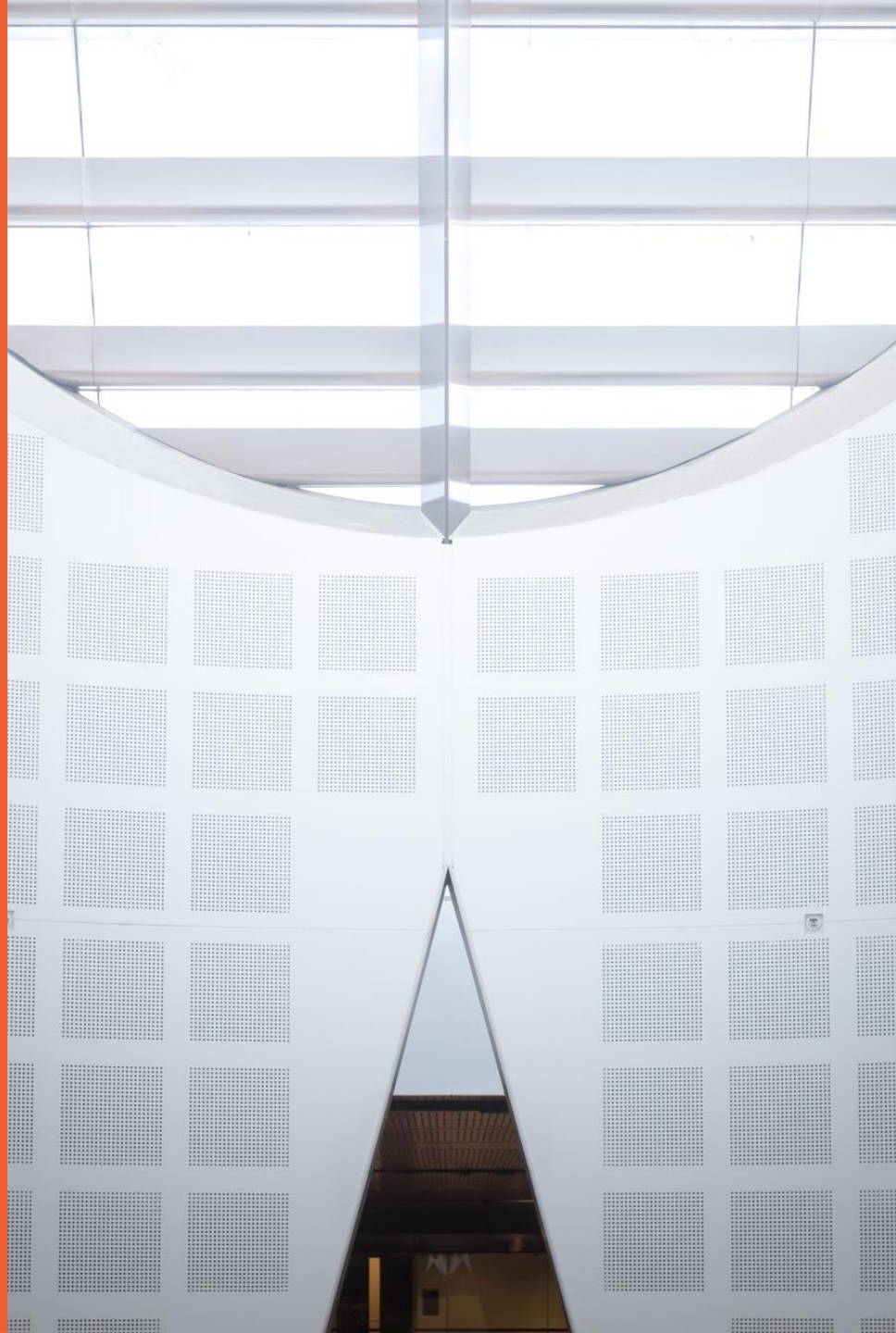
COMP9121: Design of Networks and Distributed Systems

Week 6: Network Layer 3

Wei Bao

School of Computer Science

THE UNIVERSITY OF SYDNEY



Shortest path algorithm

Comparison of LS and DV algorithms

message complexity

Link state

- ***LS:*** with n nodes, E links, O(nE) msgs sent
- ***DV:*** exchange between neighbors only

distance vector

speed of convergence

- ***LS:*** $O(n^2)$ or O(nlogn) algorithm
- ***DV:*** convergence time varies
  - count-to-infinity problem

Hierarchical routing

Hierarchical routing

our routing study thus far - idealization

- all routers identical
- network "flat"

… *not* true in practice

scale: with 600 million destinations:

- can't store all dest's in routing tables!
- routing table exchange would swamp links!

administrative autonomy

- internet = network of networks
- each network admin may want to control routing in its own network

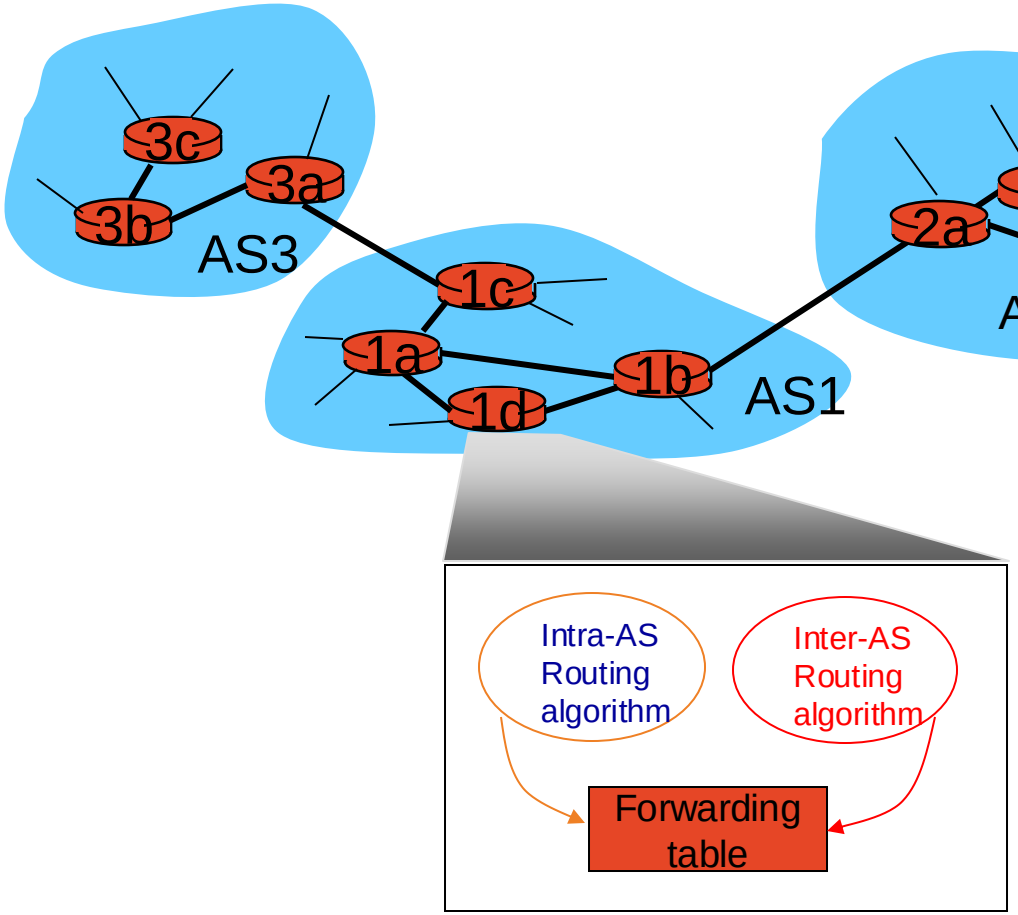
Hierarchical routing

- aggregate routers into regions, "autonomous systems" (AS)
- routers in same AS run same routing protocol
  - "intra-AS" routing protocol: routers in different AS can run different intra-AS routing protocol

gateway router:

- at "edge" of its own AS
- has link to router in another AS

Interconnected ASes



- forwarding table configured by both intra- and inter-AS routing algorithm
  - intra-AS sets entries for internal dests
  - inter-AS & intra-AS sets entries for external dests

Inter-AS tasks

- suppose router 1d in AS1 receives datagram destined outside of AS1:
  - router should forward packet to gateway router, but which one?

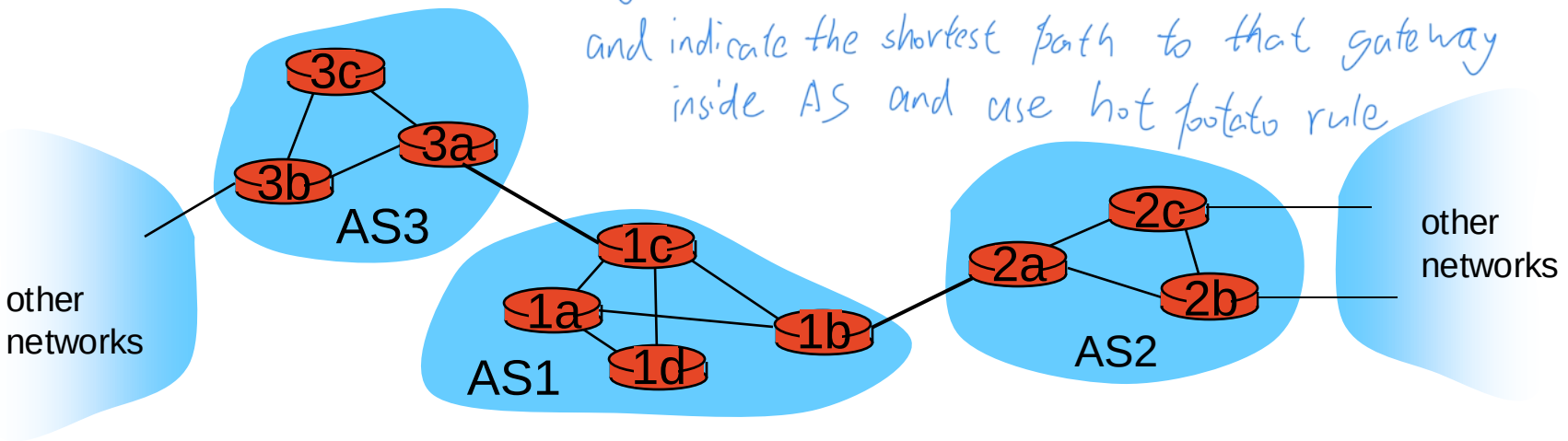
identify the gateway

AS1 must:

1. learn which dests are reachable through AS2, which through AS3
2. propagate this reachability info to all routers in AS1

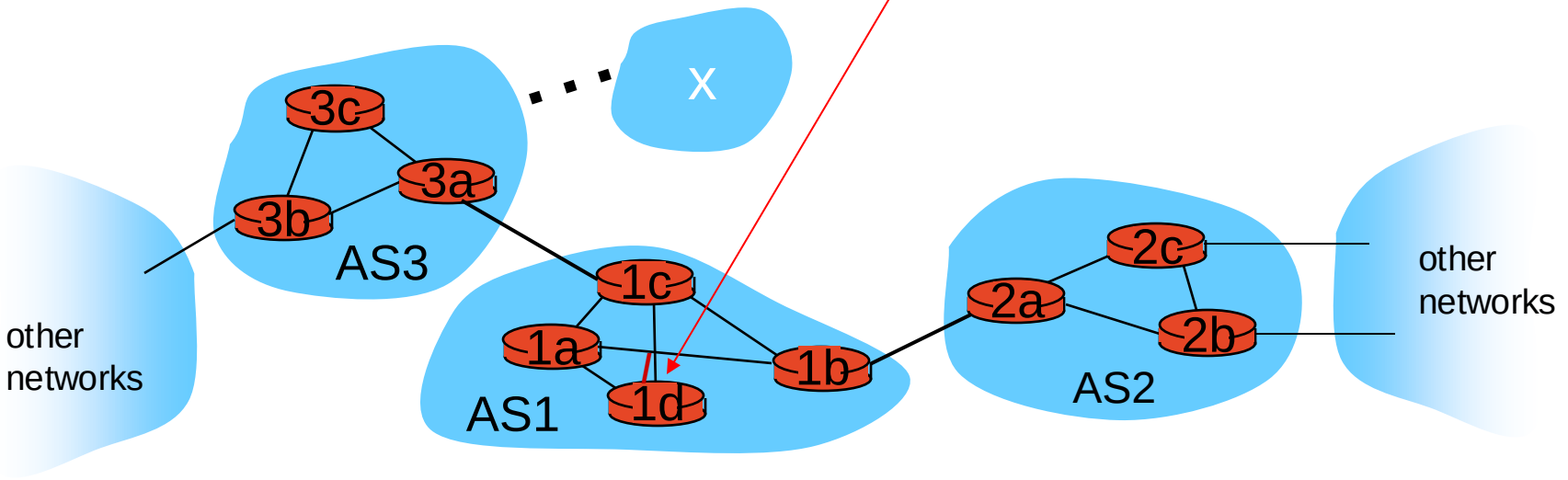
job of inter-AS routing!

and indicate the shortest path to that gateway inside AS and use hot potato rule



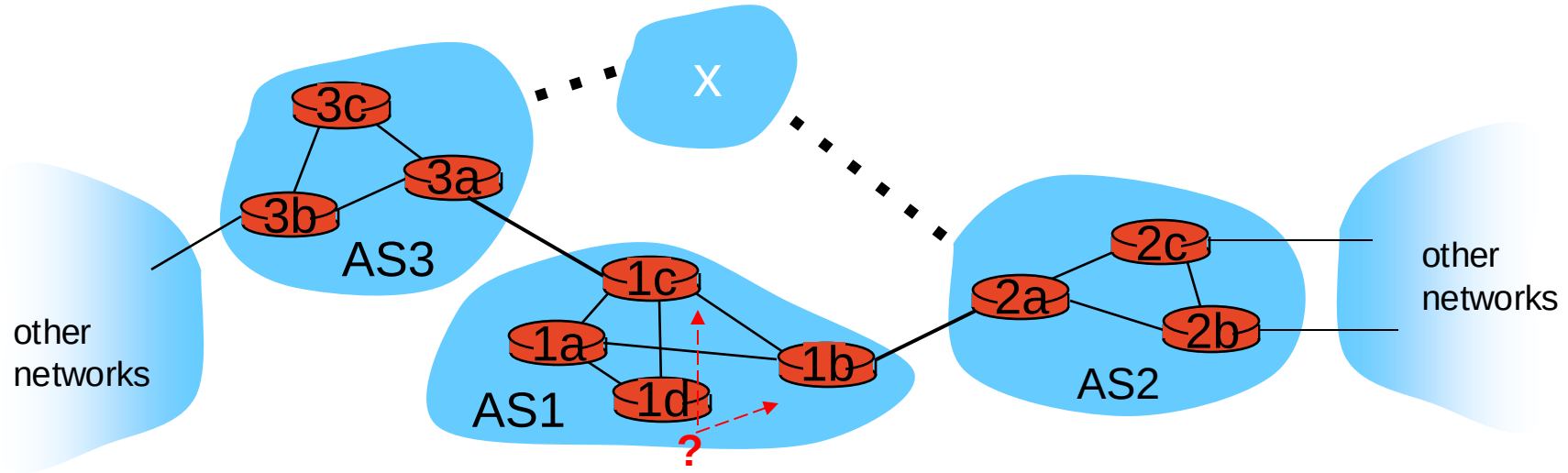
Example: setting forwarding table in router 1d

- suppose AS1 learns (via inter-AS protocol) that subnet *x* reachable via AS3 (gateway 1c), but not via AS2
  - inter-AS protocol propagates reachability info to all internal routers
- router 1d determines from intra-AS routing info that its interface *I* is on the least cost path to 1c
  - installs forwarding table entry *(x,I)*



Example: choosing among multiple ASes

- now suppose AS1 learns from inter-AS protocol that subnet *x* is reachable from AS3 *and* from AS2.
- to configure forwarding table, router 1d must determine which gateway it should forward packets towards for dest x
  - this is also job of inter-AS routing protocol!



