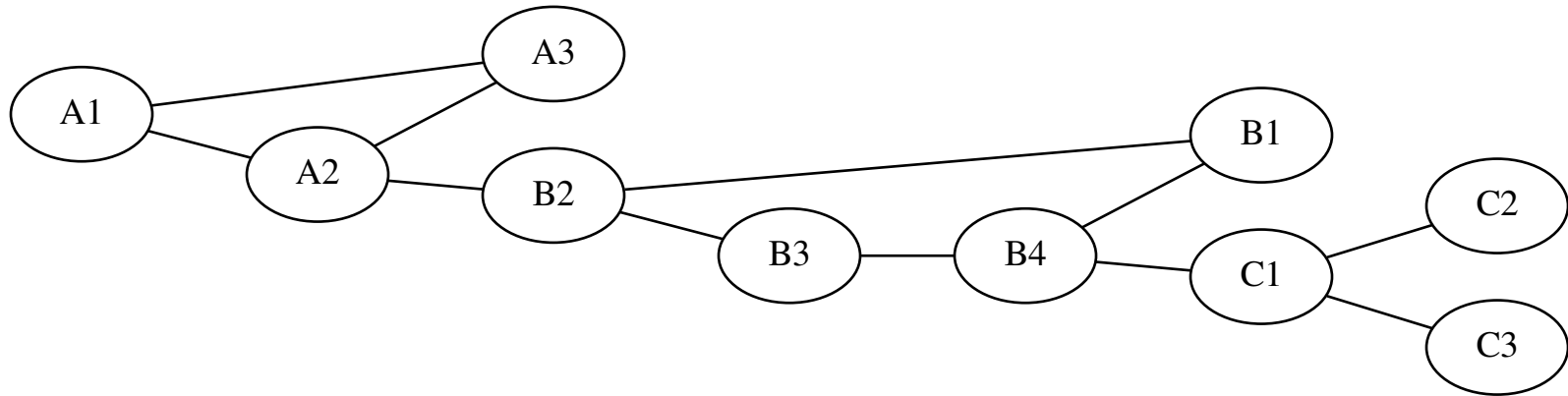


AS Types

- Stub AS – like ISP with customers, one gateway to internet
- Multihomed AS – multiple gateways (why?) redundancy. traffic generally doesn't flow through
- Transit AS – traffic can flow through network
- Internet Exchange Point (IX/IXP) – where networks can meet up



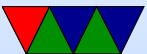
Diagram



- Packet A1 - A3 internal A1 - B2 goes to border router and across, then local A1 - C2 goes to border router to B network, across local to B/C border, then finally to C
- If flat network, need to know 10 machines in routing table



- In hierarchical only need to communicate to 2-3 other routers, find way to border router



Intra-Domain Routing / Interior Gateway Protocols



Historical – RIP (Routing Information Protocol)

- Used by ARPAnet until 1979(?)
- by Xerox, included in BSD, routed RFC 2543
- distance vector routing, with hop count, max 15 hops
 - RIP advertisements over UDP port 52
 - Send advertisement every 30s, or when changes
 - Only sends to neighbors
- Routing table: dest, next hop, distance
- Algorithm



- Get table update
- Increment all hops by 1 (you're one hop away)
- Go down list.

If route not in table, add it

If route there, and next hop same (but cost diff),
replace it as this is new info

If route there but cost less, replace it

- On power up, comes up with hard-coded routes and values of 1 and no next-hop. Can send packet to request immediate update from neighbors.
- Packet description



- Timers
 - Periodic timer, technically 30s, reality randomized between 25 and 35 (why?)
 - Expiration timer – 180s. If no update in this time, problem, hop count set to 16 (unreachable)
 - Garbage collection – 120s – once unreachable, advertise it as such for a while before removing so others notice
- Issues
 - Slow Convergence – a change in routing tables takes 30s per hop to propagate through



Part of why limited to 15 hops

- Instability – packets can be caught in loops. Ways to fix:

Triggered update – send update info immediately, not wait 30s

Split Horizon – if a router sends you update info, don't send this back to it
Poison Reverse – like split horizon, but when send back, mark as 16 the routes received from that interface.

