CS 43: Computer Networks

NATs & IPv6

November 16, 2026



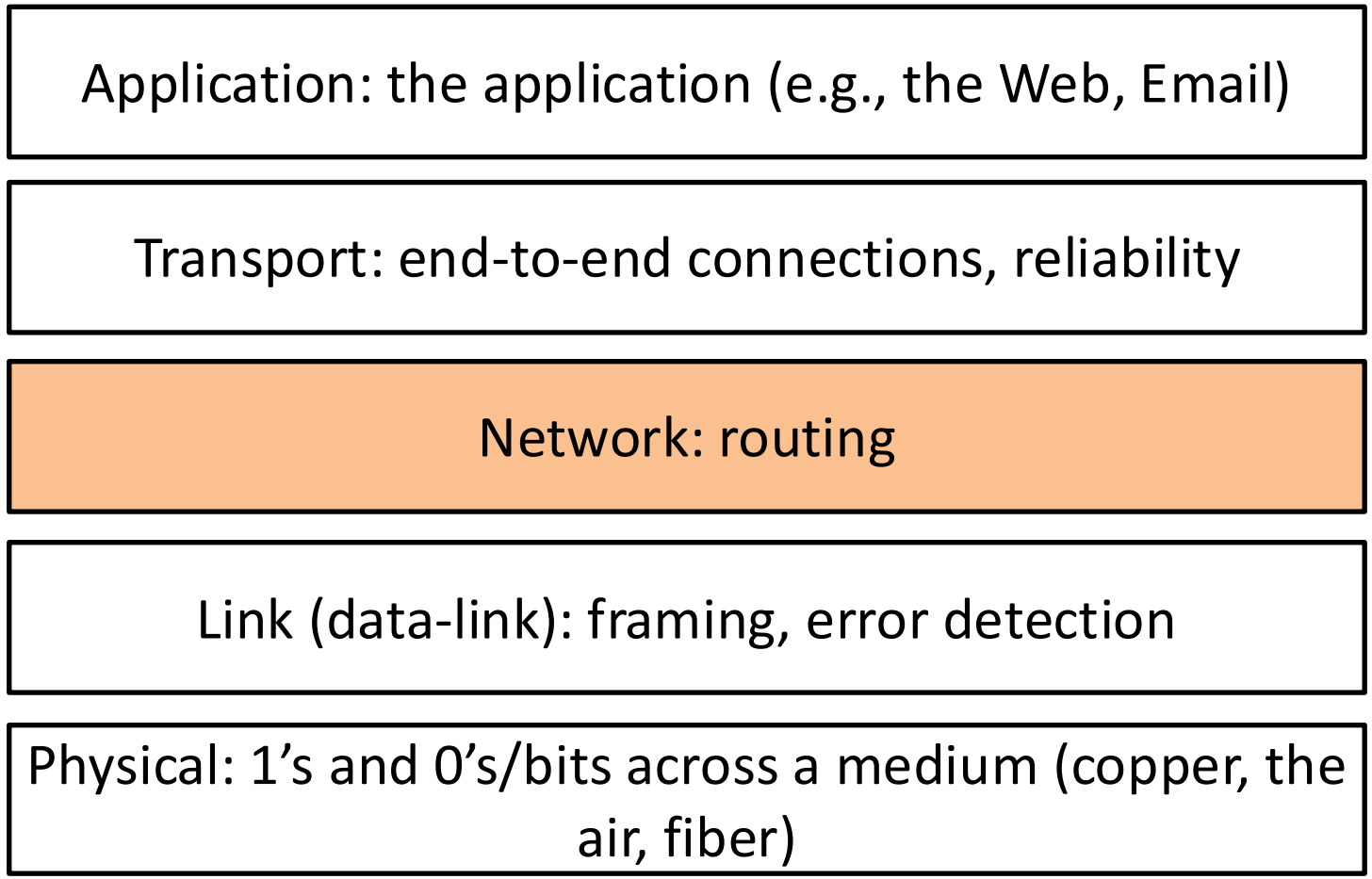
Reading Quiz
(graded for correctness)

Put devices away

Note: Reading quiz slides will have an orange border.

End of Reading Quiz!

The Network Layer!



Recall: IPv4 Addresses

- 32-bit number, must be globally unique
- 2^{32} => 4,294,967,296 possible addresses
- How many do you have?



CORE NETWORKING AND SECURITY

By Scott Hogg

About

Scott Hogg is the CTO for Global Technology Resources, Inc. (GTRI). Scott provides network engineering, security consulting, and training services to his clients.

OPINION

ARIN Finally Runs Out of IPv4 Addresses



IPv4 Address Cupboards are Bare in North America.

Network World | Sep 22, 2015 7:25 AM PT

RELATED TOPICS

Internet

Cisco Subnet | IPv6

6 COMMENTS

It is often said, "the Internet is running out of phone numbers," as a way to express that the Internet is running out of IPv4 addresses, to those who are unfamiliar with Internet technologies. IPv4 addresses, like phone numbers are assigned hierarchically, and thus, have inherent inefficiency. The world's Internet population has been growing and the number of Internet-connected devices continues to rise, with no end in sight. In the next week, the American Registry for Internet Numbers (ARIN) will have exhausted their supply of IPv4 addresses. The metaphorical IPv4 cupboards are bare. This long-predicted Internet historical event marks opening a new chapter of the Internet's evolution. However, it is somehow anti-climactic now that this date has arrived. The Internet will continue to operate, but all organizations must now accelerate their efforts to deploy IPv6.

INSIDER

Network jobs are hot; salaries expected to rise in 2016

Wireless network

ARIN IPv4 Address Exhaustion

The Internet Assigned Numbers Authority (IANA) delegates authority for Internet resources to the five RIRs that cover the world. The American Registry for Internet Numbers (ARIN) is the Regional Internet Registry (RIR) for the United States, Canada, the Caribbean, and North Atlantic islands. ARIN has been managing the assignment of IPv4 and IPv6 addresses and Autonomous System (AS) numbers for several decades. Each RIR has been managing their limited IPv4 address stores and going through their various phases of exhaustion policies. ARIN has been in Phase 4 of their IPv4 depletion plan for more than a year now. ARIN will soon announce that they have completely extinguished their supply of IPv4 addresses.

RELATED



An insider's guide to the private IPv4 market

Techniques for Prolonging the Lifespan of IPv4



ARIN's registry and transfer policies can help bridge the gap from IPv4 to IPv6

on IDG Answers

If I buy a Chromebook and can't get to grips with OS can I convert to windows?

Seriously, we're done now. We're done

Exhausted with never-ending internet exhaustion

By Kieren McCarthy in San Francisco 15 Feb 2017 at 23:07 214 SHARE ▼



You may have heard this before, but we are really, really running out of public IPv4 addresses.

This week, the regional internet registry responsible for Latin America and the Caribbean, LACNIC, announced it has moved to "phase 3" of its plan to dispense with the remaining network addresses, meaning that only companies that have not received any IPv4 space are eligible. There is no phase 4.

That means LACNIC is down to its last 4,698,112 public IPv4 addresses (although that may increase as it recovers a little bit of space over time).

Data Centre ▸ Networks

OK, this time it's for real: The last available IPv4 address block has gone

Now for the last time, will you all please shift to IPv6?!

By Kieren McCarthy in San Francisco 18 Apr 2018 at 22:10 211 SHARE ▼



You may have heard this one before, but we have now really run out of public IPv4 address blocks.

The Internet Assigned Numbers Authority – the global overseers of network addresses – said it had run out of new addresses to dish out to regional internet registries (RIRs) in 2011. One of those RIRs, the Asia-Pacific Network Information Centre, said it was out of available IPv4 addresses later that year.

Then Europe's RIR, Réseaux IP Européens aka RIPE, ran dry in September 2012, followed by the Latin America and Caribbean Network Information Centre (LACNIC) in June 2014. Next, the American Registry for Internet Numbers hit an IPv4 drought in September 2015.