Example: choosing among multiple ASes

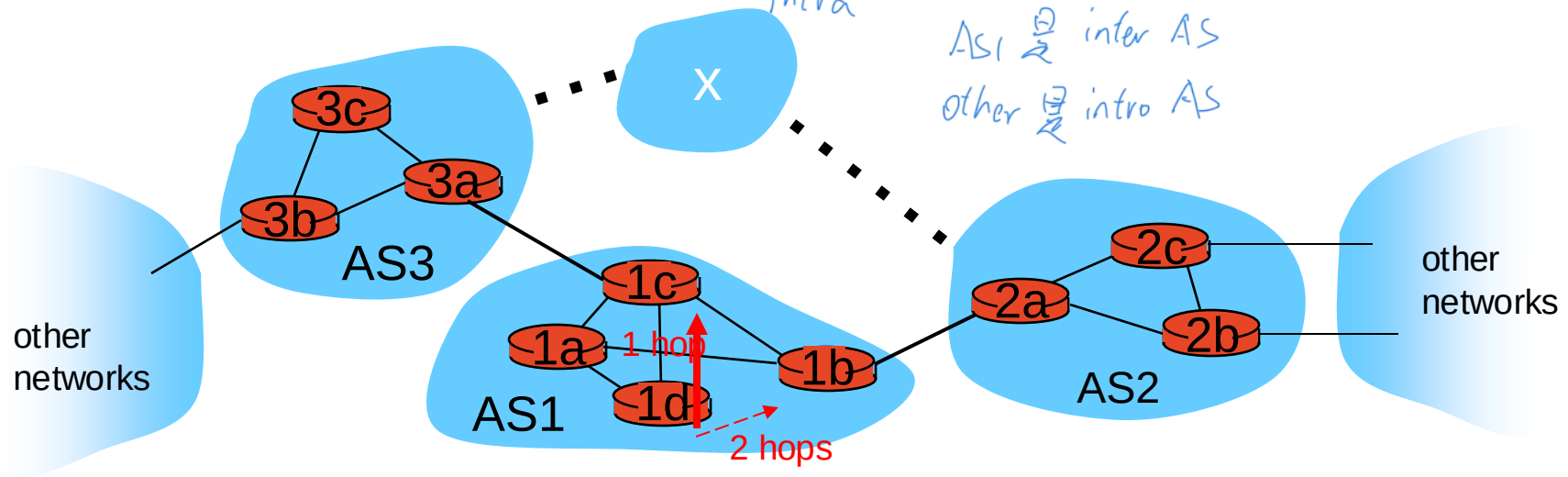
- now suppose AS1 learns from inter-AS protocol that subnet *x* is reachable from AS3 *and* from AS2.
- to configure forwarding table, router 1d must determine towards which gateway it should forward packets for dest x
  - this is also job of inter-AS routing protocol!
- *hot potato routing: send* packet towards closest of two gateway routers.

只判断 AS 的 shortest path

intra

AS1 是 inter AS

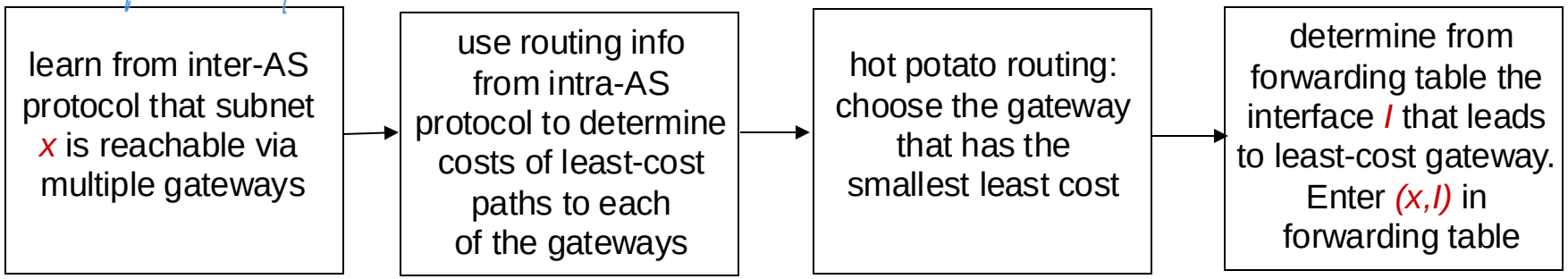
other 是 intra AS



Example: choosing among multiple ASes

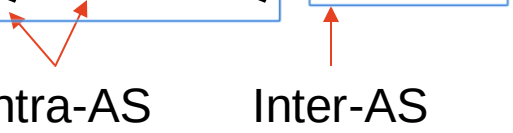
- now suppose AS1 learns from inter-AS protocol that subnet *x* is reachable from AS3 *and* from AS2.
- to configure forwarding table, router 1d must determine towards which gateway it should forward packets for dest x
  - this is also job of inter-AS routing protocol!
- *hot potato routing: send* packet towards closest of two gateway routers.

Example Steps:



Routing in the Internet

– RIP, OSPF, BGP

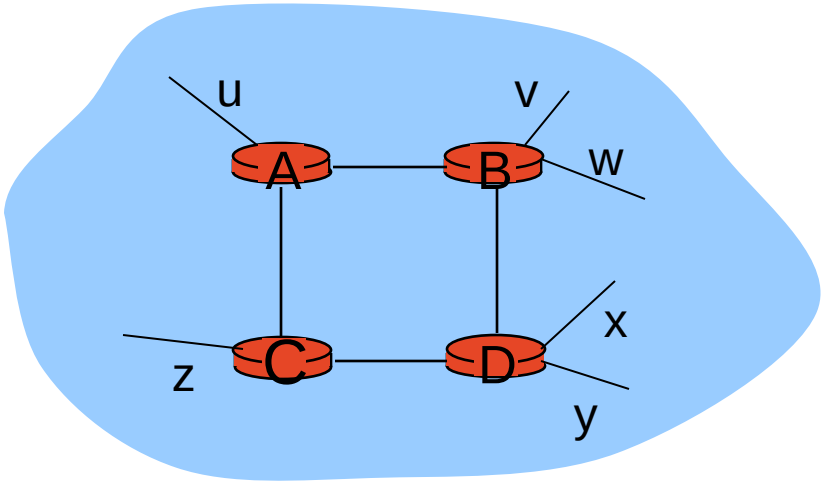


Intra-AS Routing

- also known as *interior gateway protocols (IGP)*
- most common intra-AS routing protocols:
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (Cisco proprietary)

RIP (Routing Information Protocol)

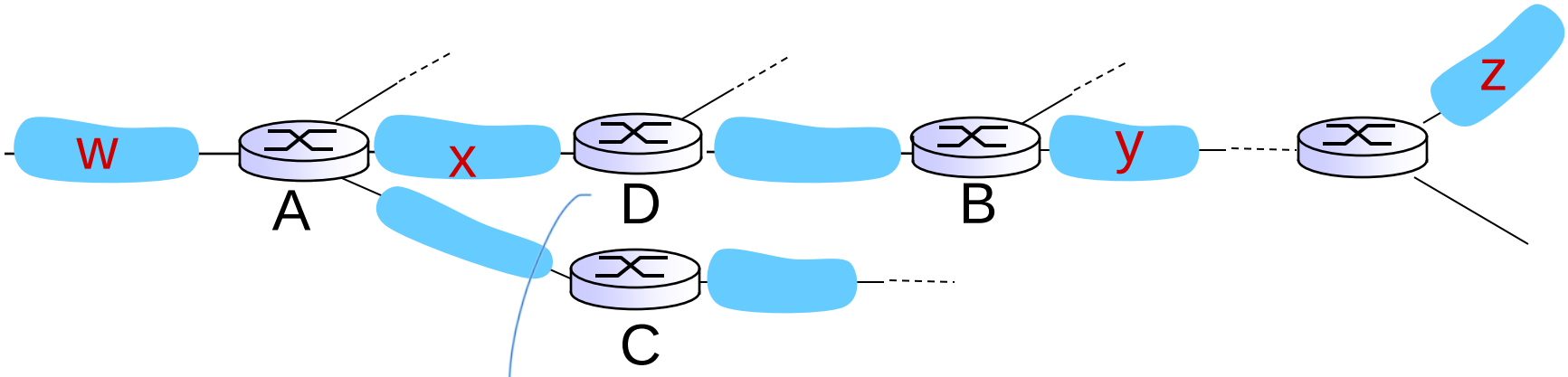
- included in BSD-UNIX distribution in 1982
- distance vector algorithm
  - distance metric: # hops (max = 15 hops), each link has cost 1
  - DVs exchanged with neighbors every 30 sec in response message (aka advertisement)
  - each advertisement: list of up to 25 destination *subnets (in IP addressing sense)*



from router A to destination *subnets:*

subnet	hops
u	1
v	2
w	2
x	3
y	3
z	2

RIP: example

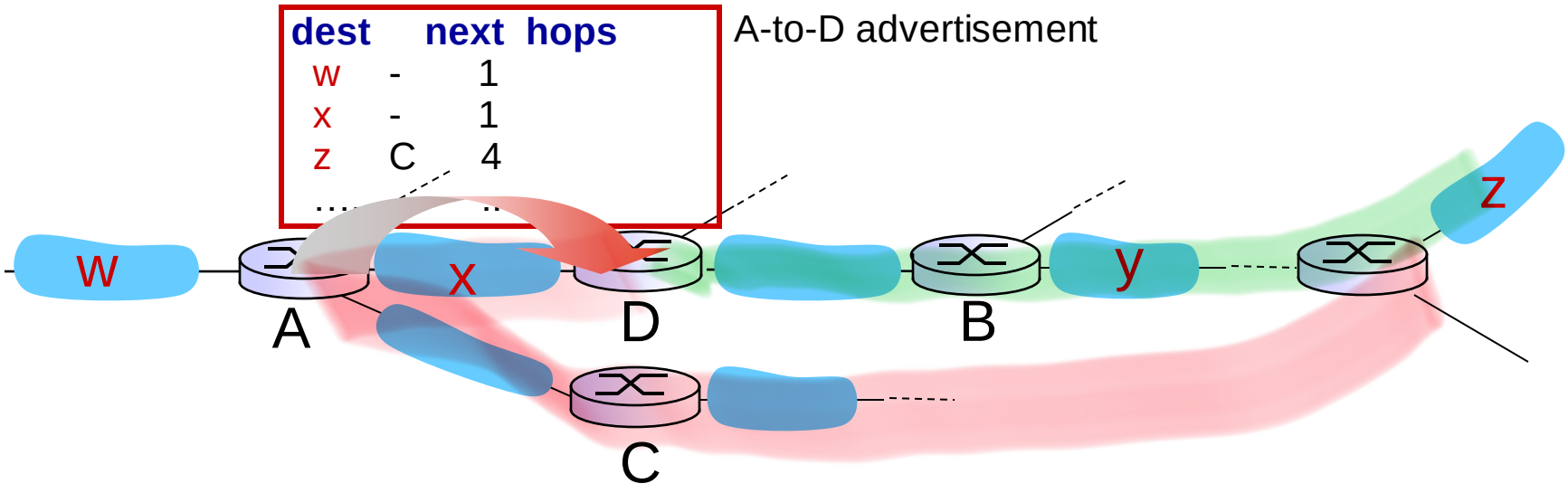


routing table in router D

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	B	7
x	--	1
....	....	....

directly connected

RIP: example



routing table in router D

destination subnet	next router	# hops to dest
w	A	2
y	B	2
z	B	7
x	--	1
....	....	....