- There was a RIP2



OSPF (Open Shortest Path First)

- successor to RIP. RFC2328 (5340 for IPv6)
- Idea of Areas inside of an AS. Split up into areas.
- Each area connected by backbone router
- Router on two areas is area border router
- Link-state Routing
 - State is flooded: when a change happens (and only then) it sends this state to all neighbors, which send to all neighbors, until the whole network receives it
 - each router uses Dijkstra to find least cost for self,



builds table

- Types of link
 - Point-to-point – routers directly connected
 - Transient Link – network with several routers
can be simplified?
 - Stub Link – a network connected to only one router
 - Virtual Link – a path between two routers that
traverses other routers
- load balancing – supports equal-cost multipath routing
(can equally use equal cost routes)
- supports CIDR routing

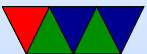


- support available for multicast
- 8-byte password for authentication
- supports hierarchical
- example? complex!



Inter-Domain Routing

- Can be complicated.
- Say company with network, and two connections to outside X,Y. Don't want to send packets out and back even if it looks like lower cost.
- Also don't want to transit packets between X and Y for outsiders. Policy.



BGP (border gateway protocol)

- Intro in 1989, four versions – BGP4 RFC 4271
- Uses TCP (reliable) port 179.
- Works for both IPv4 and IPv6 (the latter as an extension)
- Uses path vector rather than distance vector
 - full path, not just next-hop
 - exchanges info with neighbor, but includes complete path info to avoid looping.
 - Each AS has unique number, so if it sees itself in the path knows there is a loop.



- Policy routing – can also reject new route based on policy
- Four types of messages – open, update, keepalive, notification
- Whole table not passed around (Due to size), only updates
- Due to size of internet, uses distance vector over link state.
- Keeps track of all feasible paths, but only advertises the “best” one



Interior / Exterior BGP

- Interior
 - Interior is a full mesh
 - iBGP makes sure that the setups for multiple gateway routers are kept synchronized
- Exterior
 - eBGP used to talk between other exterior routers at peers.



Routing table Size

- Example. Full BGP of internet backbone router might have more than 300,000 entries (2010) now over 965,000 (2024)
- <http://bgp.potaroo.net/>
- Some routers had limit of 512k so on August 12 2014 part of internet went down when crossed the border.
- Ipv6 currently around 205k (march 2024)



Peering

- How companies agree to connect their networks together.
- There's not really a master connection, but instead companies agree to have routers talk to each other via BGP.
- Types
 - Transit – pay money to pass through network.
 - Peering – In many cases no money changes hands. Why? Well if you have a lot of users, but no content, people won't stay with you. Same if you have content



but no access to users. Averages out and is mutually beneficial.



Peering Analogy

- You and your neighbor both do online games
- You both pay the ISP (maybe per byte) for this, but latency as it goes out to the ISP network and back
- Would it make sense to just run a wire between your houses and have a direct connection for game data?
This is peering and probably mutually beneficial
- Even if it did, would you still maybe not want your neighbor be able to get all their internet access through your network?



Reasons to Peer

- Increased redundancy
- Increased capacity/performance
- Increased routing control
- Fame (high-tier network)
- Ease of requesting aid (?)
- Avoid tromboning (without peering, your connection might go from UMaine to New York, then back to Orono to your apartment if UMaine and your local provider don't peer)



Peering Locations

- Peering locations, often in large data centers.
- Internet Exchange Points (IXP)
- At one point there were 4 major ones (Metropolitan Area Exchange) MAE-East (Virginia) [in basement of parking garage, at one point half of internet went through here], Chicago, NY, SF. All defunct now
- Exchange map: <https://www.internetexchangemap.com/>
- PNI – instead of IXP can just have a direct connection



between two networks



Peering Tiers

- Tier 1 network is one that can reach rest of internet without paying for transit;
- Tier 2 peers with some but purchases for other;
- Tier 3 only purchases



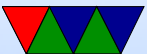
Depeering

- If you think you aren't getting a good deal, break up
- Some situations there is a fight, a hope that the customers lose enough performance will have to repeer.
- Can be a lot of drama



Net Neutrality

- This is a related issue
- Should content providers have to pay ISPs for carrying their packets
- Can ISPs prioritize packets from content providers willing to pay more



IPv6 Peering Issues

- IPv6 Peering issues – see

https://www.theregister.co.uk/2018/08/28/ipv6_peering_squabbles/



Routing Security Issues

- Problems – routing black hole, use BGP to send addresses intentionally to 0.0.0.0 and get dropped. BGP will propagate
- router update mistakes can accidentally blackhole parts of the internet
- In 2008 Pakistan was trying to blackhole Youtube and accidentally announced to world via BGP and took it down world wide



