CSCI-1680
Network Layer:
IP Forwarding realities

Nick DeMarinis

Based partly on lecture notes by Rodrigo Fonseca, David Mazières, Phil Levis, John Jannotti

Administrivia

- Sign up for IP milestone meetings, preferably with your mentor TA, on or before Friday (Oct 6)
  - You don’t need to show an implementation, but you are expected to talk about your design
  - Look for calendar link in email
- IP gearup II: Thursday 5-7pm in CIT368
  - Implementation and debugging tips

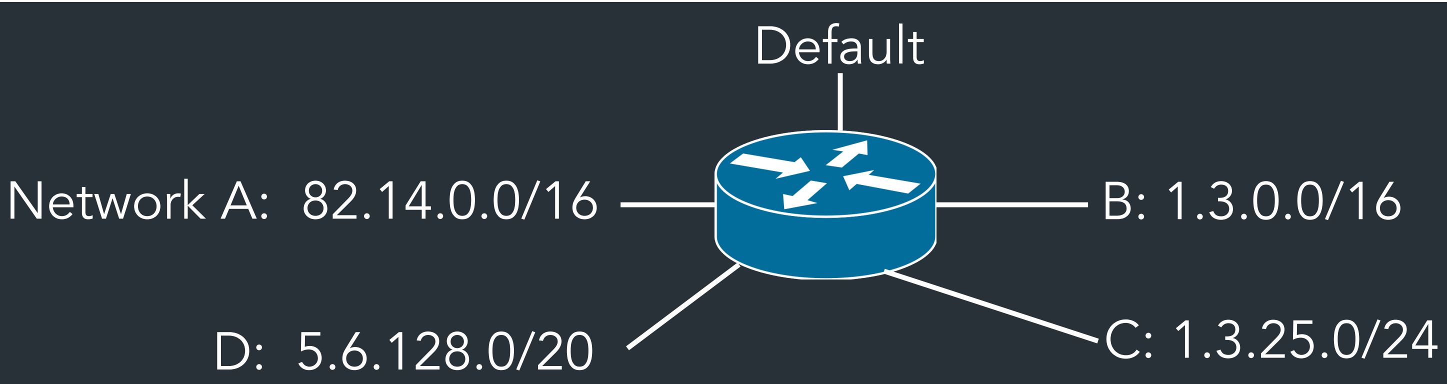
- HW1: Due Thursday (HW2 out either Thursday or next Tues)

Today

"Wrinkles" in IP forwarding

- Longest Prefix Match
- IP<->Link layer (ARP, DHCP)
- Network Address Translation (NAT)
- IPv6