A

5

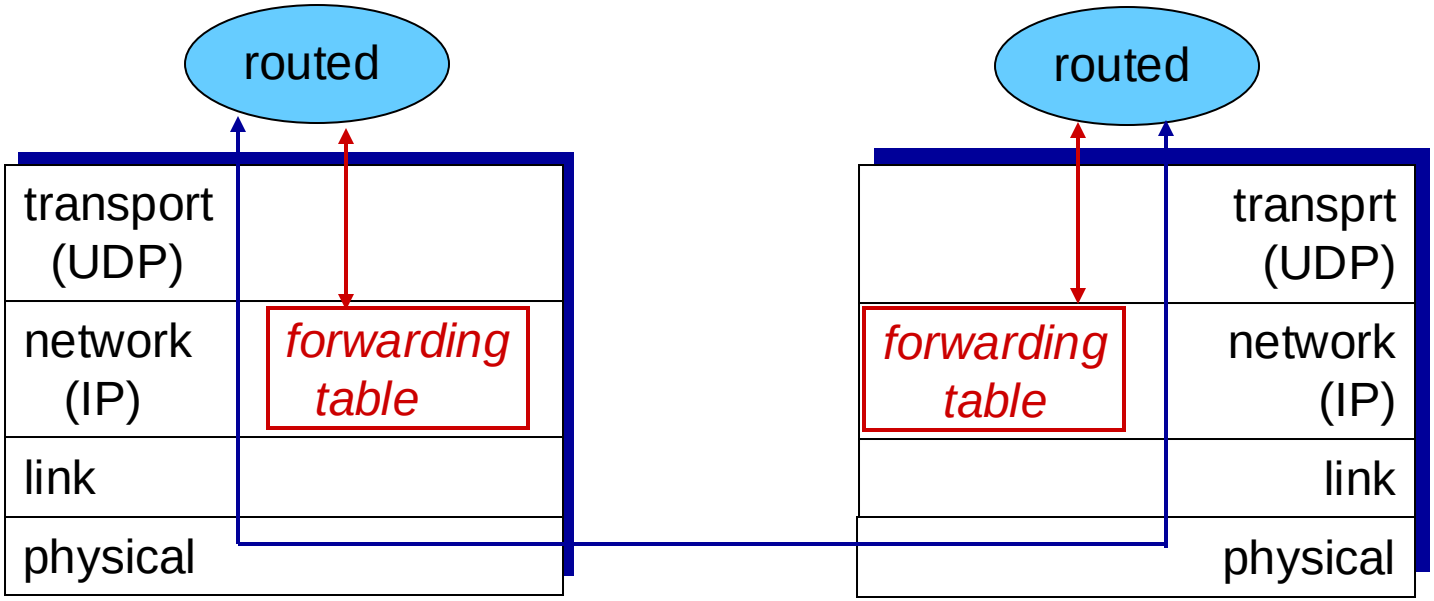
RIP: link failure, recovery

if no advertisement heard after 180 sec --> neighbor/link declared dead

- routes via neighbor invalidated
- new advertisements sent to neighbors
- neighbors in turn send out new advertisements (if tables changed)
- link failure info quickly (how?) propagates to entire net
- *poison reverse* used to prevent ping-pong loops (infinite distance = 16 hops)

RIP table processing

- RIP routing tables managed by *application-level* process called route-d (daemon, i.e., computer program that runs as a background process)
- advertisements sent in UDP packets, periodically repeated



OSPF (Open Shortest Path First)

- open: publicly available
- uses link state algorithm
  - LS packet dissemination
  - topology map at each node
  - route computation using Dijkstra's algorithm
- advertisements flooded to *entire* AS
  - carried in OSPF messages directly over IP (rather than TCP or UDP

OSPF advanced features (not in RIP)

- *security:* all OSPF messages authenticated (to prevent malicious intrusion)
- multiple same-cost paths allowed (only one path in RIP)
- for each link, multiple cost metrics for different ToS (Type of Service) (e.g., satellite link cost set "low" for best effort ToS; high for real time ToS)
- integrated uni- and multicast support:
  - Multicast OSPF (MOSPF) uses same topology data base as OSPF
- hierarchical OSPF in large domains.

BGP
Border Gateway Protocol

BGP

- Internet is so big that it is impossible to have the same routing protocol running on all routers
- Each collection of networks and routers managed by one administrative authority is called an Autonomous System (AS)
  - Each AS is identified by a number
- We need a protocol for routing among Autonomous Systems

Motivation

- Intra-AS protocol: as efficient as possible.
- Inter-AS protocol: policy is important.
  - Shortest path: x-AS1-AS2-AS3-y
  - AS2: Why shall I carry their traffic?
  - Other policies:
    - Do not carry commercial traffic on the educational network.
    - Use AT&T instead of Verizon because it is cheaper.
    - Traffic starting or ending at Apple should not transit Google.

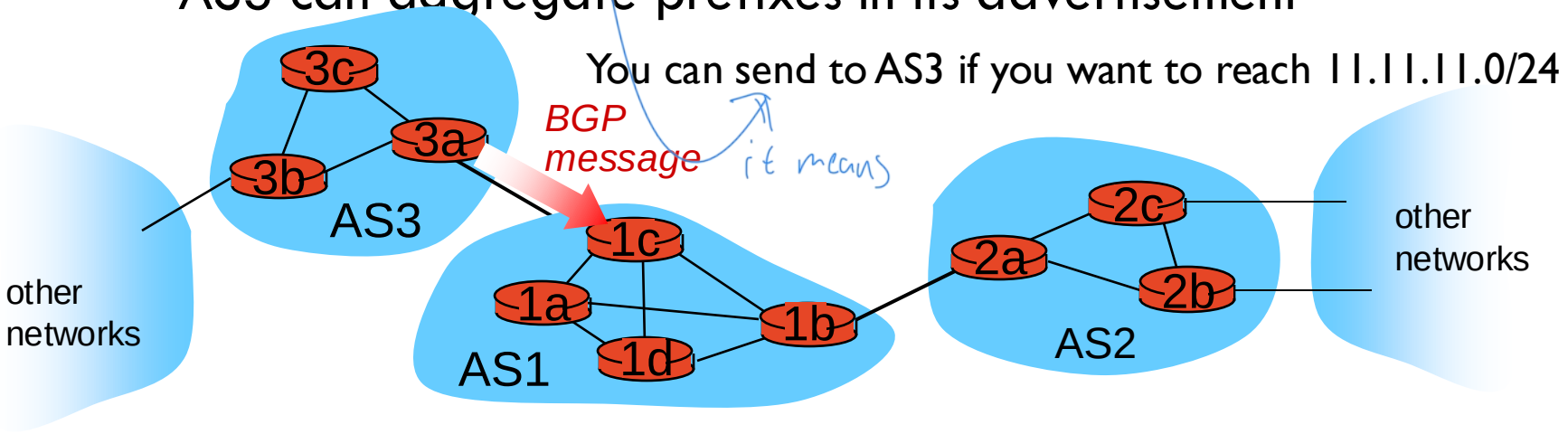
Internet inter-AS routing: BGP

- BGP (Border Gateway Protocol): *the* de facto inter-domain routing protocol
  - "glue that holds the Internet together"
- BGP provides each AS a means to:
  - eBGP: obtain subnet reachability information from neighboring ASs.
  - iBGP: propagate reachability information to all AS-internal routers.
  - determine "good" routes to other networks based on reachability information and policy.
- allows subnet to advertise its existence to rest of Internet: *"I am here"*

BGP basics

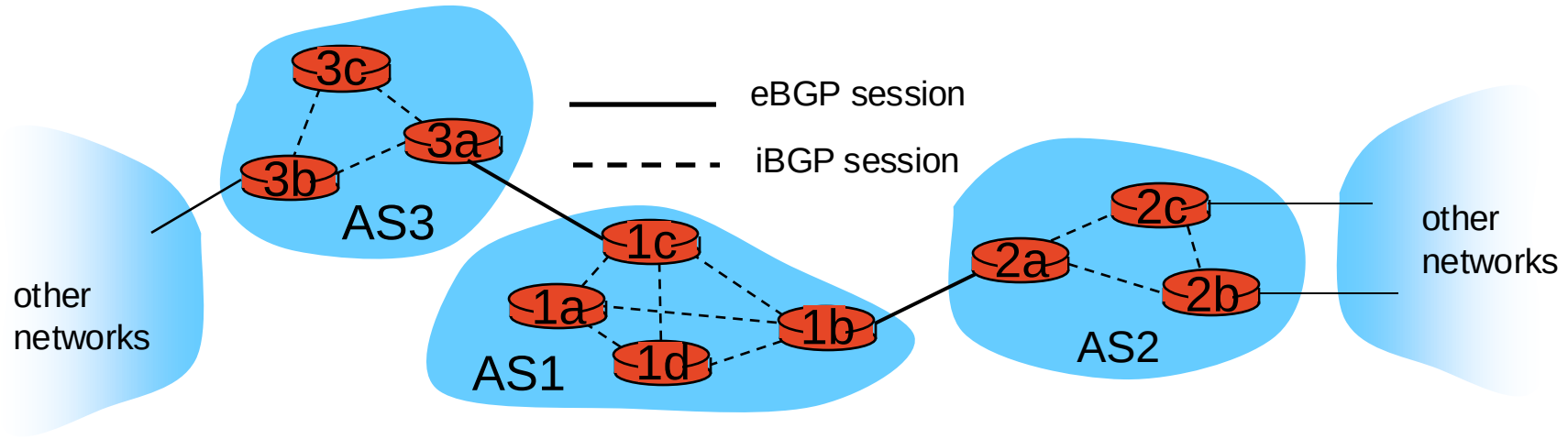
- **BGP session:** two BGP routers ("peers") exchange BGP messages:
  - advertising *paths* to different destination network prefixes (e.g. 11.11.11.0/24)

– when AS3 advertises a prefix (e.g. 11.11.11.0/24) to AS1:
  – AS3 *promises* it will forward datagrams towards that prefix
  – AS3 can aggregate prefixes in its advertisement



BGP basics: distributing path information

- using eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
  - 1c can then use iBGP do distribute new prefix info to all routers in AS1
  - 1b can then re-advertise new reachability info to AS2 over 1b-to-2a eBGP session



Path attributes and BGP routes

- advertised prefix includes BGP attributes
  - Route: prefix + attributes (AS-PATH, NEXT-HOP)
- two important attributes:
  - AS-PATH: contains AS path through which prefix advertisement has passed: e.g., AS 67, AS 17
  - NEXT-HOP: indicates specific internal-AS router to next-hop AS.)
- gateway router receiving route advertisement uses import policy to accept/decline
  - e.g., never route through AS x
  - *policy-based* routing

BGP attributes

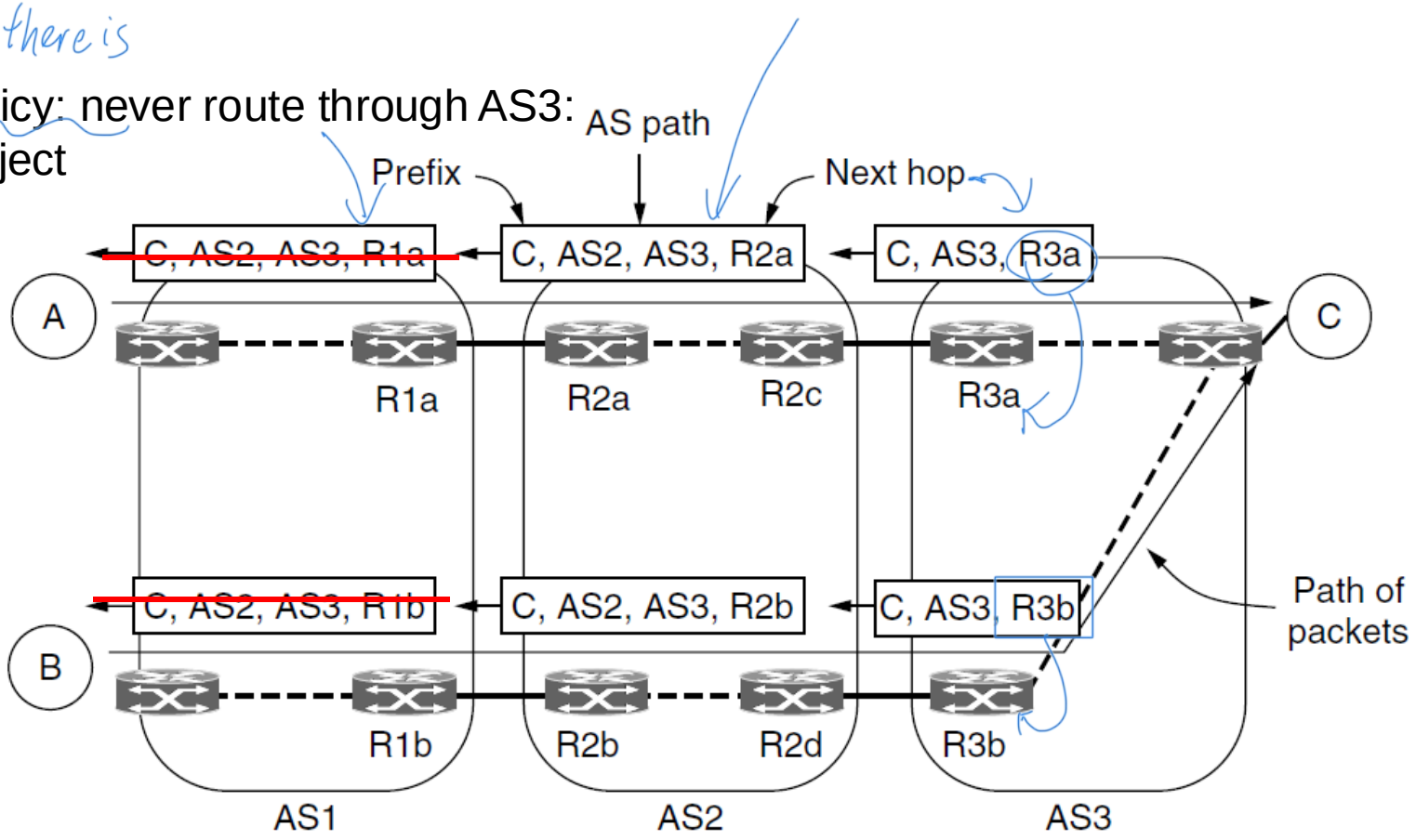
BGP routes

BGP Attributes

If there is

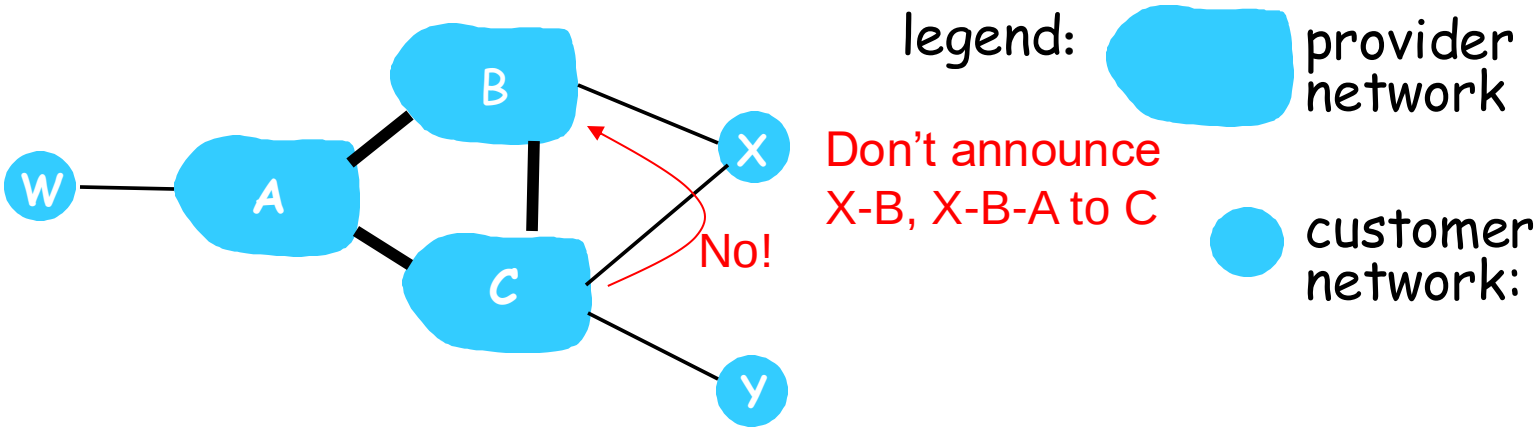
policy: never route through AS3:

then Reject



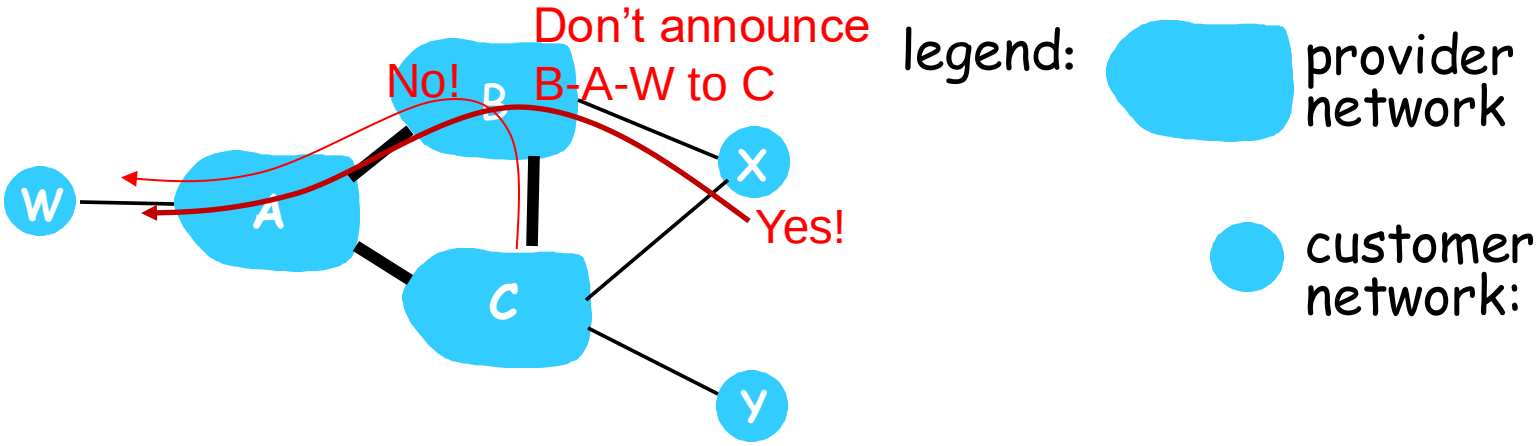
Source: Computer network, 5th edition, Tanenbaum

BGP routing policy



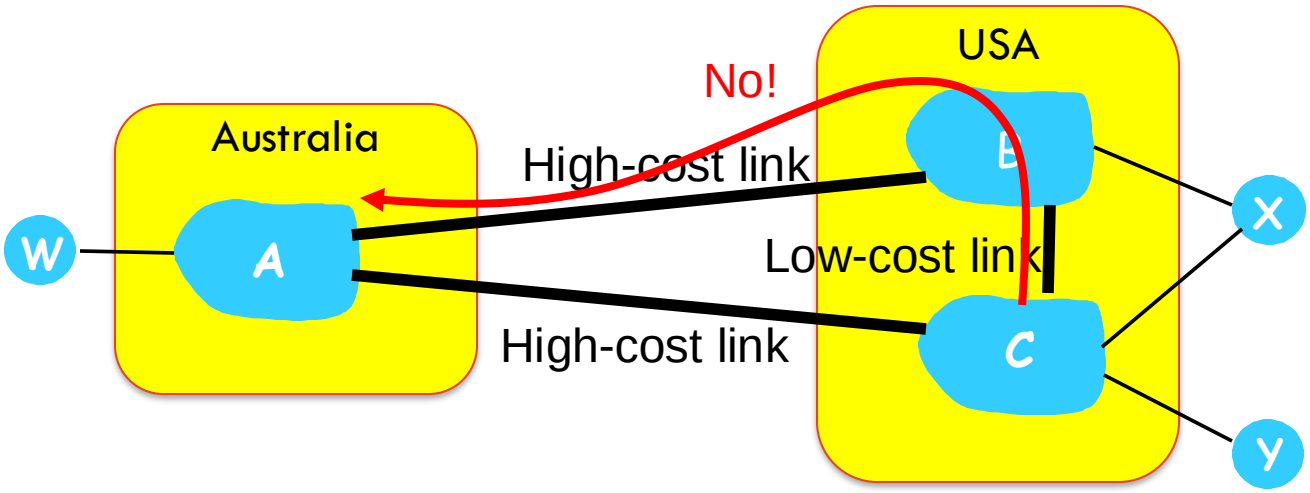
- A,B,C are provider networks
- X,W,Y are customer (of provider networks)
- X is dual-homed: attached to two networks Case 1
  - X does not want to route C-X-B
  - .. so X will not advertise to "X-B route" to C

BGP routing policy (2)



- A advertises path AW to B
- B advertises path BAW to X
- Should B advertise path BAW to C? Case 2
  - No way! B gets no "revenue" for routing CBAW since neither W nor C are B's customers
  - B wants to force C to route to w via A
  - B wants to route *only* to/from its customers!

BGP routing policy (2)



- Avoid free-rider, a real-world example

BGP route selection

- router may learn about more than 1 routes to destination AS, selects route based on:
  1. local preference / policy decision
  2. shortest AS-PATH
  3. closest NEXT-HOP router: hot potato routing
  4. additional criteria

How does entry get in forwarding table?

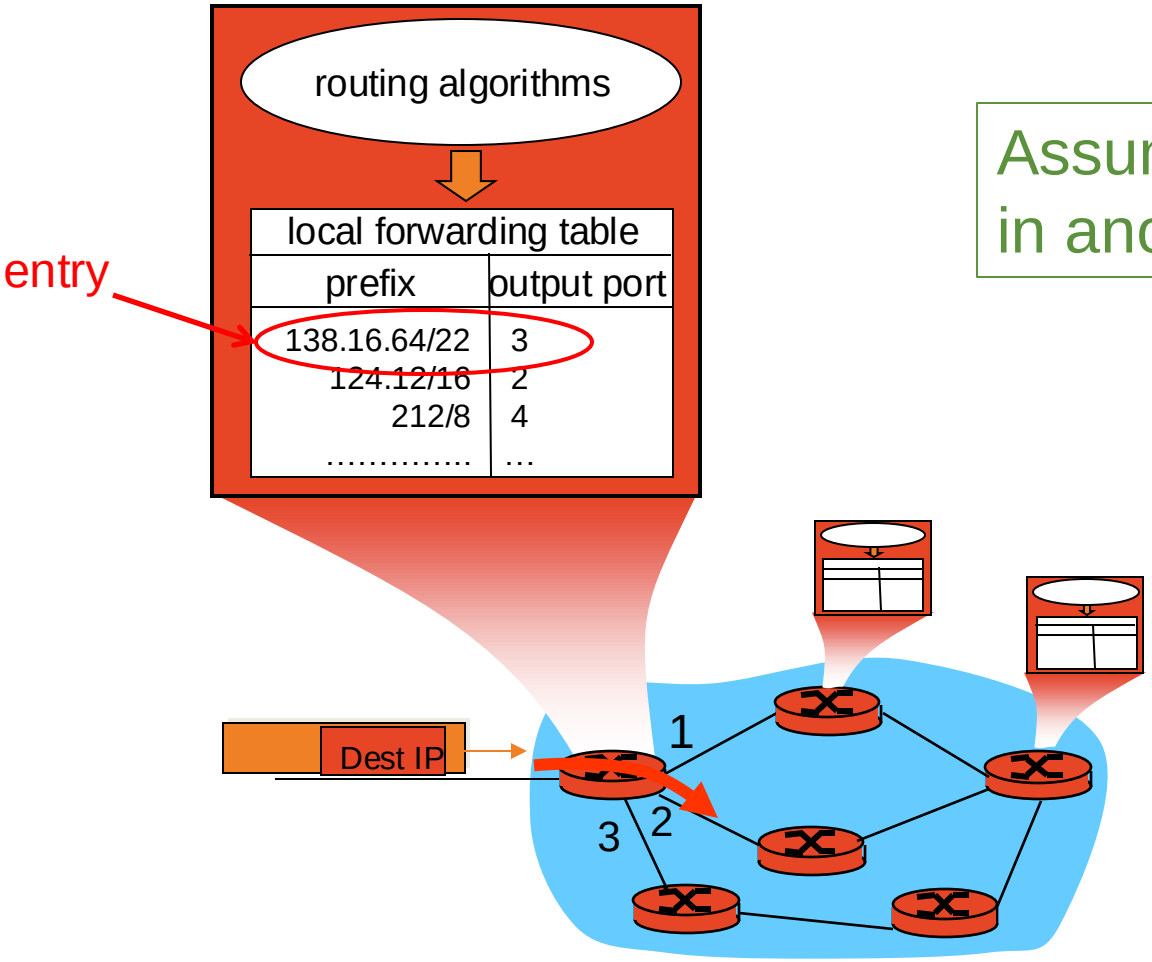
Putting it Altogether:

How Does an Entry Get Into a Router's Forwarding Table?

- Answer is complicated!

- Ties together hierarchical routing, with BGP and OSPF.

How does entry get in forwarding table?



Assume prefix is in another AS.

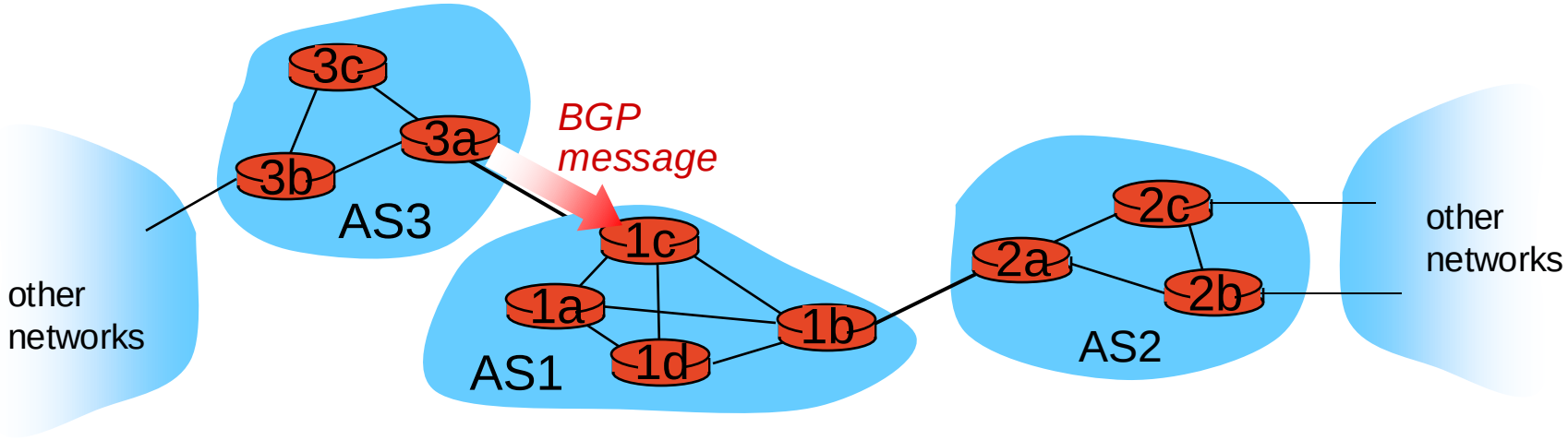
How does entry get in forwarding table?

Step of BGP

High-level overview

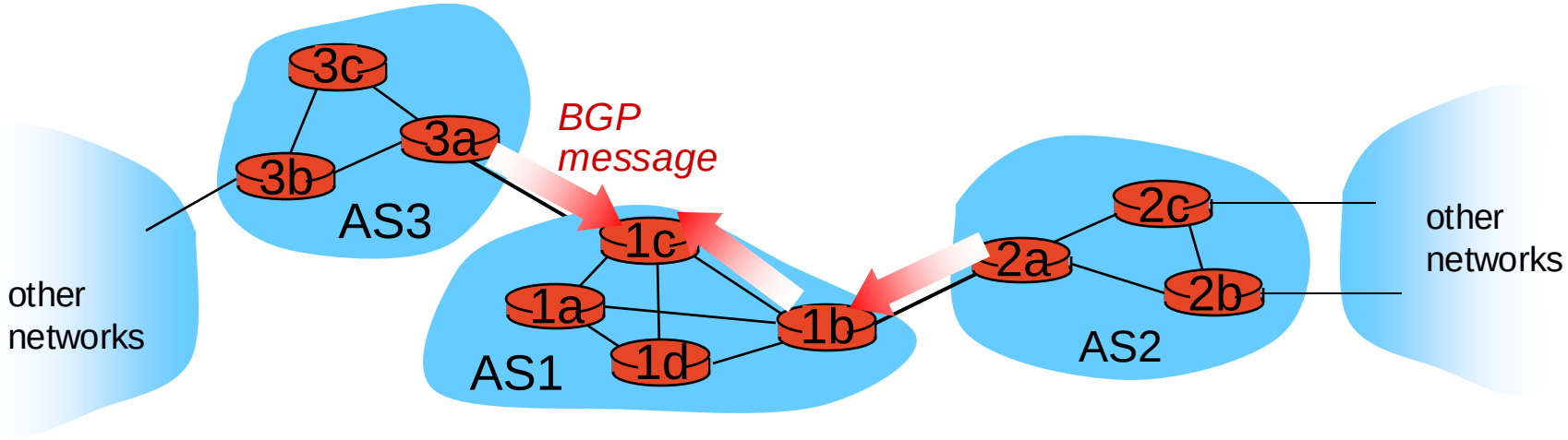
1. Router becomes aware of prefix
   - BGP
2. Router determines output port for prefix
   - BGP, OSPF 决定内部的最短 distance
3. Router enters prefix-port in forwarding table

Router becomes aware of prefix



- BGP message contains "routes"
- "route" is a prefix and attributes: AS-PATH, NEXT-HOP,…
- Example: route:
  - Prefix:138.16.64/22 ; AS-PATH: AS3 AS131 ; NEXT-HOP: 201.44.13.125

Router may receive multiple routes



- Router may receive multiple routes for same prefix
- Has to select one route

Select best BGP route to prefix

check From

Select for BGP

1. local preference / policy decision
2. shortest AS-PATH
3. closest NEXT-HOP router: hot potato routing
4. additional criteria

- Example:
  - ~~AS100 to 138.16.64/22~~
  - AS2 AS17 to 138.16.64/22
  - AS3 AS131 AS201 to 138.16.64/22