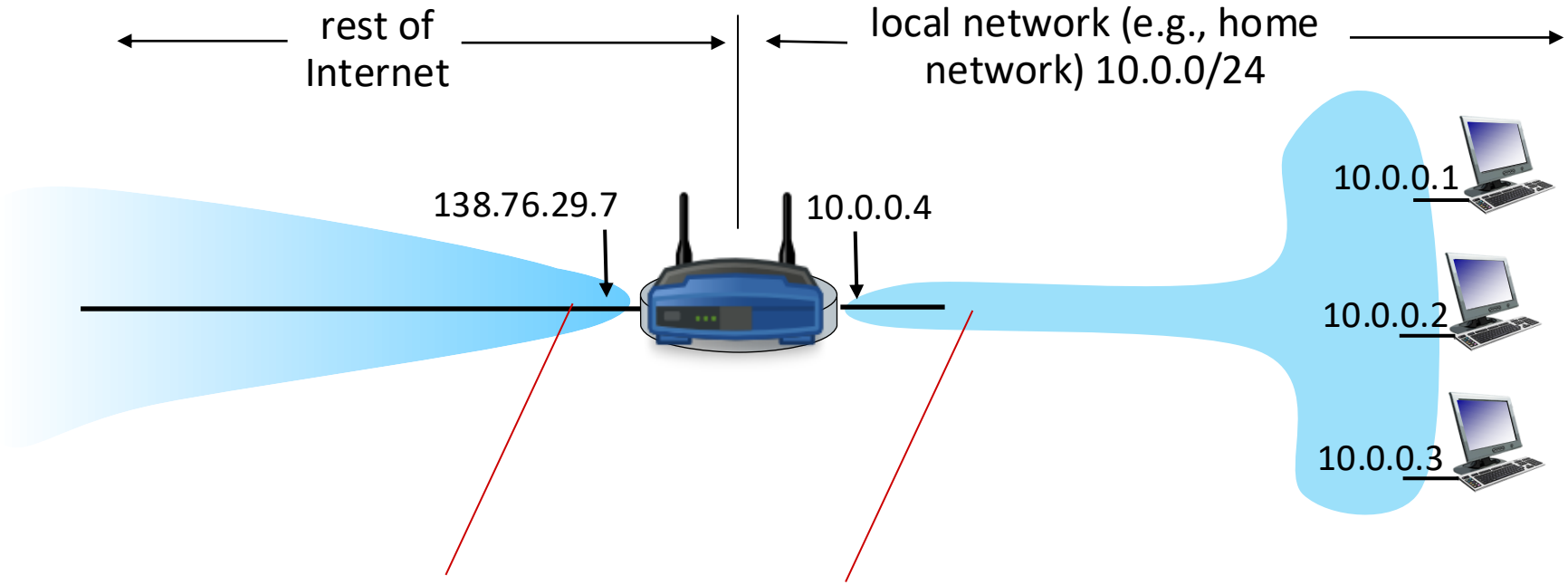
Private Addresses

- Defined in RFC 1918:
  - 10.0.0.0/8
  - 172.16.0.0/12
  - 192.168.0.0/16

- These addresses shouldn’t be routed.
  - Anyone can use them.
  - Often adopted for use with NAT.

NAT: network address translation

NAT: all devices in local network share just one IPv4 address as far as outside world is concerned

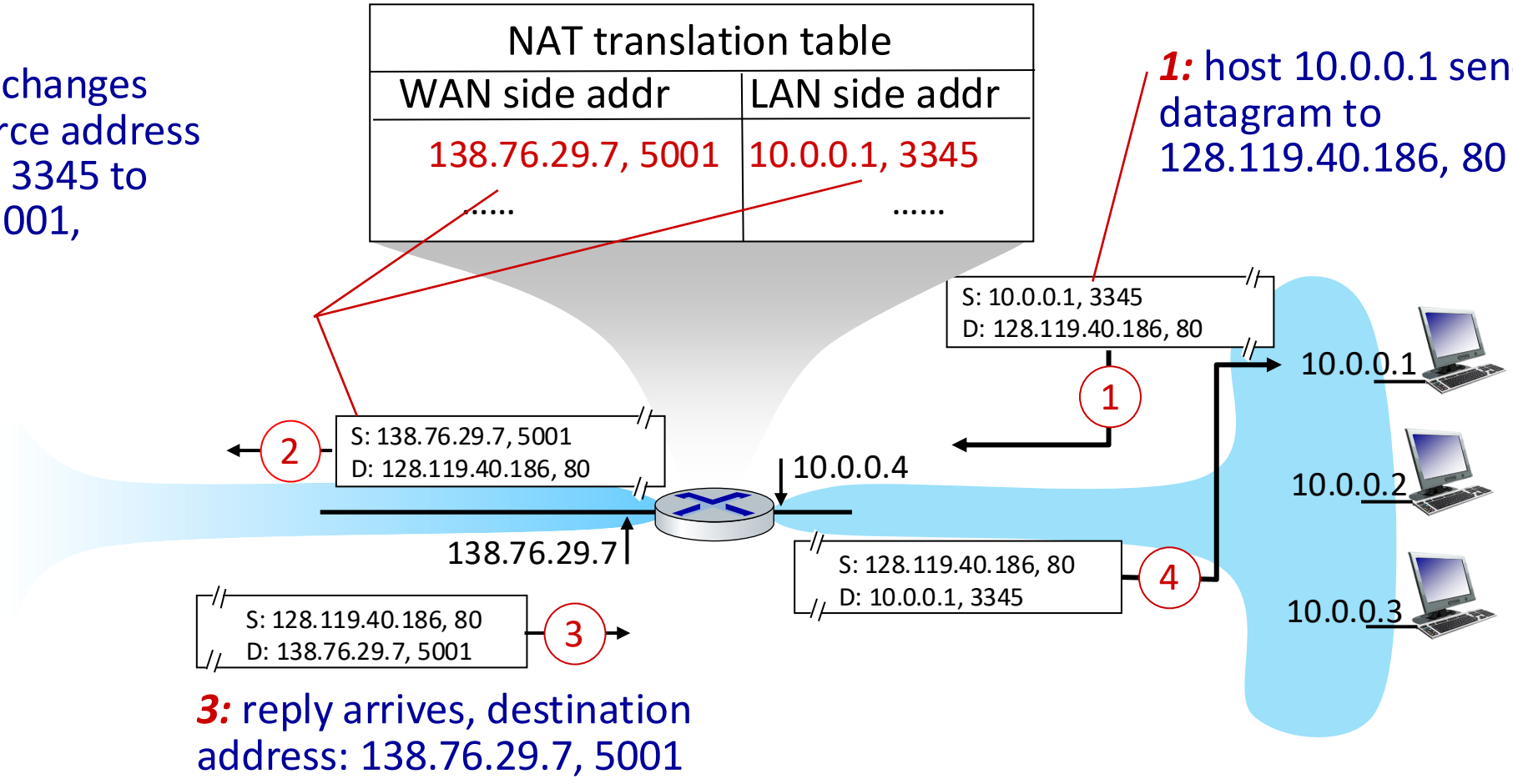


all datagrams *leaving* local network have *same* source NAT IP address: 138.76.29.7, but *different* source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