After this: Routing



Default

Network A: 82.14.0.0/16

B: 1.3.0.0/16

C: 1.3.25.0/24

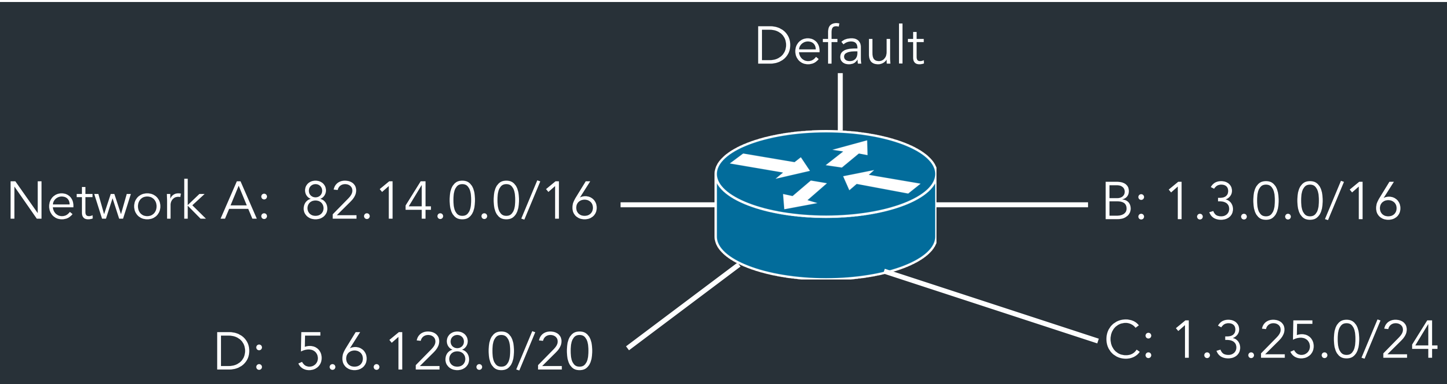
D: 5.6.128.0/20

Prefix	IF/Next hop
82.14.0.0/16	(A)
1.3.0.0/16	(B)
1.3.4.0/24	(C)
5.6.128.0/20	(D)
0.0.0.0/0	(Default)

(X) is placeholder—could be an IP or an interface name

Warmup: based on the table, where would the router send packets destined for the following addresses:

1. 5.6.128.100
2. 1.3.1.1
3. 8.8.8.8



Default

Network A: 82.14.0.0/16

B: 1.3.0.0/16

D: 5.6.128.0/20

C: 1.3.25.0/24

Prefix	IF/Next hop
82.14.0.0/16	(A)
1.3.0.0/16	(B)
1.3.4.0/24	(C)
5.6.128.0/20	(D)
0.0.0.0/0	(Default)

(X) is placeholder—could be an IP or an interface name

Warmup: based on the table, where would the router send packets destined for the following addresses:

1. 5.6.128.100 D
2. 1.3.1.1 B
3. 8.8.8.8 DEFAULT
4. 1.3.4.8

What happens when prefixes overlap?

Prefix	IF/Next hop
1.3.0.0/16	(B)
1.3.4.0/24	(C)
1.3.4.5/32	
0.0.0.0/0	(Default)

An IP can match on more than one row
=> **<u>need to pick the most specific (longest) prefix</u>**

1.3.0.0/16 00000001 00000011 xxxxxxxx xxxxxxxx

1.3.4.0/24 00000001 00000011 00000100 xxxxxxxx

More specific => best match!

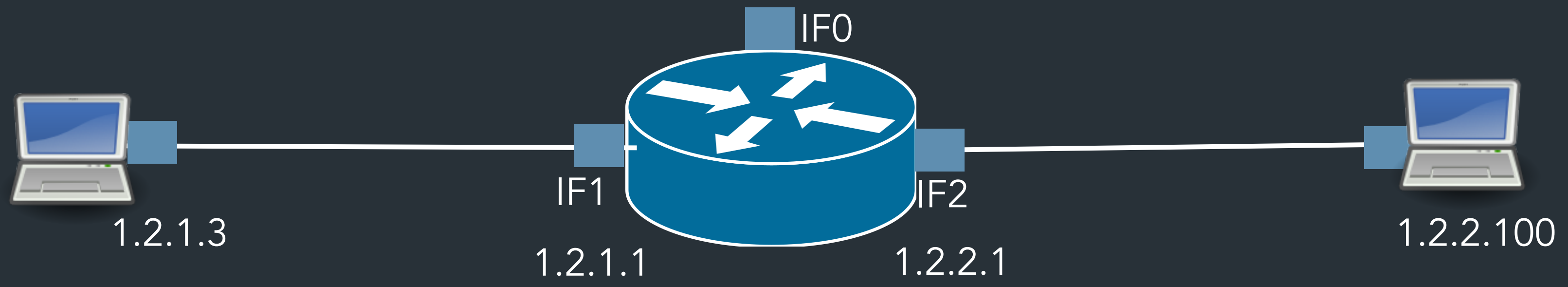
Other examples you'll see…

0.0.0.0/0 xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx => Least specific!
(Used for default "catchall" routes)

1.2.3.5/32 00000001 00000011 00000100 00000101 => Most specific!
(Refers to a single host,
often a local IP)

=><u>Longest prefix matching</u>: can keep forwarding tables small by summarizing routes where possible, otherwise using specific prefixes

What happens at the link layer?