Breaks policy

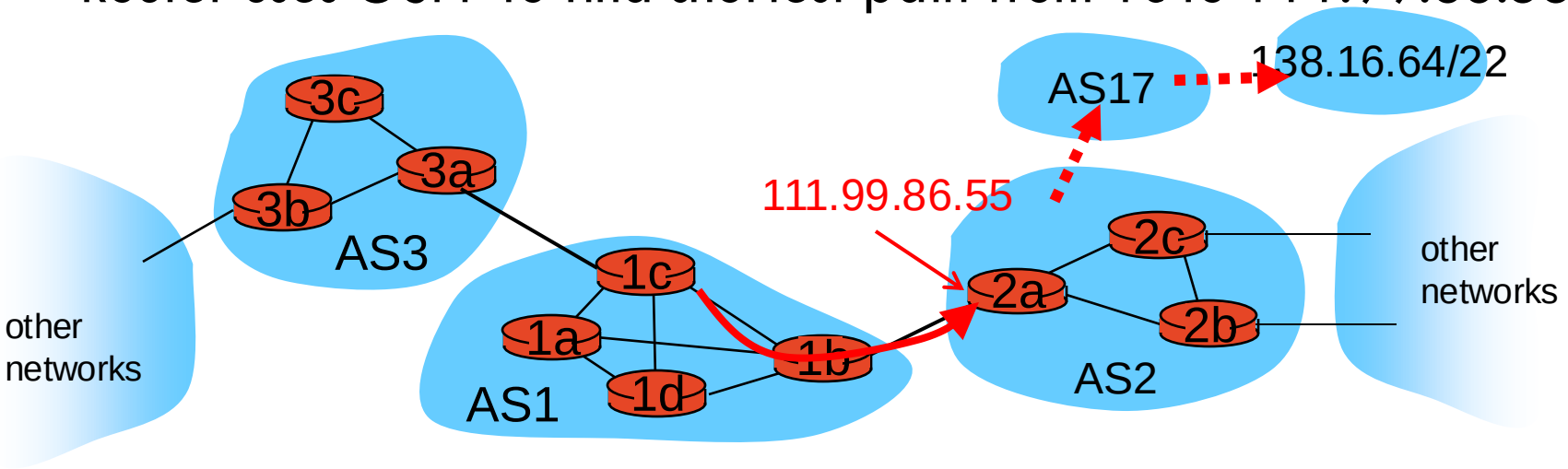
select

we choose this since shorter

- What if there is a tie? Hot potato routing. We'll come back to that!

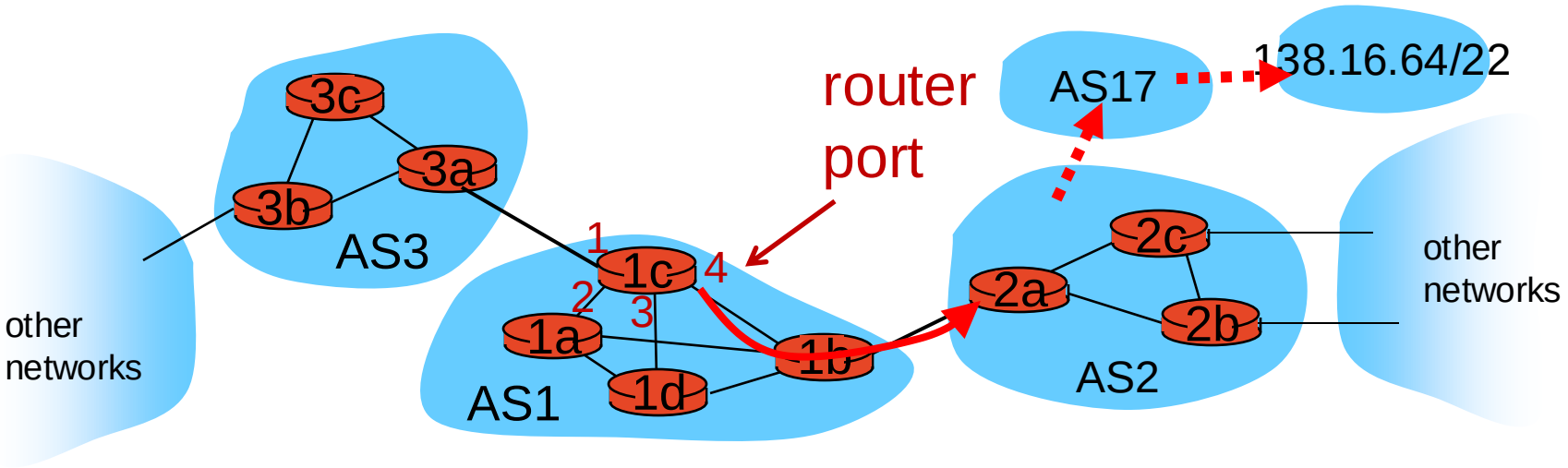
Find best intra-AS route to BGP route

- Use selected route's NEXT-HOP attribute
  - Route's NEXT-HOP attribute is the IP address of the router interface that begins the AS PATH.
- Example:
  - Prefix:138.16.64/22
  - AS-PATH: AS2 AS17
  - NEXT-HOP: 111.99.86.55
- Router uses OSPF to find shortest path from 1c to 111.99.86.55



Router identifies port for route

- Identifies port along the OSPF shortest path
- Adds prefix-port entry to its forwarding table:
  - (138.16.64/22 , port 4)



Select best BGP route to prefix

1. local preference / policy decision
2. shortest AS-PATH
3. closest NEXT-HOP router: hot potato routing
4. additional criteria

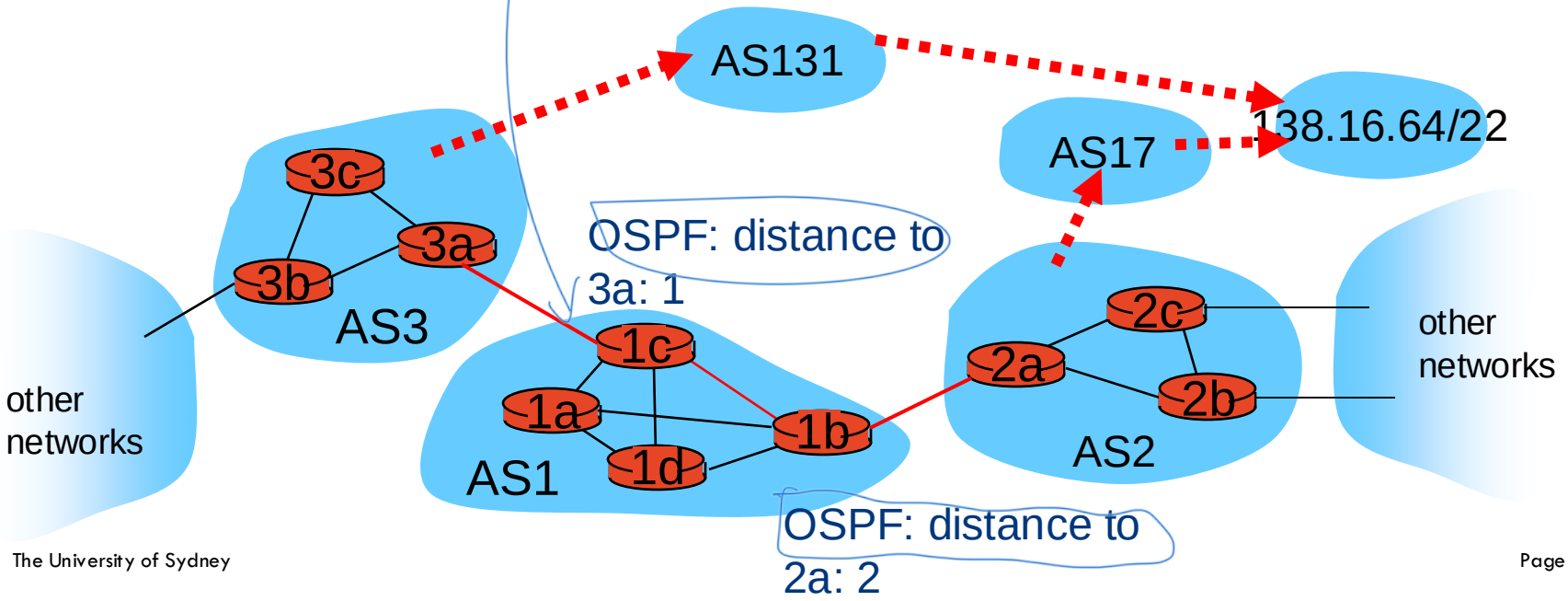
- Example:
  - ~~AS100 to 138.16.64/22~~
  - AS2 AS17 to 138.16.64/22
  - AS3 AS131 to 138.16.64/22

Breaks policy

- What if there is a tie? Hot potato routing.

Hot Potato Routing

- Suppose there two or more best inter-routes.
- Then choose route with closest NEXT-HOP
  - Use OSPF to determine which gateway is closest
  - Q: From 1c, chose AS3 AS131 or AS2 AS17?
  - A: route AS3 AS131 since it is closer
  - From 1b, route AS2 AS17 since it is closer



How does entry get in forwarding table?

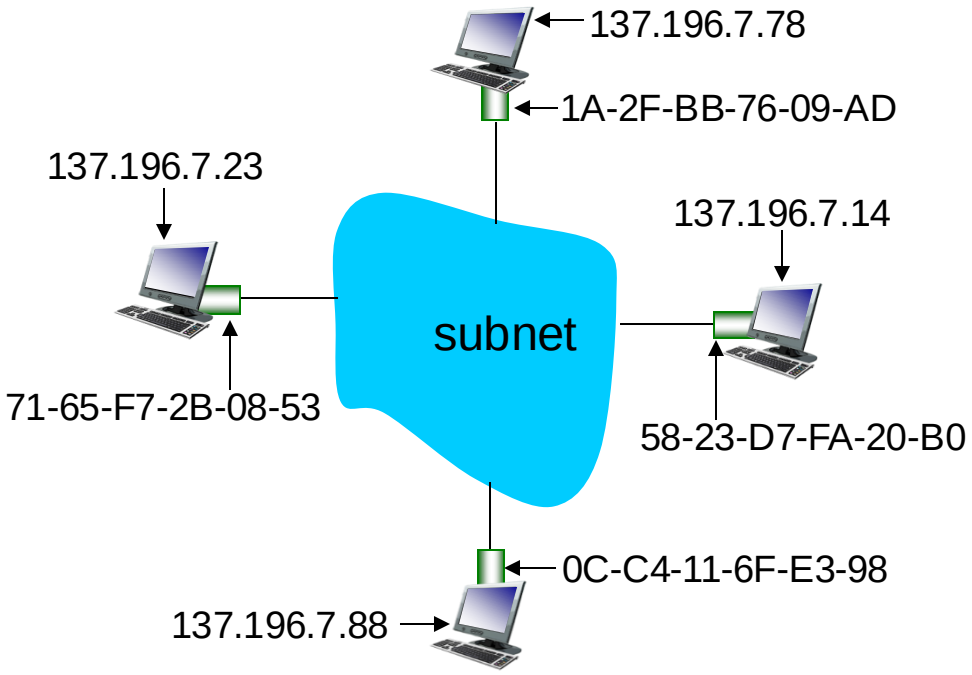
Summary

1. Router becomes aware of prefix
   - via BGP (eBGP/iBGP) route advertisements from other routers
2. Determine router output port for prefix
   - Use BGP route selection to find best inter-AS route
   - Use OSPF to find best intra-AS route leading to best inter-AS route
   - Ties: hot potato
   - Router identifies router port for that best route
3. Enter prefix-port entry in forwarding table

ARP
Address Resolution Protocol

ARP: address resolution protocol

Question: how to determine interface's MAC address, knowing its IP address?



ARP table: each IP node (host, router) on subnet (LAN) has table

- IP/MAC address mappings for some LAN nodes:

  < IP address; MAC address; TTL>

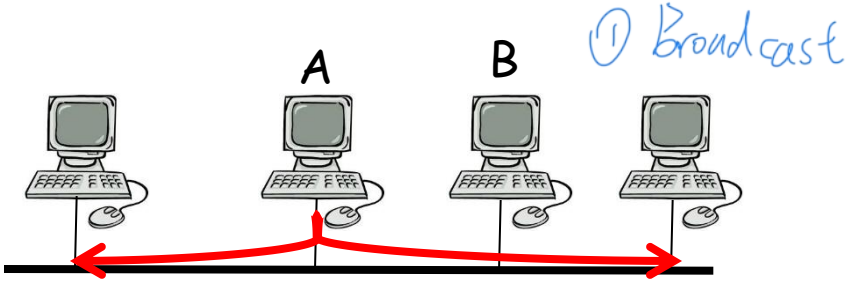
- TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

ARP protocol: LAN (same subnet)

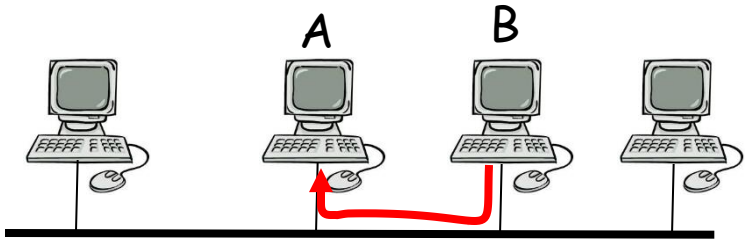
- A wants to send datagram to B
  - B's MAC address not in A's ARP table.
- A broadcasts ARP query packet, containing B's IP address
  - dest MAC address = FF-FF-FF-FF-FF-FF
  - all nodes on LAN receive ARP query
- B receives ARP packet, replies to A with its (B's) MAC address
  - frame sent to A's MAC address (unicast)
- A caches (saves) IP-to-MAC address pair in its ARP table until information becomes old (times out)
  - information that times out (goes away) unless refreshed
- ARP is "plug-and-play":
  - nodes create their ARP tables *without intervention from net administrator*

ARP protocol

Example



I'm 111.111.111.110, MAC address is AB-CD-EF-12-34-56, who is 111.111.111.111?