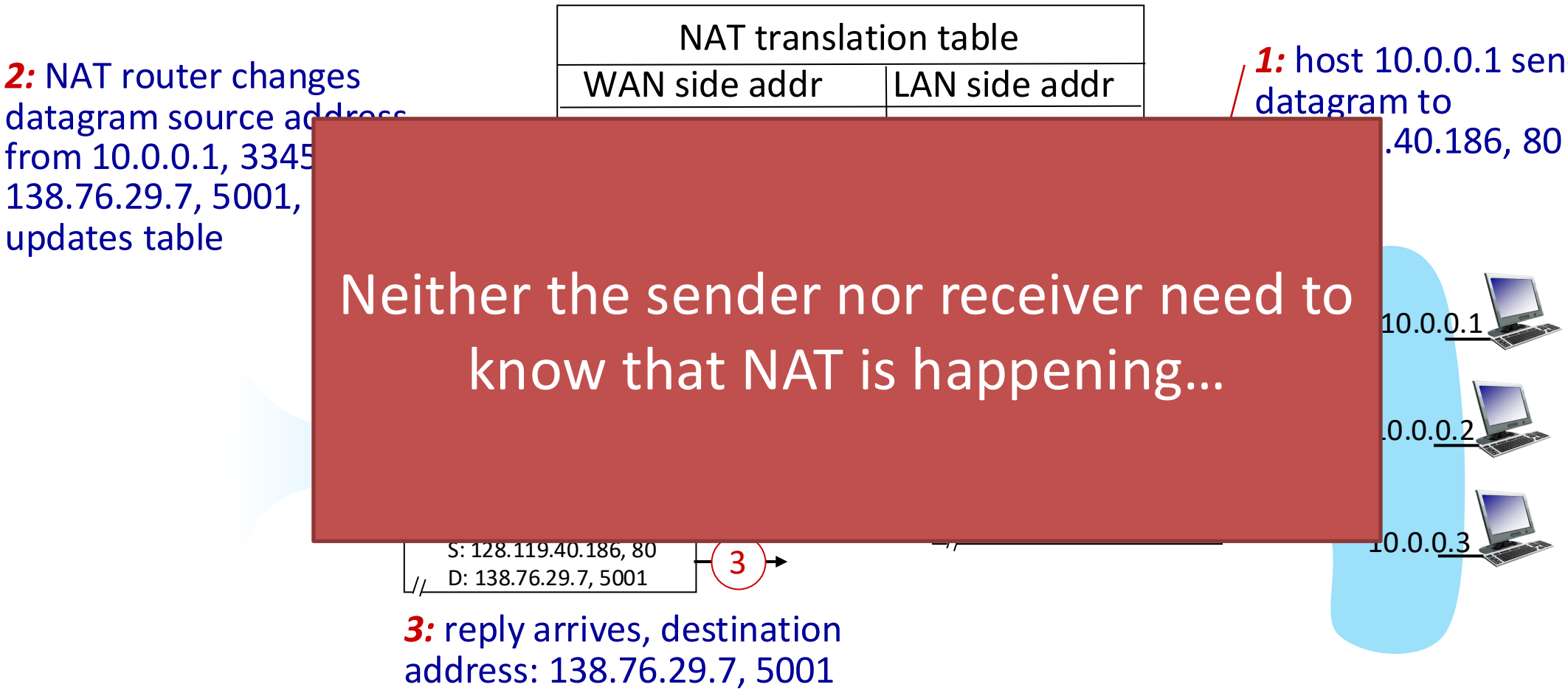
NAT: network address translation

2: NAT router changes datagram source address from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table



3: reply arrives, destination address: 138.76.29.7, 5001

NAT: network address translation



NAT Worksheet

A second laptop at the same home, 192.168.10.12, also connects to 138.76.29.8 port 80, and it happens to pick source port 53924 too, while the first connection is still open. What does the NAT router do?

A. Refuses the second connection because outside port 53924 is already in use
B. Sends both out from 203.0.113.254:53924. The server tells them apart by source IP
C. Makes sure the two leave with different source ports and adds a second table entry
D. Tells 192.168.10.12 to pick a new port and resend
E. I don’t know

Discussion:

A second laptop at the same home, 192.168.10.12, also connects to 138.76.29.8 port 80, and it happens to pick source port 53924 too, while the first connection is still open. What does the NAT router do?

A. Refuses the second connection because outside port 53924 is already in use
B. Sends both out from 203.0.113.254:53924. The server tells them apart by source IP
C. Makes sure the two leave with different source ports and adds a second table entry
D. Tells 192.168.10.12 to pick a new port and resend
E. I don’t know



A second laptop at the same home, 192.168.10.12, also connects to 138.76.29.8 port 80, and it happens to pick source port 53924 too, while the first connection is still open. What does the NAT router do?

A. Refuses the second connection because outside port 53924 is already in use
B. Sends both out from 203.0.113.254:53924. The server tells them apart by source IP
C. Makes sure the two leave with different source ports and adds a second table entry
D. Tells 192.168.10.12 to pick a new port and resend
E. I don’t know

Implementing NAT

- Two hosts communicate with same destination
  - Destination needs to differentiate the two
- Map outgoing packets
  - Change source address and source port
- Maintain a translation table
  - Map of (src addr, port #) to (NAT addr, new port #)
- Map incoming packets
  - Map the destination address/port to the local host

NAT Advantages

- Organizations need fewer IP addresses from their ISP.
  - With a 16-bit port field, we can put 65535 connections behind one external IP address!
- Organizations can change internal network IPs without having to change outside world IPs.
- Better privacy/anonymization: All hosts in one network get one IP

The second laptop, 192.168.10.12, also runs a web server on port 8000. A friend on campus browses to http://203.0.113.254:8000. The NAT table holds only the entries from the worksheet and PI 2. What happens?

A. The NAT forwards it to 192.168.10.12:8000 because only one host runs a server
B. The NAT sends a copy to every host on the home network
C. The friend's browser connects directly to 192.168.10.12:8000
D. The NAT drops it, because no table entry matches port 8000
E. I don’t know

Discussion:

The second laptop, 192.168.10.12, also runs a web server on port 8000. A friend on campus browses to http://203.0.113.254:8000. The NAT table holds only the entries from the worksheet and PI 2. What happens?

A. The NAT forwards it to 192.168.10.12:8000 because only one host runs a server
B. The NAT sends a copy to every host on the home network
C. The friend's browser connects directly to 192.168.10.12:8000
D. The NAT drops it, because no table entry matches port 8000
E. I don’t know



The second laptop, 192.168.10.12, also runs a web server on port 8000. A friend on campus browses to http://203.0.113.254:8000. The NAT table holds only the entries from the worksheet and PI 2. What happens?

A. The NAT forwards it to 192.168.10.12:8000 because only one host runs a server
B. The NAT forwards it to 192.168.10.11, the host with the most recent connection
C. The friend's browser connects directly to 192.168.10.12:8000
D. The NAT drops it, because no table entry matches port 8000
E. I don’t know

Principled Objections Against NAT

- Routers are not supposed to look at port #s
  - Network layer should care only about *IP* header
  - … and not be looking at the *port numbers* at all
- NAT violates the end-to-end argument
  - Network nodes should not modify the packets
- IPv6 is a cleaner solution
  - Better to migrate than to limp along with a hack

NATs are an example of a middlebox

A middlebox is a networking device that transforms, inspects, filters, and manipulates traffic for purposes other than packet forwarding.

Includes firewalls, network address translators (NATs), load balancers, and deep packet inspection (DPI).



Middleboxes are everywhere in the Internet!

How do we feel about NAT?

A. NAT is great! It conserves IP addresses and makes it harder to reach non-public machines.

B. NAT is mostly good, but has a few negative features. No big deal.

C. NAT is mostly bad, but in some cases, it's a necessary evil.

D. NAT is an abomination that violates the end to end principle, and we should not use it!

IPv6

- Initial motivation: 32-bit address space soon to be completely allocated, any day now™.