What does it mean to send to IF1?

Prefix	IF/Next hop
1.2.1.0/24	IF1
1.2.2.0/24	IF2
8.0.0.0/30	IF0
Default	8.0.0.2

"Local delivery": what does it mean to send to IF1?

So far: "easy" to communicate with nodes on the same network. But how?

IN ORDER TO SEND ON LOCAL NET, NEED:

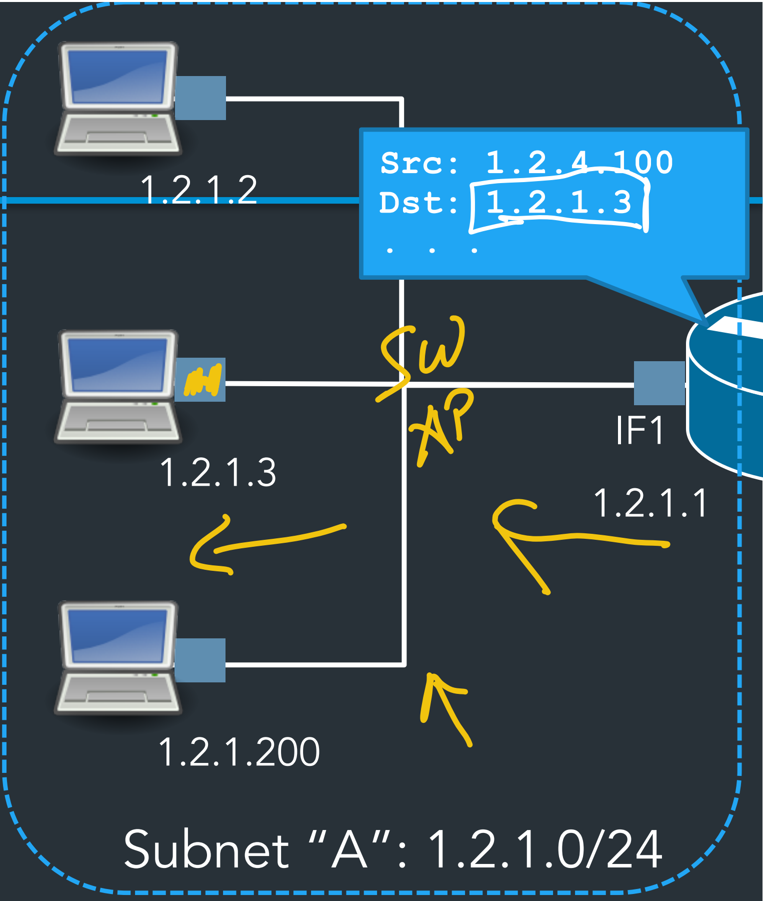
- DEST IP (L3)
- DEST MAC ADDRESS

	Src	Dest
Link	Know	???
IP	10.2.4.100	1.2.1.3

ETH/WIFI/...

HEADER INFO

NEED TO FIND THIS SOMEHOW.



Src: 1.2.4.100
Dst: 1.2.1.3
. . .

Prefix	IF/Next hop
1.2.1.0/24	IF1
...	...

“Glue” between L2 and L3

ETH/WIFI/... ~ IP

Need a way to connect get link layer info (mac address) from network-layer info (IP address)

“What MAC address has IP 1.2.3.4?”

“Glue” between L2 and L3

Need a way to connect get link layer info (mac address) from network-layer info (IP address)

“What MAC address has IP 1.2.3.4?”

Ask the network!