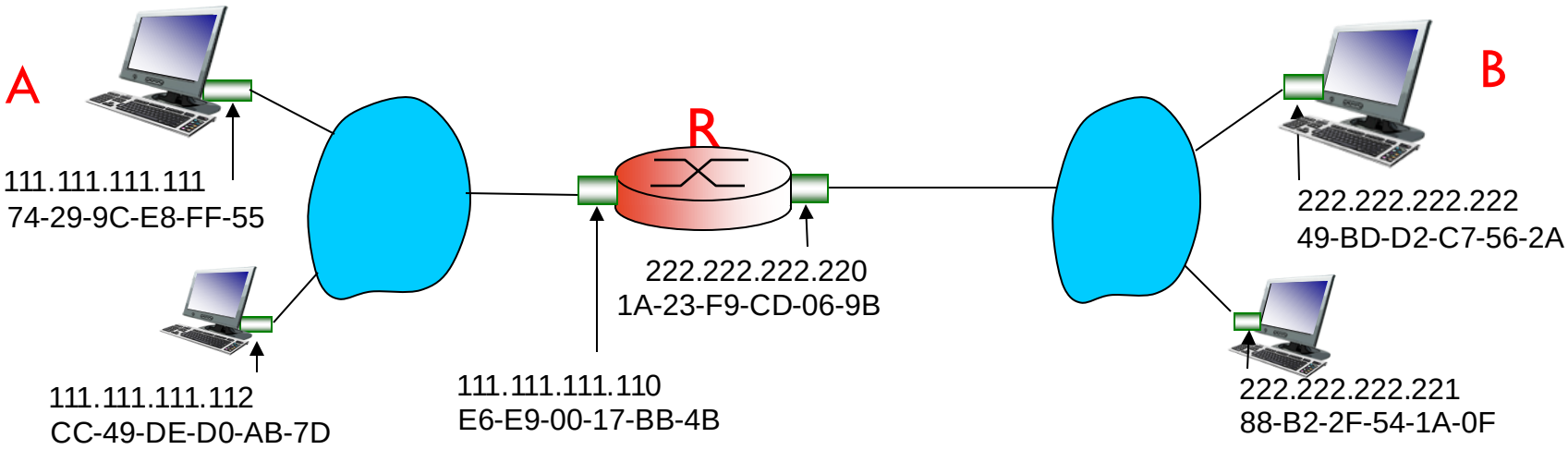


I'm 111.111.111.111, MAC address AB-CD-EF-12-34-57

Addressing: routing to another subnet

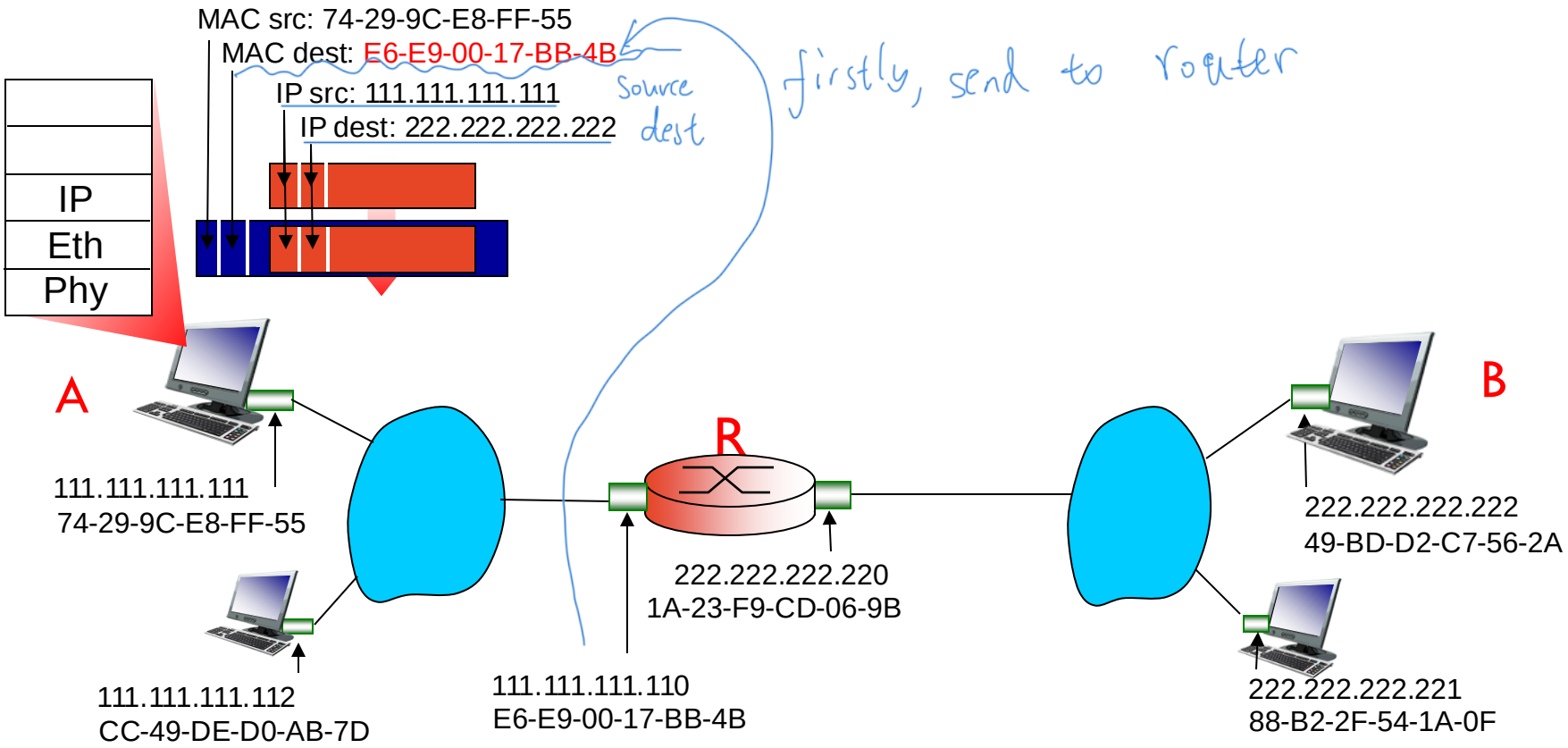
walkthrough: send datagram from A to B via R

- focus on addressing – at IP (datagram) and MAC layer (frame)
- assume A knows B's IP address — DNS (later)
- assume A knows IP address of first hop router, R (how?) — By DHCP
- assume A knows R's MAC address (how?) — By ARP



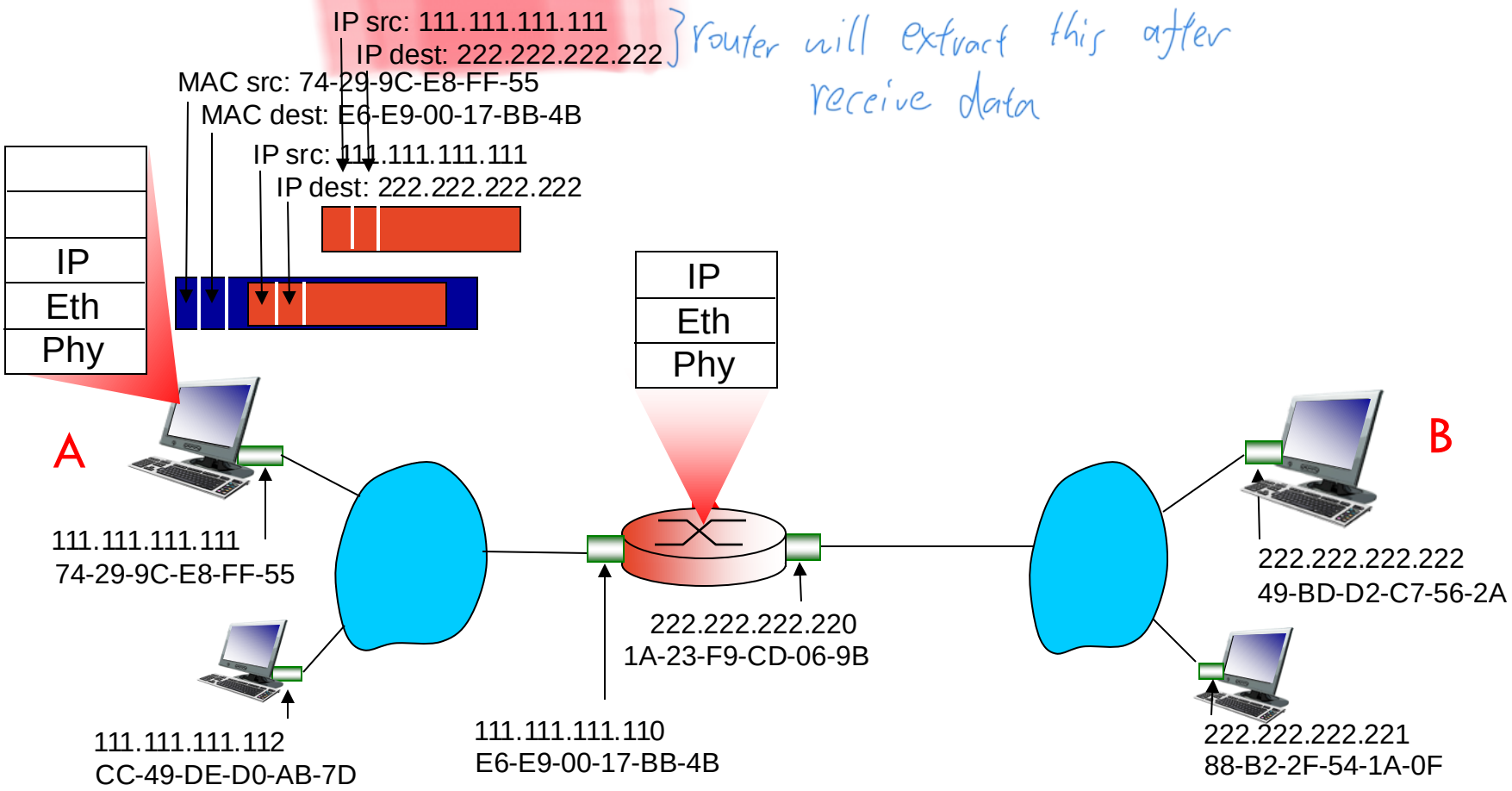
Addressing: routing to another subnet

- A creates IP datagram with IP source A, destination B
- A creates link-layer frame with R's MAC address as dest, frame contains A-to-B IP datagram



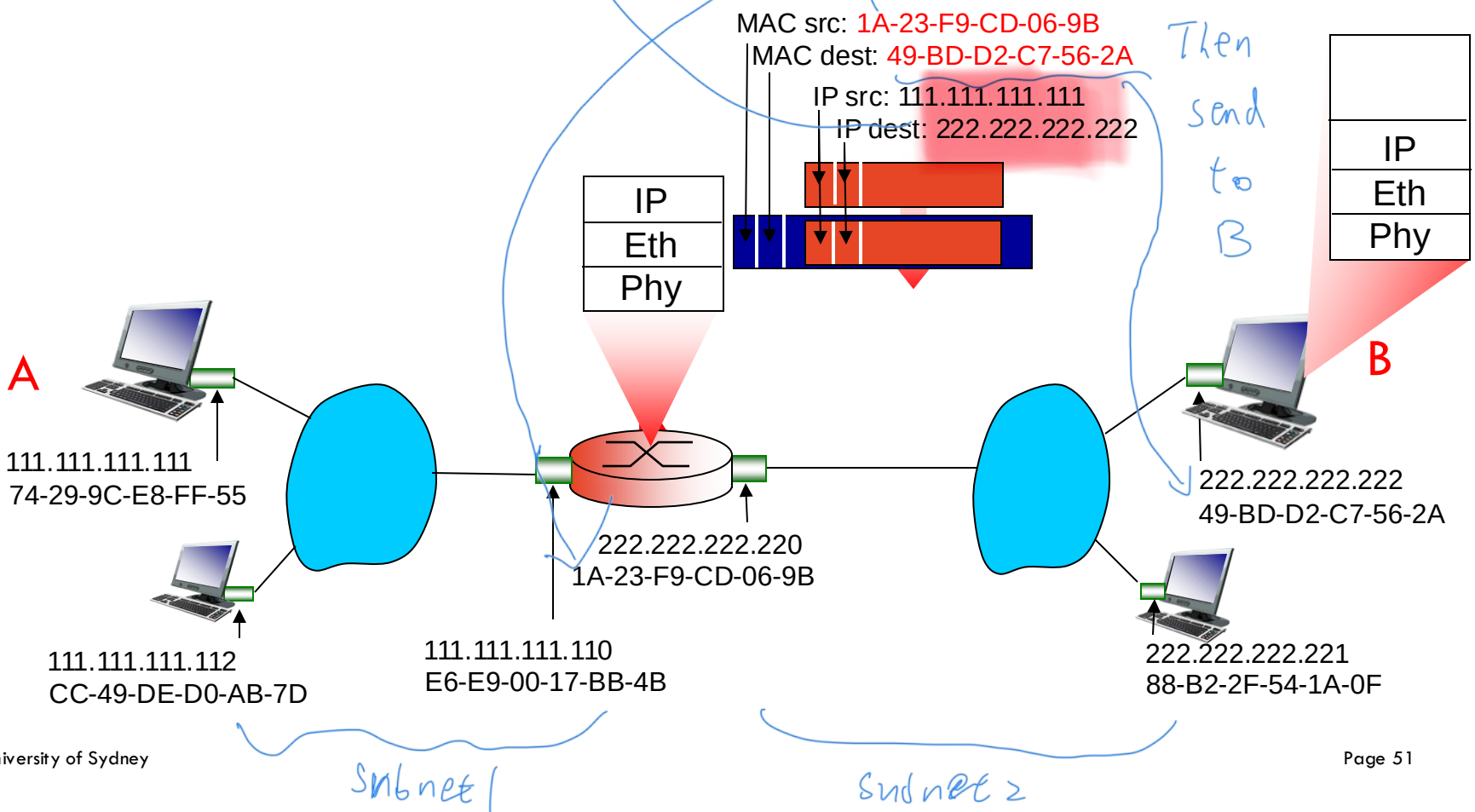
Addressing: routing to another subnet

- frame sent from A to R
- frame received at R, datagram removed, passed up to IP