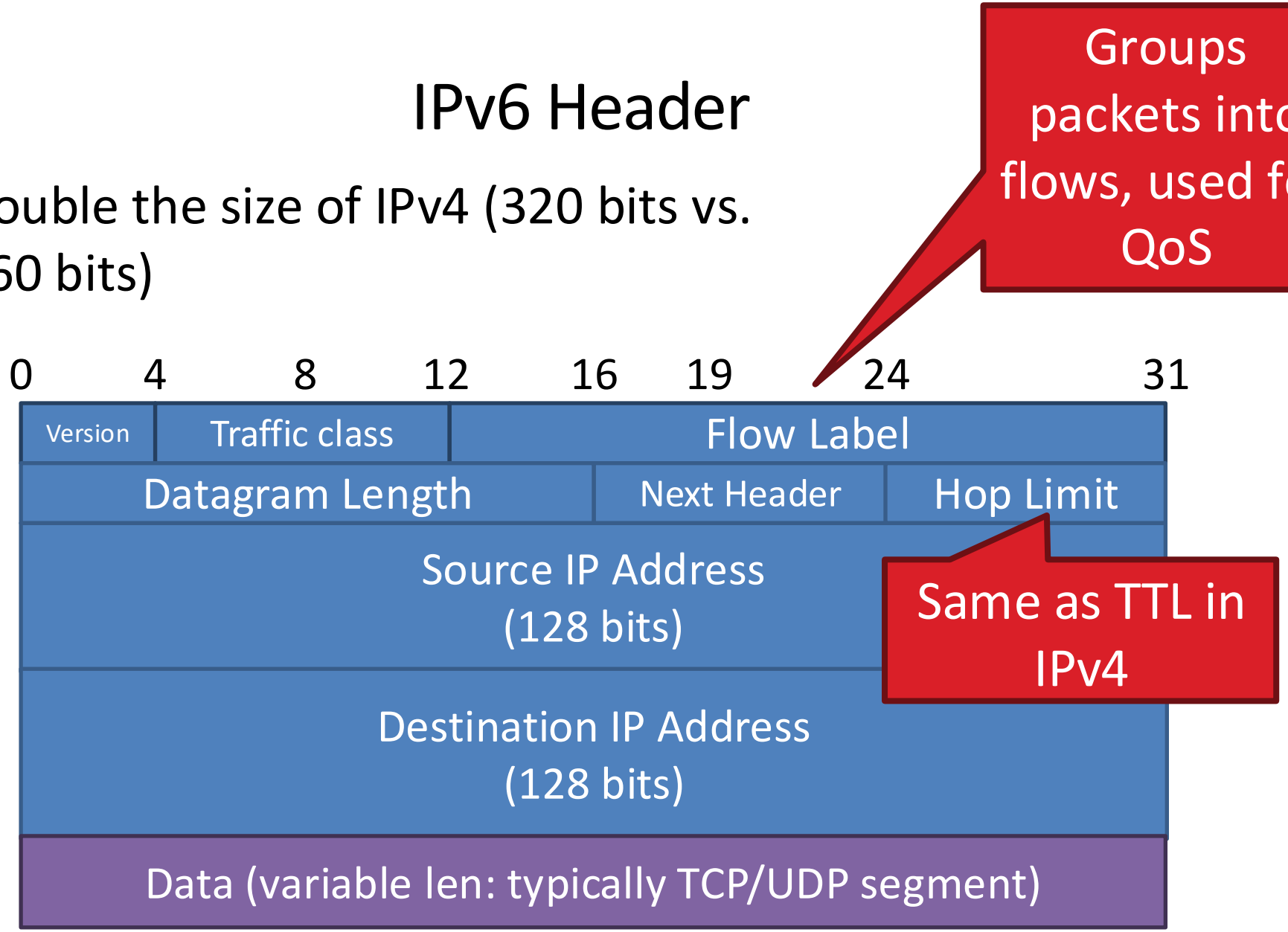
- Additional motivation:
  - header format helps speed processing/forwarding
  - header changes to facilitate QoS

IPv6 datagram format:

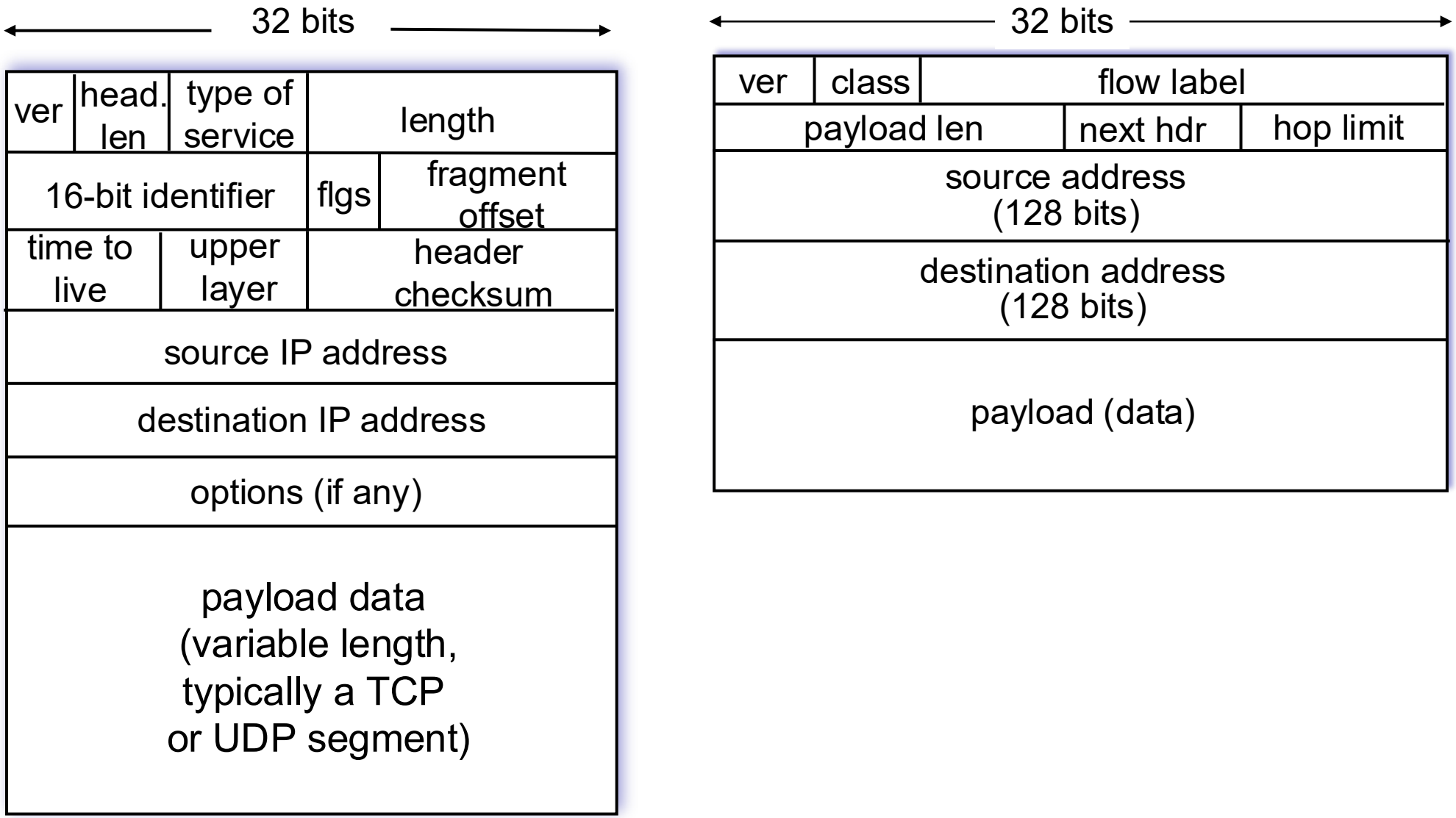
  - fixed-length 40 byte header
  - no fragmentation allowed at routers

IPv6 Header

- Double the size of IPv4 (320 bits vs. 160 bits)