=> Address Resolution Protocol (ARP)

ARP: Address resolution protocol

Given an IP address, ask network for the MAC address

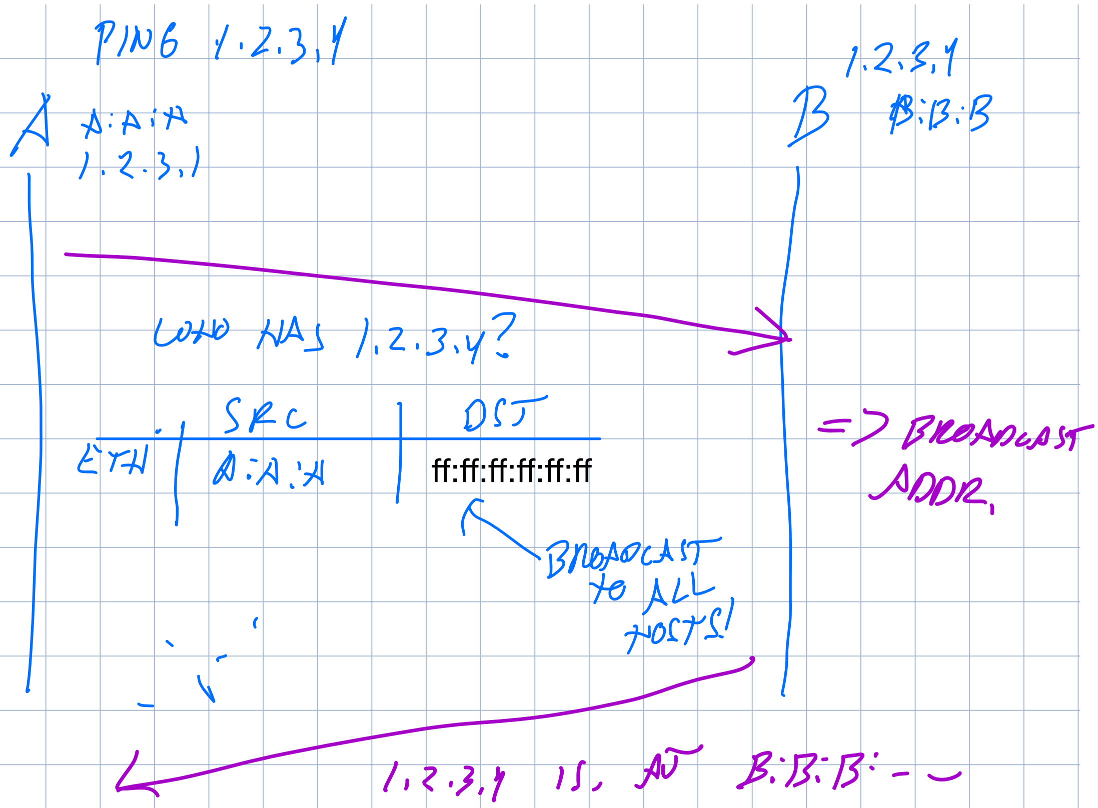
- Maps IP addresses to mac addresses
  - Request: “`Who has 1.2.3.4?`”
  - Response: “`aa:bb:cc:dd:ee:ff is at 1.2.3.4`”