- <https://arstechnica.com/information-technology/2019/06/bgp-mishap-sends-european-mobile-traffic-t>
- https://www.theregister.co.uk/2019/06/24/verizon_bgp_misconfiguration_cloudflare/ Verizon accidentally routed a lot of internet through Pittsburgh Steel Mill
- October 4 2021 – Facebook dropped off internet for 6 hours, DNS took down the BGP links. Had trouble getting back up, including story that they couldn't get card access to datacenter due to internet being down
- March 2022, part of twitter routed through Russia



- Resource Public Key Infrastructure and Route Origin Authorizations can help



Implementations

- Actual Router
- Can install on your Linux machine
- Zebra was traditional, discontinued
- Quagga
- BIRD
- OpenBGPD and OpenSPFD
- Potentially dangerous to mess around with unless you isolate your network well



Other types of Routing

- Mobile – what do you do when machines can come and go?
have a “home” location. Packets go there. When you get on network, update with actual location. Network gets packets at home location, encapsulates and sends to actual location
- Ad Hoc Routing
Bunch of machines in an area, routers and devices can come or go more or less randomly.



route discovery

- Peer to Peer File Sharing

- Centralized server? Napster? Easy to take down.
- Want Distributed, no central control.
- Flooding: connect to one other connected node. Floods requests (sort of like broadcast) until it finds who has file, then direct connect to transfer.
- distributed hash table

- Secret routing



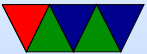
TOR / The onion Router

Packet encrypted multiple times, in layers. Randomly sent to next machine which decrypts that layer, passed on

At end comes out random “exit node” and drops onto regular internet



Broadcast Routing



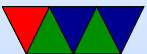
Casting

- Unicast – 1:1 – one sender, one destination
- Broadcast – 1:all
- Multicast – 1:many – specify a subset of all
- Anycast – a set of equivalent hosts, which one gets the packet depends on something like closeness / latency
- Geocast – broadcast to limited geographic area



Anycast problems/benefits

- Can spread server load around (DNS servers, web servers, netflix servers)
- Can hijack connection if you can get your fake routing info into a server.



Unicast / Multicast / Broadcast

- Unicast – send from one machine to another
- What if want to send to multiple?
 - Multi-unicast – open direct connection to each destination. Inefficient
 - Broadcast – send to *every* destination? Waste bandwidth, but also need to know all possible destinations
 - Flooding? Also too much bandwidth
 - Multi-destination routing



Multicast Goals

- Only send to users who want it
- Each member only receives one copy
- No loops
- Path traveled should be optimal



Multicast Structure

- Spanning tree – tree with source as root and members as leaves
- Reverse-path forwarding



Why would you multicast?

- Live streams? Backups?
- Why not just multi-unicast?
 - More work on sender, many more packets sent
 - Latency between first and last packet sent



Multicast IP

- For IP, just join a class D network
- To both sender and receiver it's like sending/receiving a unicast packet
- all the hard work done by routers
- How do you join a multicast group?
- Router two tasks: group membership management, packet delivery.



Group Management

- IGMP (Internet Group Management Protocol)
IGMPv3 RFC 3376
query, report, leave
querier and noquerier
router with lowest IP is querier
no real controls on who can join or send



Multicast Trees

- Steiner tree – NP complete, no one uses
- Heuristics, but none generate entire tree as need centralized and global knowledge
- DVMRP (Distance-Vector Routing Protocol) original protocol, MBONE (tell story)
- Reverse path Forwarding – flood packet out all interfaces except one it came in on. Can have loops; drop dupes.



Then forward on the one that has traveled the shortest path.

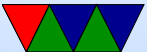
Is running the routing table backwards

- Reverse path Broadcast – avoid getting multiple packets
- Protocol Independent Multicast (PIM)
DVRMP not scalable for multicast groups with sparse members
- MOSPF
- CBT



Local Network Broadcasts

- 224.0.0.0/4 was reserved from Class D for multicast
- 224.0.0.0 to 224.0.0.255 for local network broadcasts
- Things like cluster stats (ganglia, can never get to work?)
- Routing info protocol (RIPv2) OSPF, mDNS, etc.



mDNS

- Multicast local network hostname resolution
- Bonjour (mac), Avahi (Linux)
- Multicast to 224.0.0.251 (ipv4) or ff02::fb (ipv6)
- Issue if two machines have same name
- Broadcast name as connect to network, all devices on local net subscribe to broadcast at that address