Comparing IPv4 and IPv6 header



IPv6 is simpler than IPv4

IPv6

removed

- fragmentation
- checksum
- header length

reason

leave problems to the end host

simplify handling
length is fixed

added…

- new options mechanism
- expanded addresses
- flow label

simplify handling

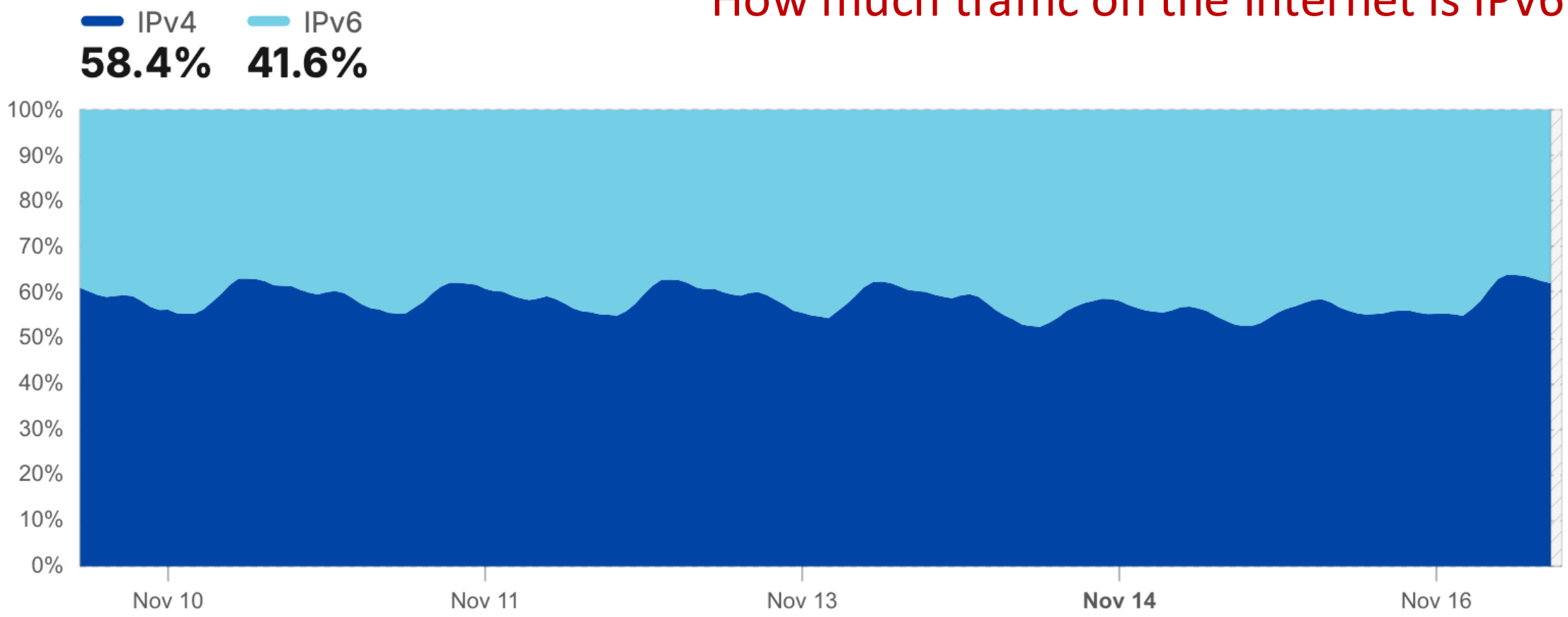
flexibility

IPv4 vs. IPv6 worldwide

Distribution of HTTP requests by IP version

Traffic type: Human traffic

How much traffic on the Internet is IPv6?



CLOUDFLARE RADAR

Last 7 days | Nov 16, 2026, 14:15 UTC

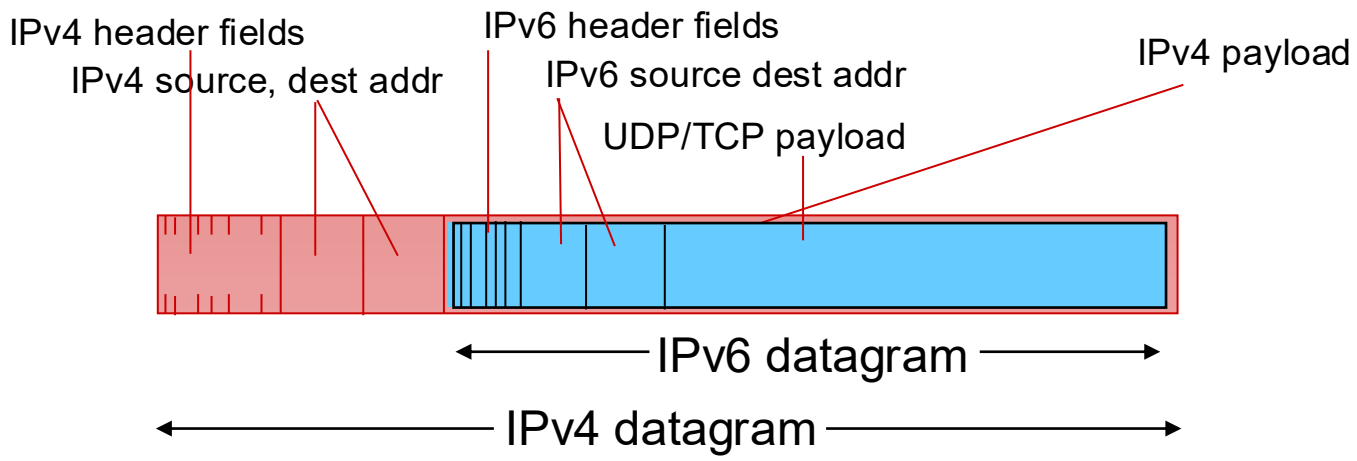
Transitioning to IPv6

- Option 1: “Flag day”
  - How do we get *everyone* on the Internet to agree?
  - Whose authority to decide when?
  - Can you imagine how much would break?

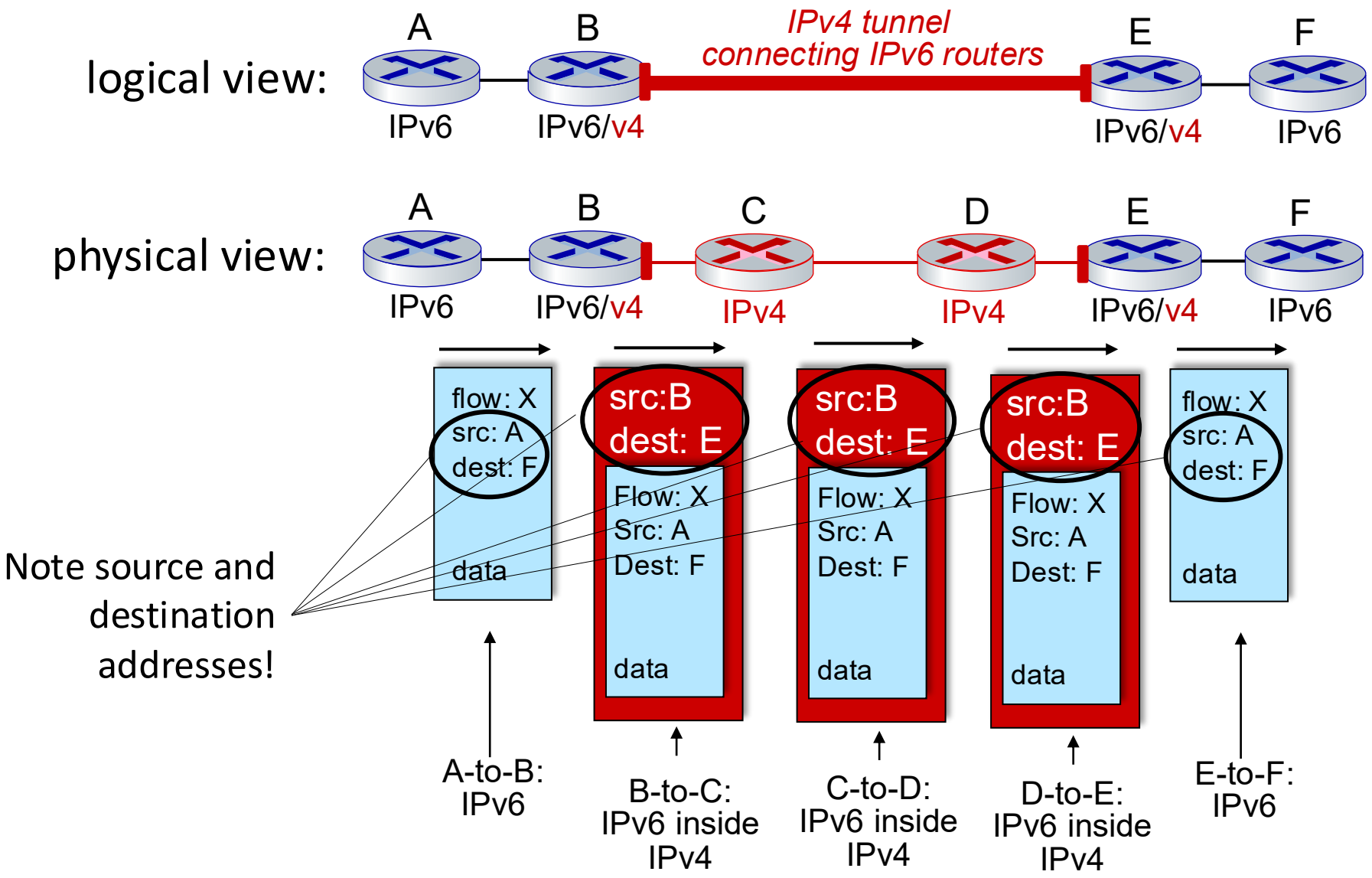
- Option 2: Slow transition
  - Some hosts/routers speak both versions
  - Must have some way to deal with those who don’t
  - Lack of incentive to switch