ARP: Address resolution protocol

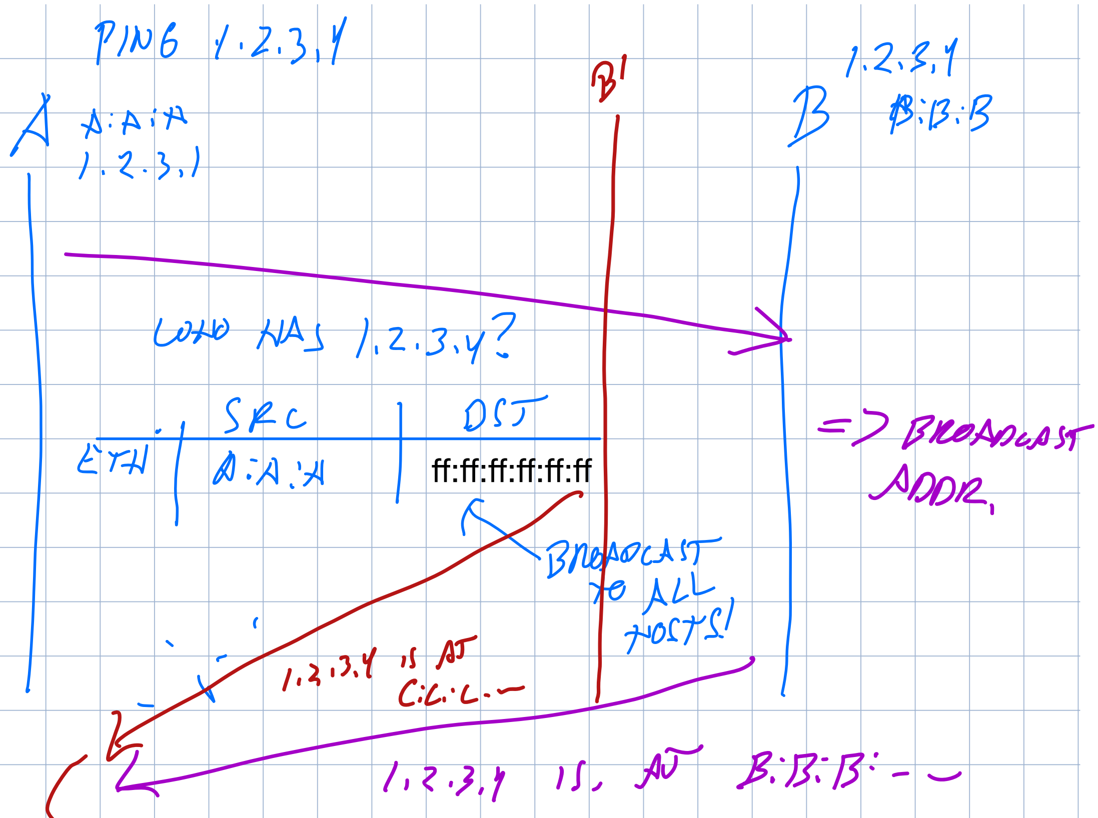
Given an IP address, ask network for the MAC address

- Maps IP addresses to mac addresses
  - Request: “`Who has 1.2.3.4?`”
  - Response: “`aa:bb:cc:dd:ee:ff is at 1.2.3.4`”

- ARP table: hosts cache IP->mac mappings
- Requests send to broadcast address: `ff:ff:ff:ff:ff:ff`
  - Anyone can respond: problem?