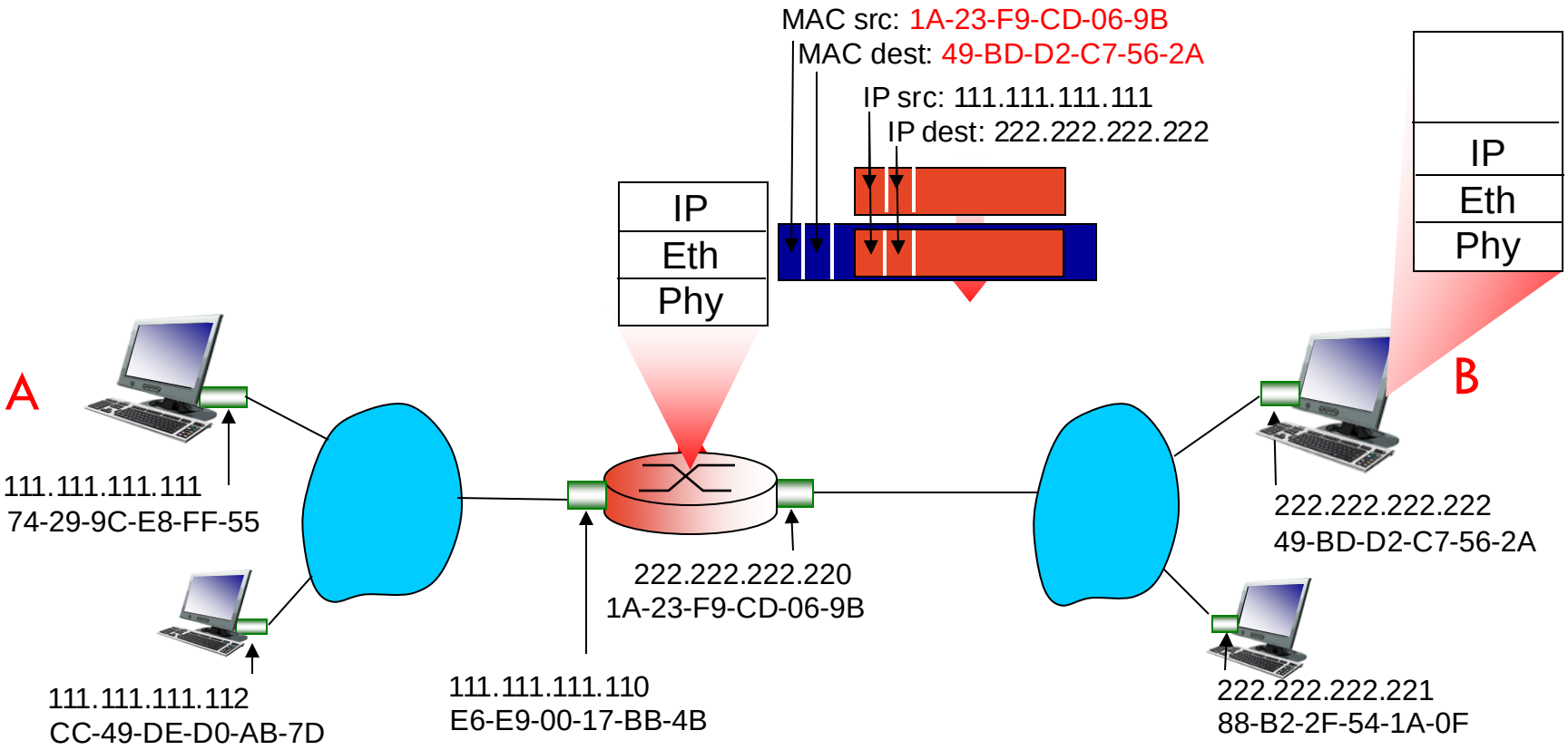
Addressing: routing to another subnet

- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as dest, frame contains A-to-B IP datagram



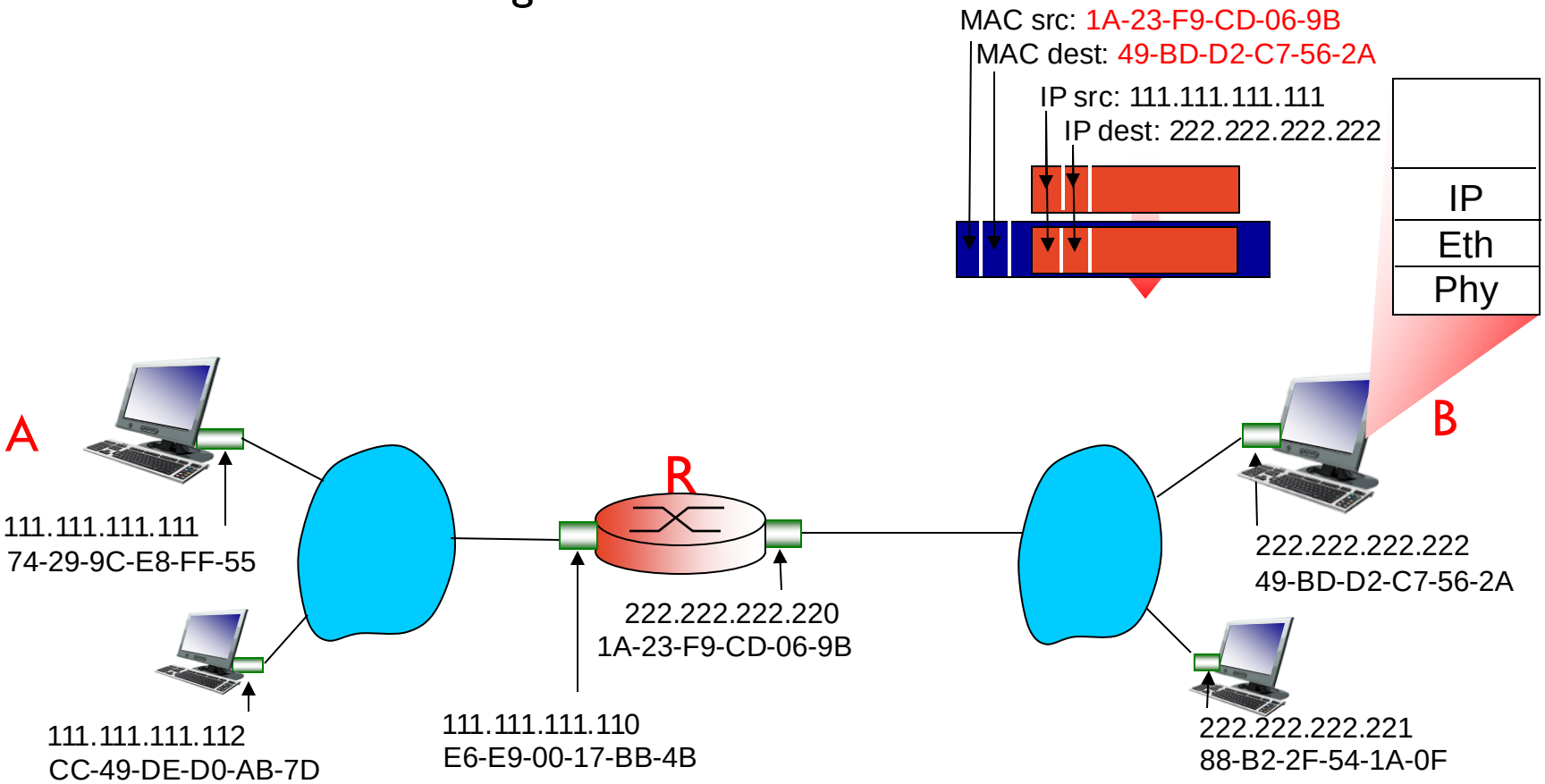
Addressing: routing to another subnet

- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as dest, frame contains A-to-B IP datagram



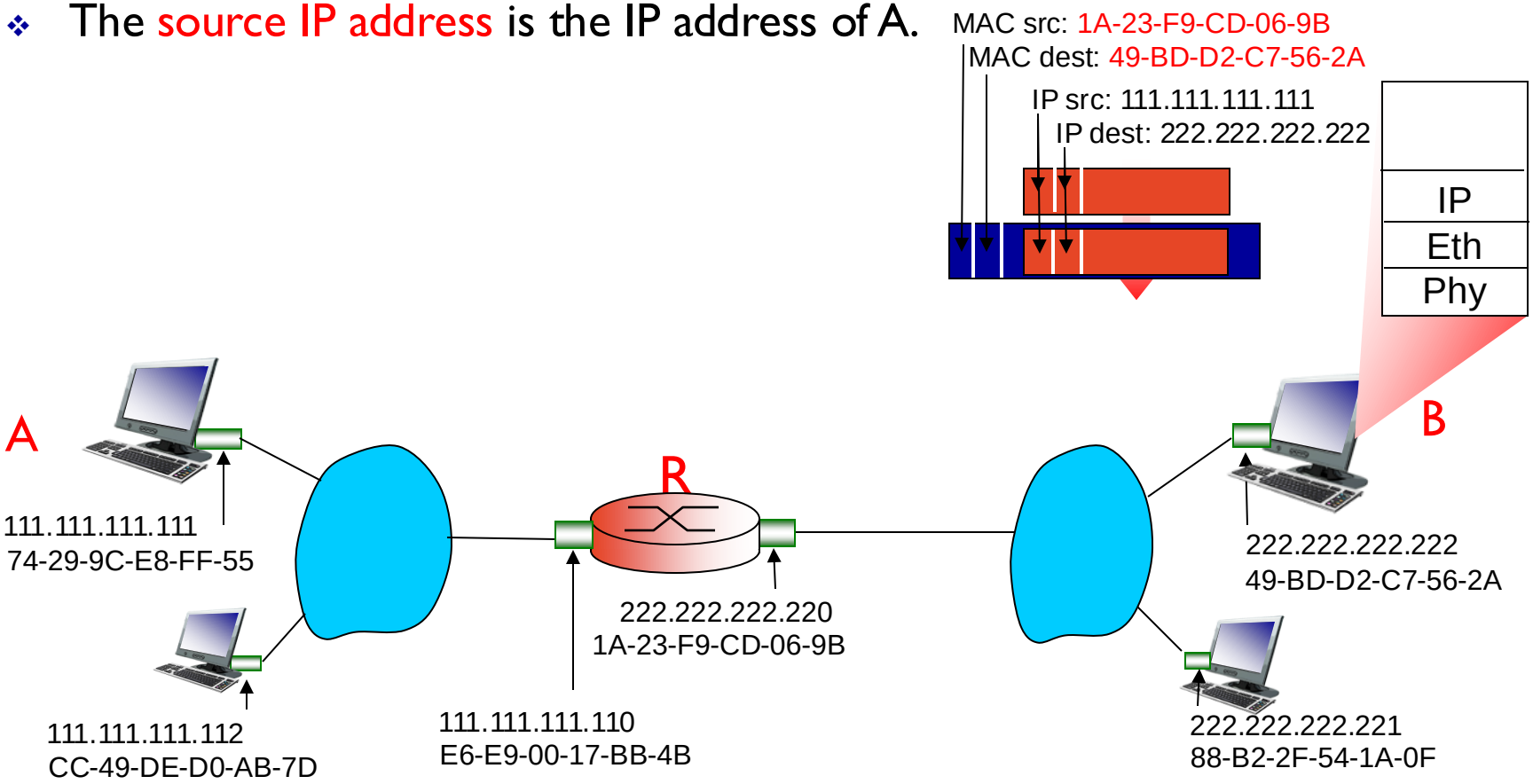
Addressing: routing to another subnet

- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B's MAC address as dest, frame contains A-to-B IP datagram



Addressing: routing to another subnet

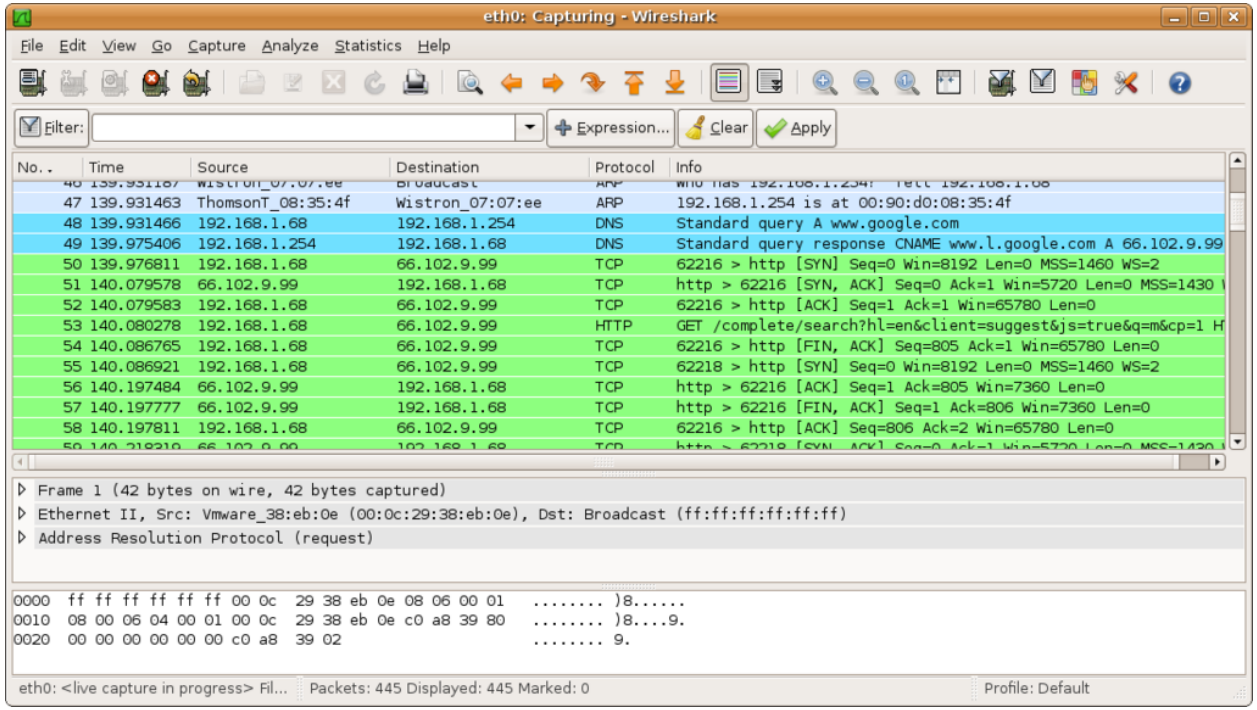
- When B receives the frame:
- The source MAC address is the MAC address of R.
- The source IP address is the IP address of A.