Tunneling

- IPv6 datagram carried as *payload* in IPv4 datagram among IPv4 routers



Tunneling


logical view:
A
B
IPv4 tunnel
connecting IPv6 routers
E
F
IPv6
IPv6/v4
IPv6/v4
IPv6
physical view:
A
B
C
D
E
F
IPv6
IPv6/v4
IPv4
IPv4
IPv6/v4
IPv6
Note source and
destination
addresses!
flow: X
src: A
dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
flow: X
src: A
dest: F
data
A-to-B:
IPv6
B-to-C:
IPv6 inside
IPv4
C-to-D:
IPv6 inside
IPv4
D-to-E:
IPv6 inside
IPv4
E-to-F:
IPv6

Reference Slides

Other changes from IPv4

- checksum: removed entirely to reduce processing time at each hop
- options: allowed, but outside of header, indicated by “Next Header” field
- ICMPv6: new version of ICMP
  - additional message types, e.g. “Packet Too Big”
  - multicast group management functions

ICMP: Internet Control Message Protocol

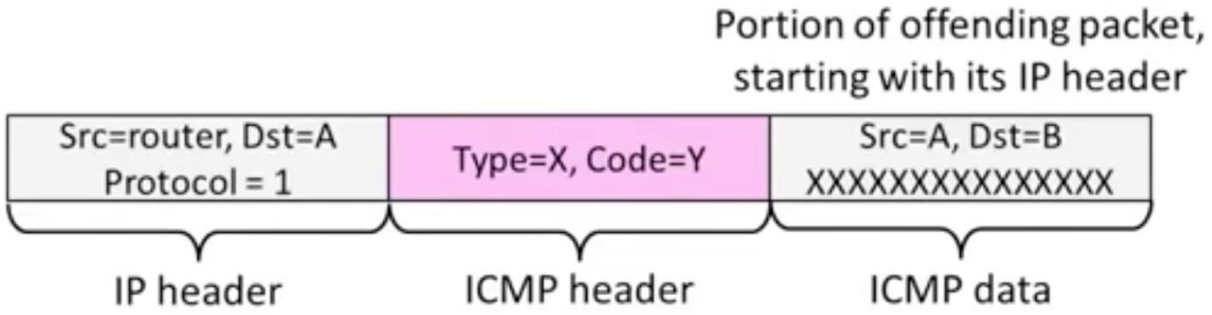
- Service Model
  - Reporting message: self-contained message reporting error
  - Unreliable: Simple datagram service – no retries.

ICMP: Internet Control Message Protocol

- Used to communicate network information
  - "Control messages", i.e., not data themselves
  - Error reporting
    - Unreachable host
    - Unreachable network
    - Unreachable port
    - TTL expired
  - Test connectivity
    - Echo request/response (ping)

ICMP: Internet Control Message Protocol

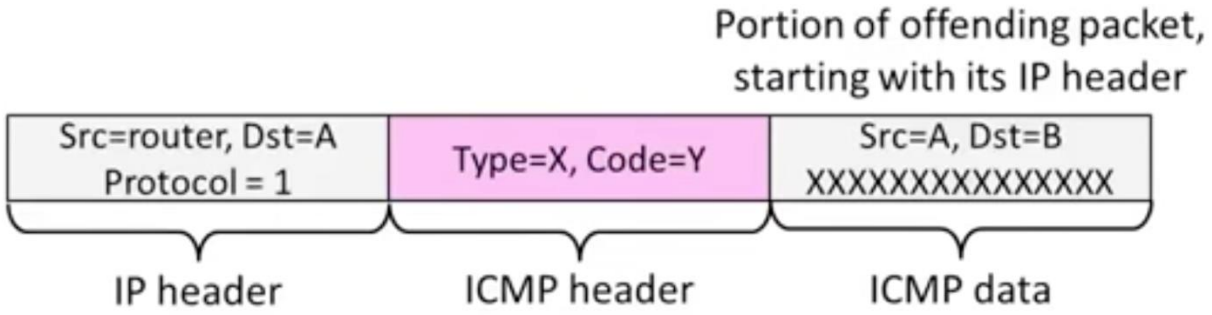
- Header:
  - 1-byte type
  - 1-byte code
  - 2-byte checksum
  - 4 bytes vary by type
- Sits above IP
  - Protocol field = 1 in IP header
  - Usually considered part of IP