	SRC	DST
ETH	A:A:A	ff:ff:ff:ff:ff:ff



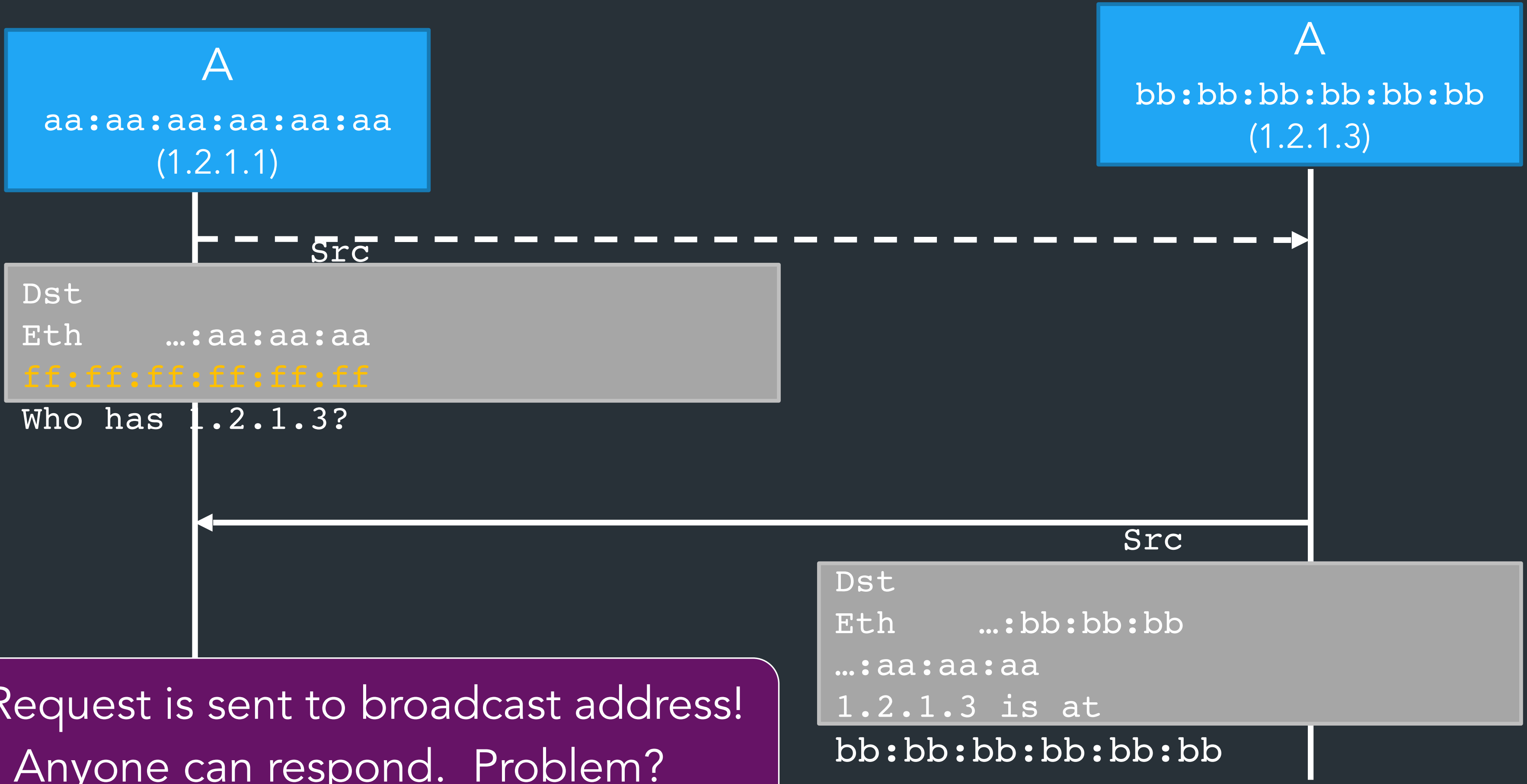
ARP POISIONING.

=> MIGHT BE POSSIBLE TO DETECT SINCE RESPONSES ARE SENT VIA BROADCAST.

Responses are cached at the host in the ARP table:

Maps IP => MAC address

Then when you send the next packet, check the ARP table for the MAC address
If table miss, send an ARP request



A

aa:aa:aa:aa:aa:aa

(1.2.1.1)

A

bb:bb:bb:bb:bb:bb

(1.2.1.3)

Src

Dst

Eth …:aa:aa:aa

ff:ff:ff:ff:ff:ff

Who has 1.2.1.3?