Wireshark

Wireshark

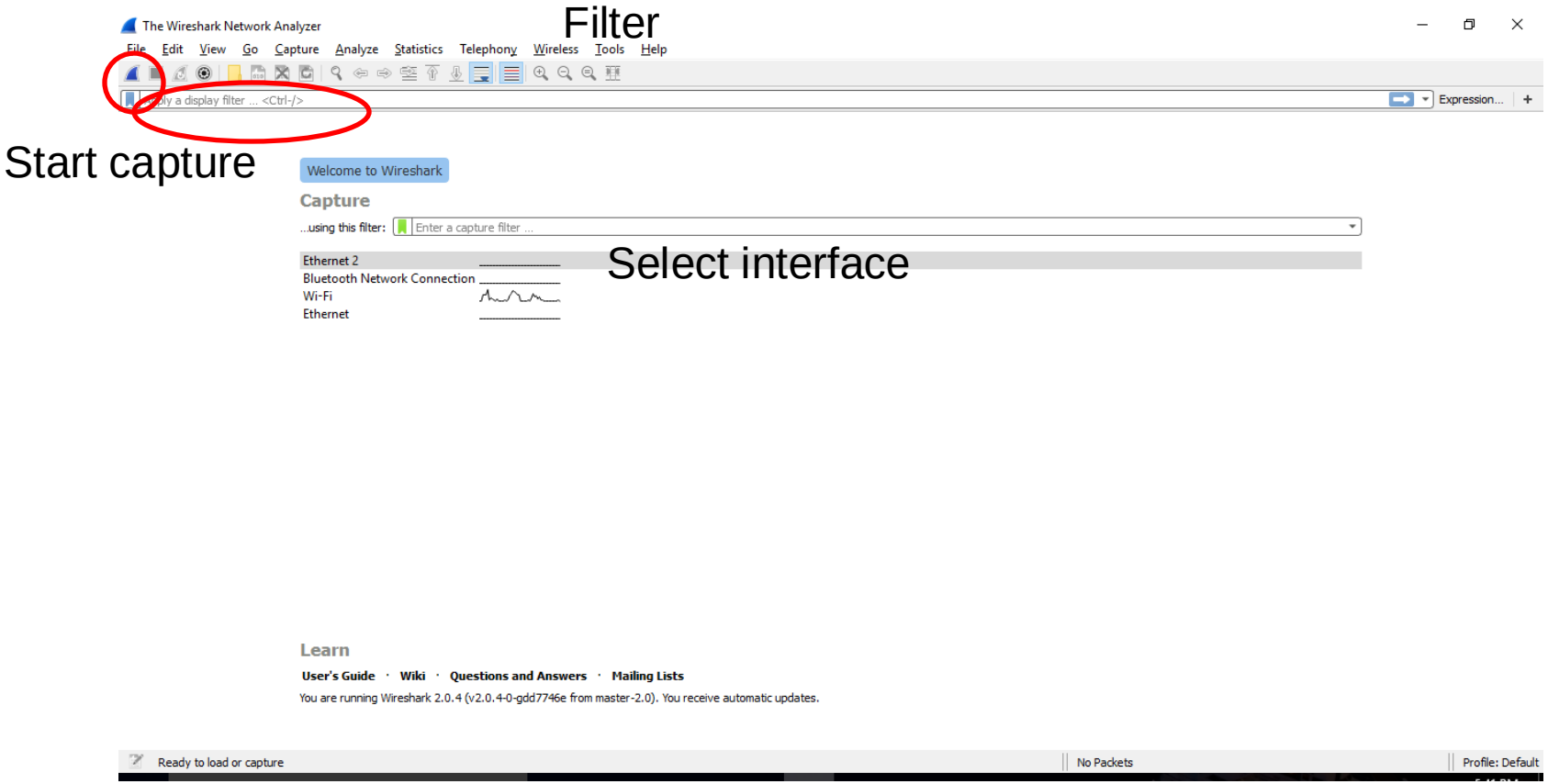
- Wireshark is a free and open source packet analyzer. It is used for network troubleshooting and analysis.



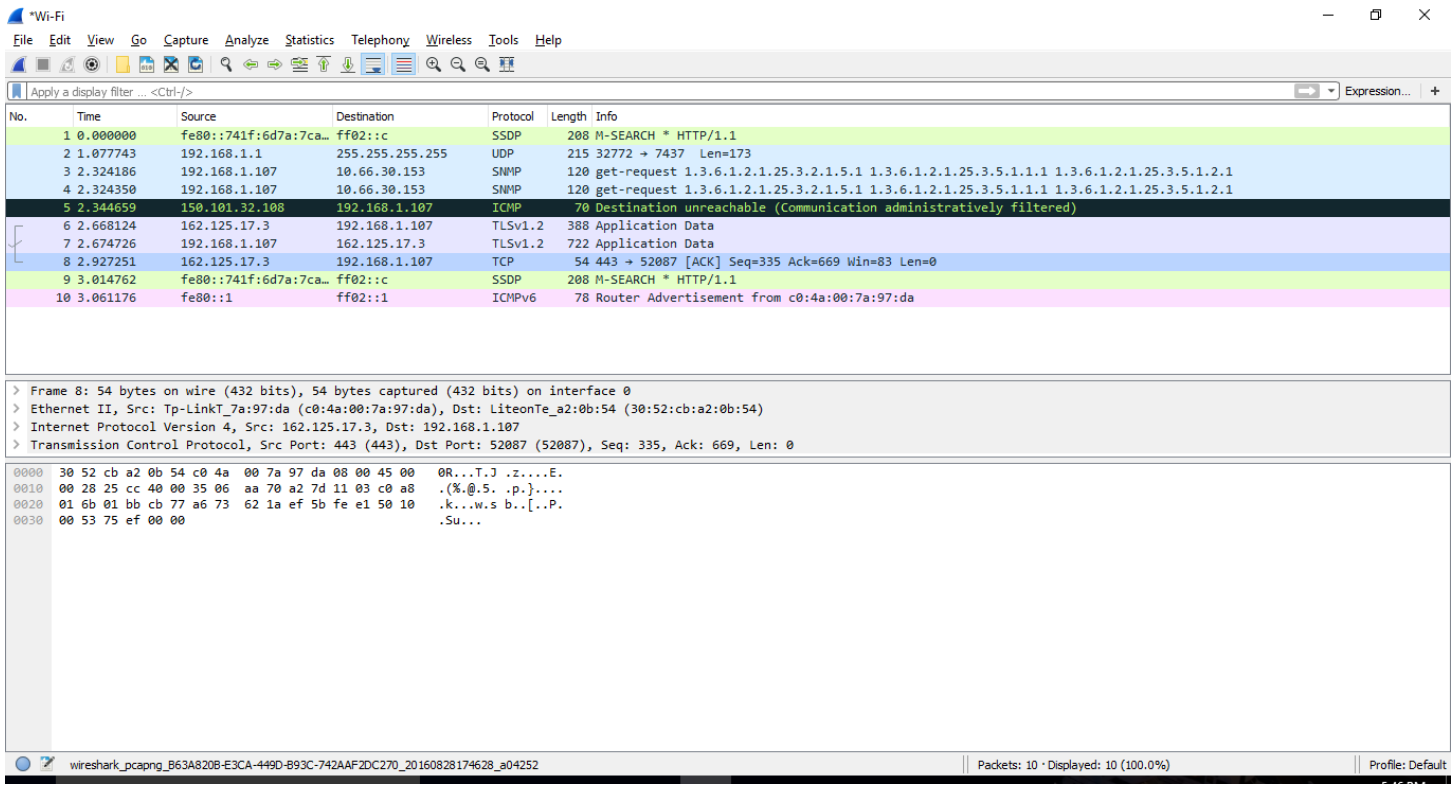
Wireshark

- Start Wireshark
- Linux: type “wireshark” at the terminal
- At your own computer
- Linux: sudo wireshark
- Windows: run as administrator

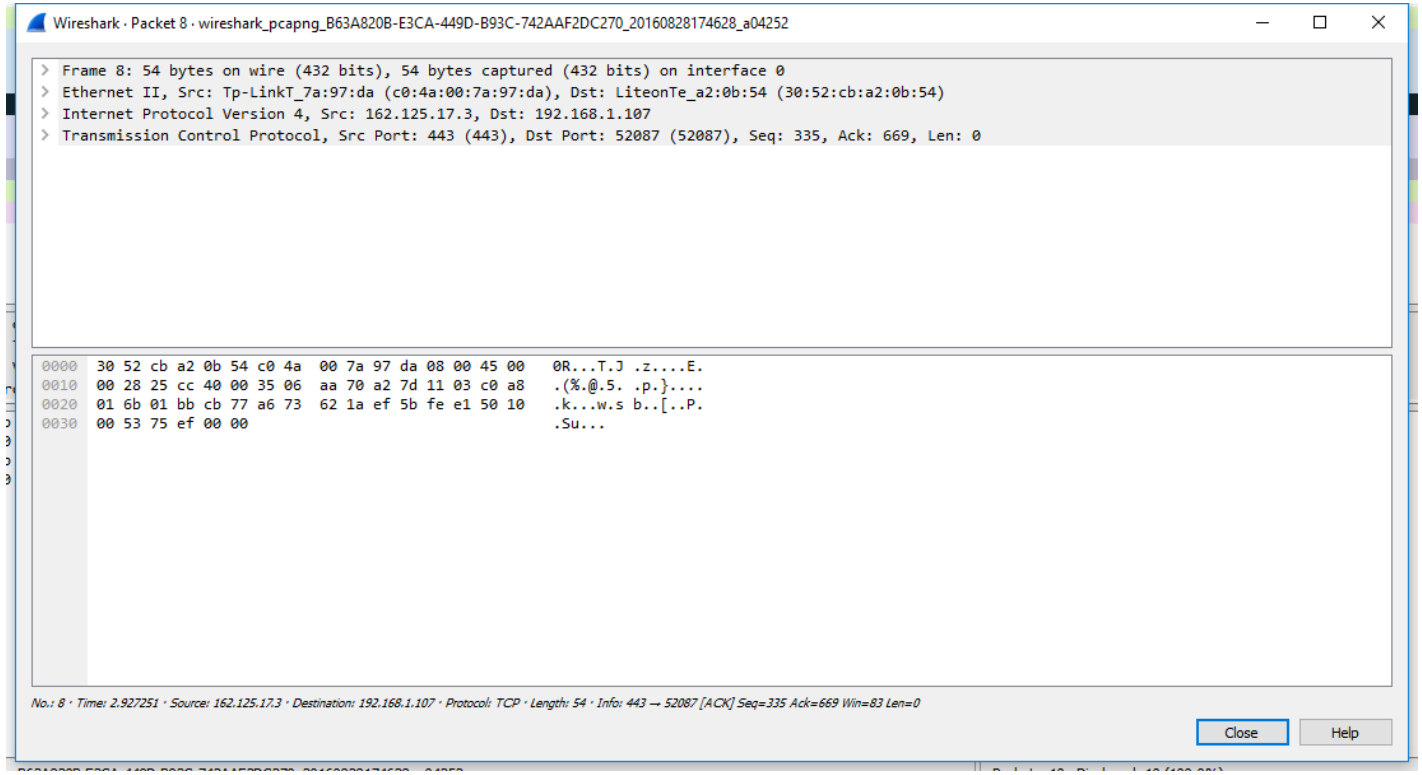
Wireshark


Filter
Start capture
Select interface
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
Ethernet 2
Bluetooth Network Connection
Wi-Fi
Ethernet
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
You are running Wireshark 2.0.4 (v2.0.4-0-gdd7746e from master-2.0). You receive automatic updates.
Ready to load or capture
No Packets
Profile: Default

Wireshark


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
No. Time Source Destination Protocol Length Info
1 0.000000 fe80::741f:6d7a:7ca... ff02::c SSDP 208 M-SEARCH * HTTP/1.1
2 1.077743 192.168.1.1 255.255.255.255 UDP 215 32772 → 7437 Len=173
3 2.324186 192.168.1.107 10.66.30.153 SNMP 120 get-request 1.3.6.1.2.1.25.3.2.1.5.1 1.3.6.1.2.1.25.3.5.1.1.1 1.3.6.1.2.1.25.3.5.1.2.1
4 2.324350 192.168.1.107 10.66.30.153 SNMP 120 get-request 1.3.6.1.2.1.25.3.2.1.5.1 1.3.6.1.2.1.25.3.5.1.1.1 1.3.6.1.2.1.25.3.5.1.2.1
5 2.344659 150.101.32.108 192.168.1.107 ICMP 70 Destination unreachable (Communication administratively filtered)
6 2.668124 162.125.17.3 192.168.1.107 TLSv1.2 388 Application Data
7 2.674726 192.168.1.107 162.125.17.3 TLSv1.2 722 Application Data
8 2.927251 162.125.17.3 192.168.1.107 TCP 54 443 → 52087 [ACK] Seq=335 Ack=669 Win=83 Len=0
9 3.014762 fe80::741f:6d7a:7ca... ff02::c SSDP 208 M-SEARCH * HTTP/1.1
10 3.061176 fe80::1 ff02::1 ICMPv6 78 Router Advertisement from c0:4a:00:7a:97:da
Frame 8: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Tp-LinkT_7a:97:da (c0:4a:00:7a:97:da), Dst: LiteonTe_a2:0b:54 (30:52:cb:a2:0b:54)
Internet Protocol Version 4, Src: 162.125.17.3, Dst: 192.168.1.107
Transmission Control Protocol, Src Port: 443 (443), Dst Port: 52087 (52087), Seq: 335, Ack: 669, Len: 0
0000 30 52 cb a2 0b 54 c0 4a 00 7a 97 da 08 00 45 00 0R...T.J .z....E.
0010 00 28 25 cc 40 00 35 06 aa 70 a2 7d 11 03 c0 a8 .(%.@.5. .p.}....
0020 01 6b 01 bb cb 77 a6 73 62 1a ef 5b fe e1 50 10 .k...w.s b..[..P.
0030 00 53 75 ef 00 00 .Su...
wireshark_pcapng_B63A820B-E3CA-449D-B93C-742AAF2DC270_20160828174628_a04252
Packets: 10 · Displayed: 10 (100.0%)
Profile: Default

Wireshark


Wireshark · Packet 8 · wireshark_pcapng_B63A820B-E3CA-449D-B93C-742AAF2DC270_20160828174628_a04252
Frame 8: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Tp-LinkT_7a:97:da (c0:4a:00:7a:97:da), Dst: LiteonTe_a2:0b:54 (30:52:cb:a2:0b:54)
Internet Protocol Version 4, Src: 162.125.17.3, Dst: 192.168.1.107
Transmission Control Protocol, Src Port: 443 (443), Dst Port: 52087 (52087), Seq: 335, Ack: 669, Len: 0
0000 30 52 cb a2 0b 54 c0 4a 00 7a 97 da 08 00 45 00 0R...T.J .z....E.
0010 00 28 25 cc 40 00 35 06 aa 70 a2 7d 11 03 c0 a8 .(%.@.5. .p.}....
0020 01 6b 01 bb cb 77 a6 73 62 1a ef 5b fe e1 50 10 .k...w.s b..[..P.
0030 00 53 75 ef 00 00 .Su...
No.: 8 · Time: 2.927251 · Source: 162.125.17.3 · Destination: 192.168.1.107 · Protocol: TCP · Length: 54 · Info: 443 → 52087 [ACK] Seq=335 Ack=669 Win=83 Len=0
Close
Help

Wireshark

– Today: If you are unable to run Wireshark live on a computer, you can download trace file lab-trace-1 in the Canvas.