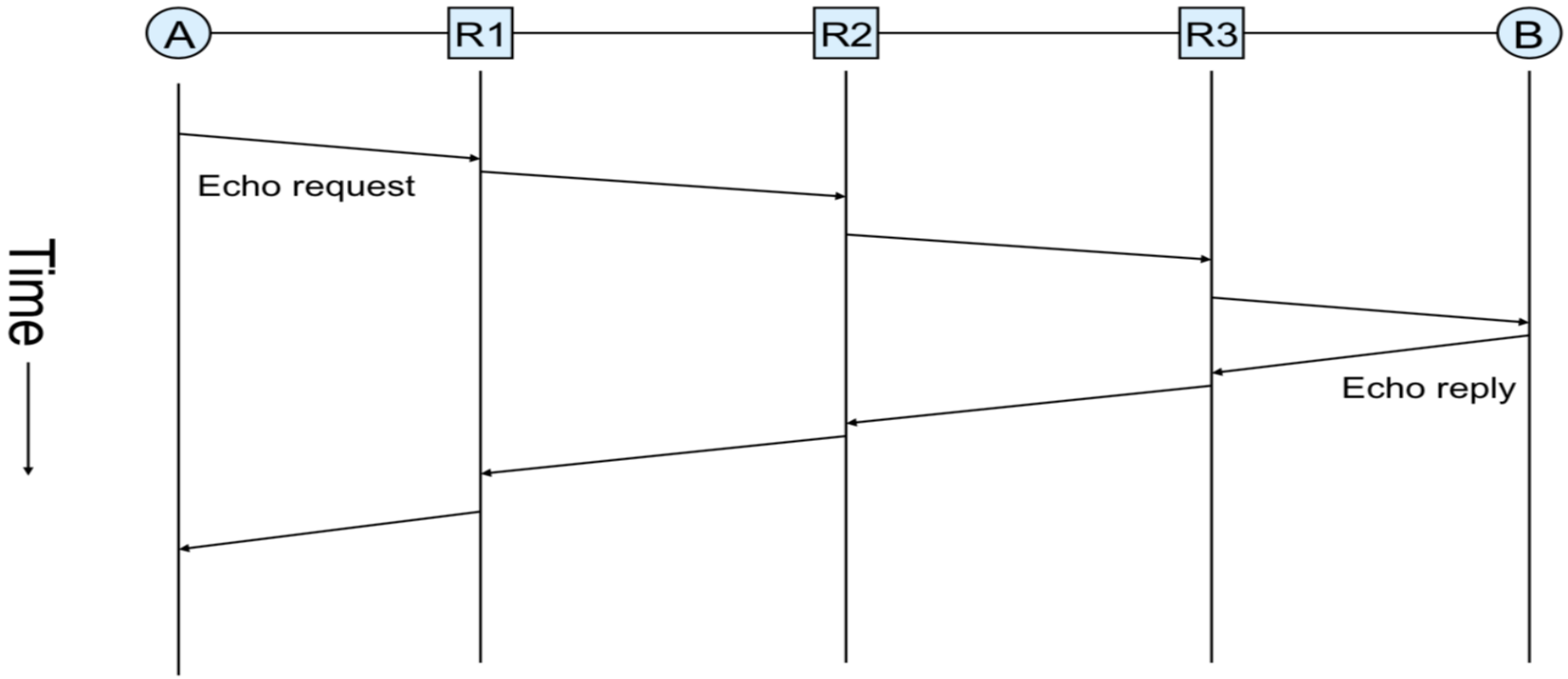
ICMP: Internet Control Message Protocol

Type	Code	Description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown

Type	Code	Description
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

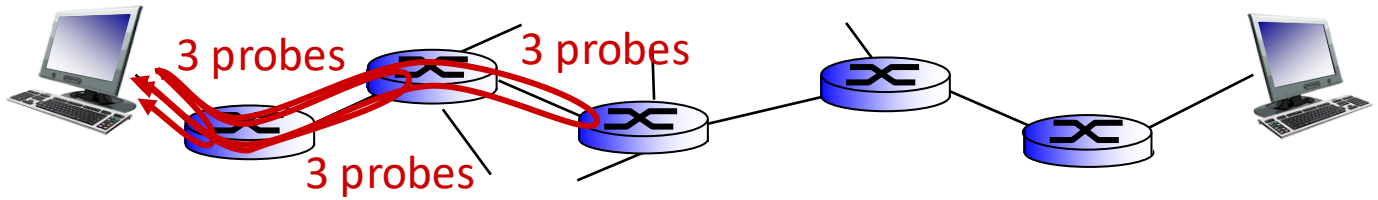


Ping



Traceroute and ICMP

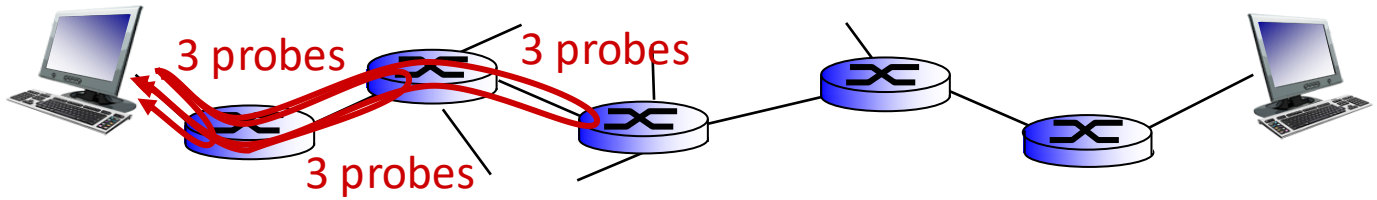
- Source sends sets of UDP segments (usually 3) to dest
  - first set has TTL =1
  - second set has TTL=2, etc.
  - unlikely port number
- When *n*th set of datagrams arrives to nth router:
  - router discards datagrams
  - and sends source ICMP messages (type 11, code 0)
  - ICMP messages includes name of router & IP address
- When ICMP messages arrives, source records RTTs