Src

Dst

Eth …:bb:bb:bb

…:aa:aa:aa

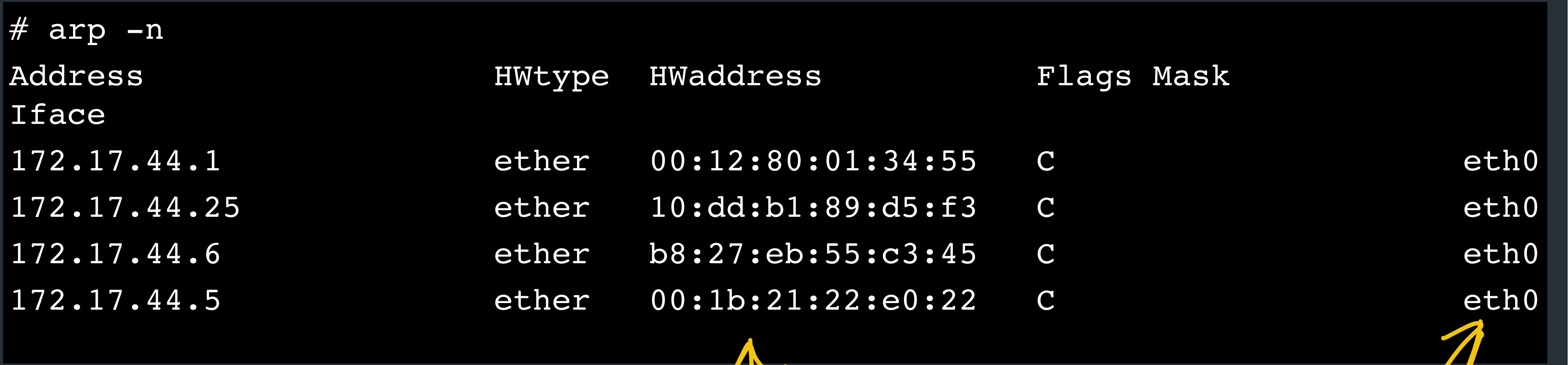
1.2.1.3 is at

bb:bb:bb:bb:bb:bb

⇒ Request is sent to broadcast address!

Anyone can respond. Problem?

Example



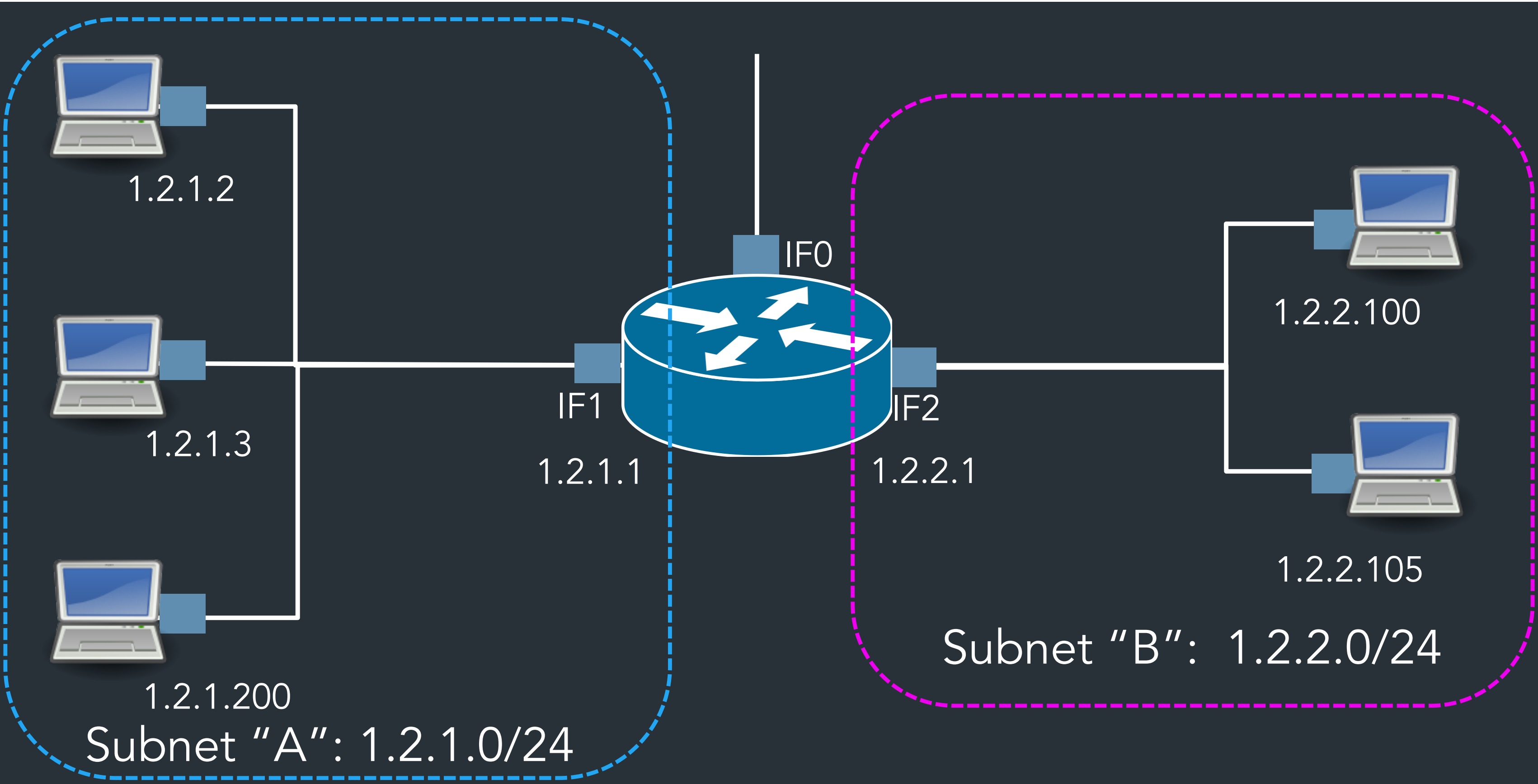
```
# arp -n
Address                 HWtype  HWaddress           Flags Mask
Iface
172.17.44.1             ether   00:12:80:01:34:55   C                     eth0
172.17.44.25            ether   10:dd:b1:89:d5:f3   C                     eth0
172.17.44.6             ether   b8:27:eb:55:c3:45   C                     eth0
172.17.44.5             ether   00:1b:21:22:e0:22   C                     eth0
```