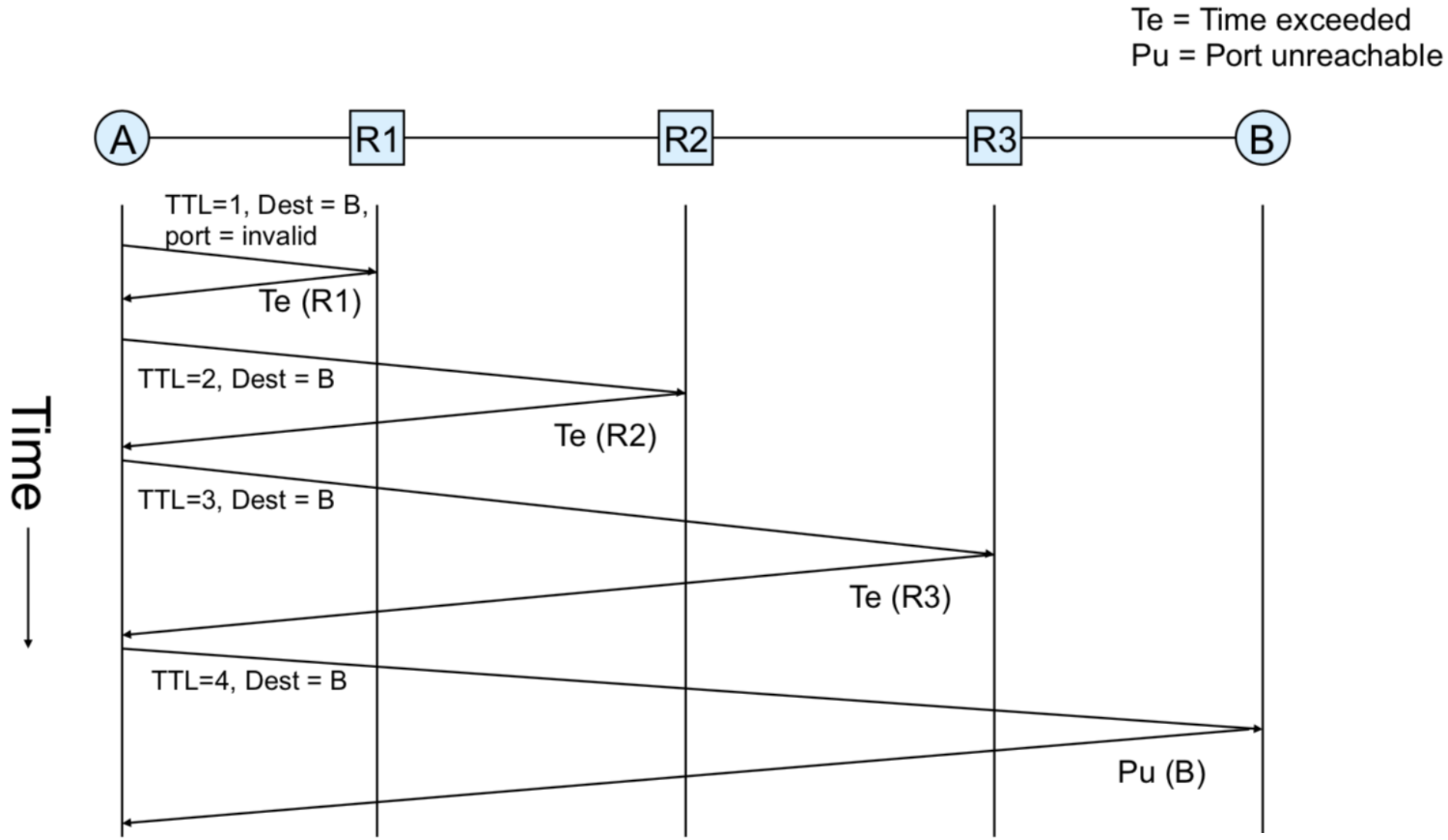
Traceroute and ICMP

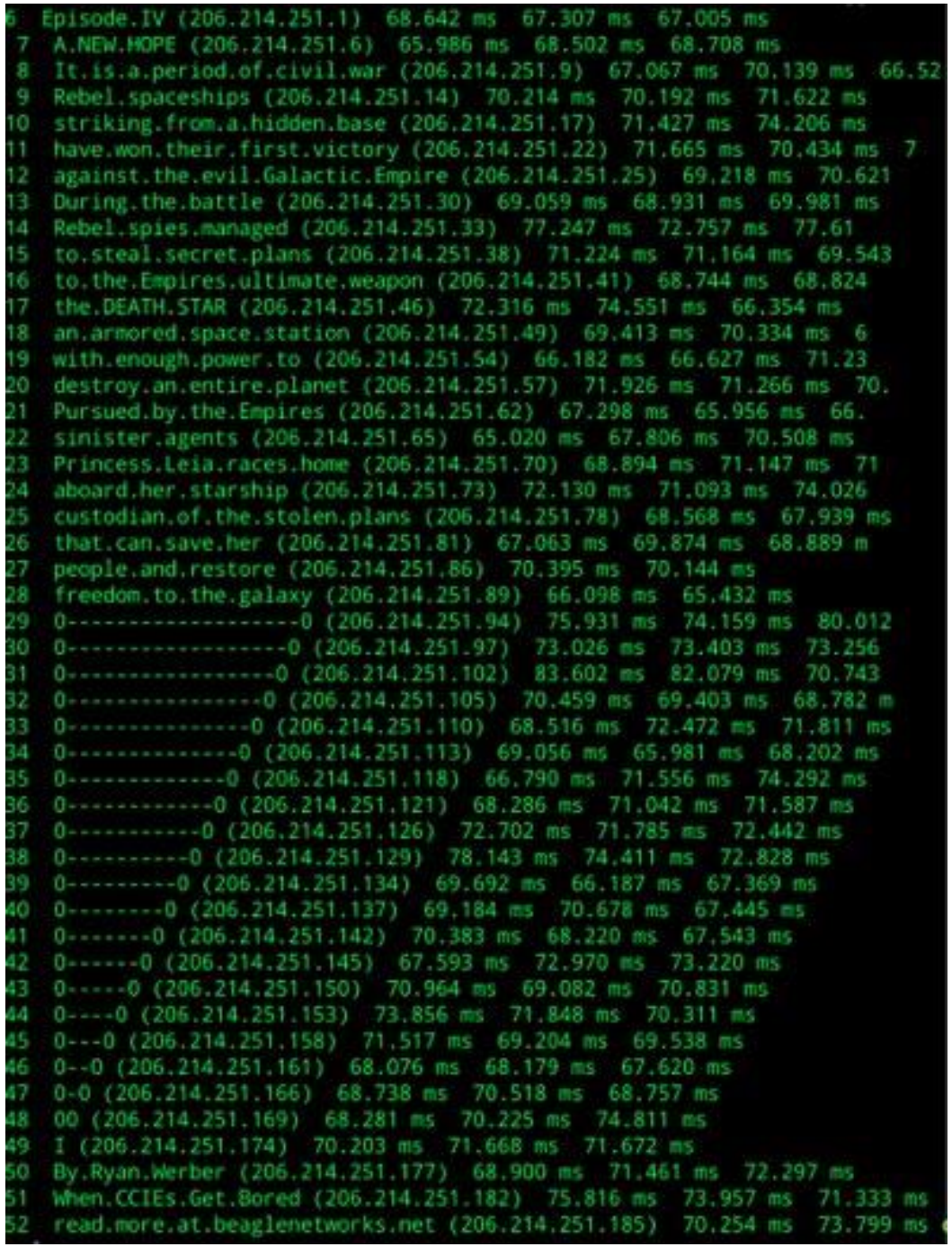
stopping criteria:

- UDP segment eventually arrives at destination host
- destination returns ICMP "port unreachable" message (type 3, code 3)
- source stops



Traceroute Demo




6 Episode.IV (206.214.251.1) 68.642 ms 67.307 ms 67.005 ms
7 A.NEW.HOPE (206.214.251.6) 65.986 ms 68.502 ms 68.708 ms
8 It.is.a.period.of.civil.war (206.214.251.9) 67.067 ms 70.139 ms 66.52
9 Rebel.spaceships (206.214.251.14) 70.214 ms 70.192 ms 71.622 ms
10 striking.from.a.hidden.base (206.214.251.17) 71.427 ms 74.206 ms
11 have.won.their.first.victory (206.214.251.22) 71.665 ms 70.434 ms 7
12 against.the.evil.Galactic.Empire (206.214.251.25) 69.218 ms 70.621
13 During.the.battle (206.214.251.30) 69.059 ms 68.931 ms 69.981 ms
14 Rebel.spies.managed (206.214.251.33) 77.247 ms 72.757 ms 77.61
15 to.steal.secret.plans (206.214.251.38) 71.224 ms 71.164 ms 69.543
16 to.the.Empires.ultimate.weapon (206.214.251.41) 68.744 ms 68.824
17 the.DEATH.STAR (206.214.251.46) 72.316 ms 74.551 ms 66.354 ms
18 an.armored.space.station (206.214.251.49) 69.413 ms 70.334 ms 6
19 with.enough.power.to (206.214.251.54) 66.182 ms 66.627 ms 71.23
20 destroy.an.entire.planet (206.214.251.57) 71.926 ms 71.266 ms 70.
21 Pursued.by.the.Empires (206.214.251.62) 67.298 ms 65.956 ms 66.
22 sinister.agents (206.214.251.65) 65.020 ms 67.806 ms 70.508 ms
23 Princess.Leia.races.home (206.214.251.70) 68.894 ms 71.147 ms 71
24 aboard.her.starship (206.214.251.73) 72.130 ms 71.093 ms 74.026
25 custodian.of.the.stolen.plans (206.214.251.78) 68.568 ms 67.939 ms
26 that.can.save.her (206.214.251.81) 67.063 ms 69.874 ms 68.889 m
27 people.and.restore (206.214.251.86) 70.395 ms 70.144 ms
28 freedom.to.the.galaxy (206.214.251.89) 66.098 ms 65.432 ms
29 0------------------0 (206.214.251.94) 75.931 ms 74.159 ms 80.012
30 0-----------------0 (206.214.251.97) 73.026 ms 73.403 ms 73.256
31 0----------------0 (206.214.251.102) 83.602 ms 82.079 ms 70.743
32 0---------------0 (206.214.251.105) 70.459 ms 69.403 ms 68.782 m
33 0--------------0 (206.214.251.110) 68.516 ms 72.472 ms 71.811 ms
34 0-------------0 (206.214.251.113) 69.056 ms 65.981 ms 68.202 ms
35 0------------0 (206.214.251.118) 66.790 ms 71.556 ms 74.292 ms
36 0-----------0 (206.214.251.121) 68.286 ms 71.042 ms 71.587 ms
37 0----------0 (206.214.251.126) 72.702 ms 71.785 ms 72.442 ms
38 0---------0 (206.214.251.129) 78.143 ms 74.411 ms 72.828 ms
39 0--------0 (206.214.251.134) 69.692 ms 66.187 ms 67.369 ms
40 0-------0 (206.214.251.137) 69.184 ms 70.678 ms 67.445 ms
41 0------0 (206.214.251.142) 70.383 ms 68.220 ms 67.543 ms
42 0-----0 (206.214.251.145) 67.593 ms 72.970 ms 73.220 ms
43 0----0 (206.214.251.150) 70.964 ms 69.082 ms 70.831 ms
44 0---0 (206.214.251.153) 73.856 ms 71.848 ms 70.311 ms
45 0--0 (206.214.251.158) 71.517 ms 69.204 ms 69.538 ms
46 0-0 (206.214.251.161) 68.076 ms 68.179 ms 67.620 ms
47 00 (206.214.251.166) 68.738 ms 70.518 ms 68.757 ms
48 I (206.214.251.169) 68.281 ms 70.225 ms 74.811 ms
49 By.Ryan.Werber (206.214.251.174) 70.203 ms 71.668 ms 71.672 ms
50 When.CCIEs.Get.Bored (206.214.251.177) 68.900 ms 71.461 ms 72.297 ms
51 read.more.at.beaglenetworks.net (206.214.251.182) 75.816 ms 73.957 ms 71.333 ms
52 (206.214.251.185) 70.254 ms 73.799 ms