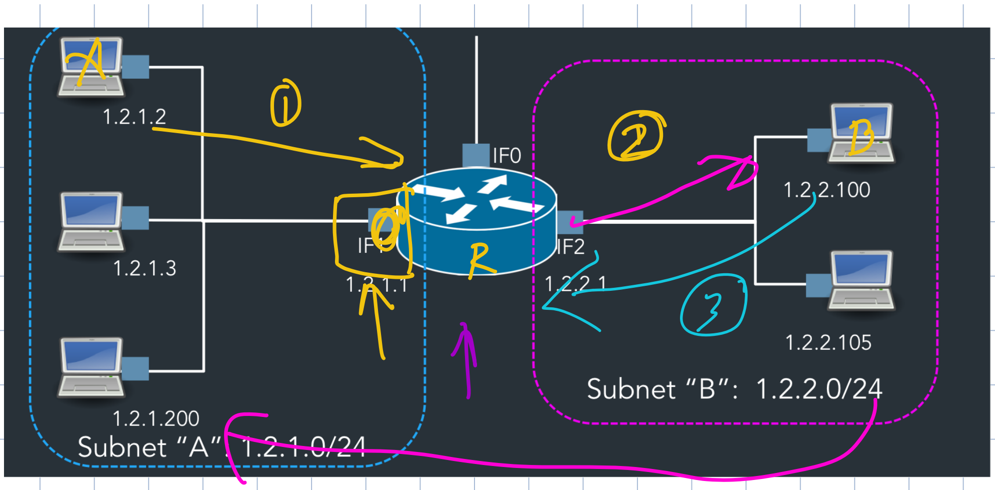
L3
(IP ADDRS)

L2

INTERFACE.

ALL ENTRIES SHOULD HAVE
TIMEOUT, ETC.


1.2.1.2
1.2.1.3
1.2.1.200
Subnet "A": 1.2.1.0/24
IF0
IF1
1.2.1.1
IF2
1.2.2.1
1.2.2.100
1.2.2.105
Subnet "B": 1.2.2.0/24



FROM 1.2.1.2 → 1.2.2.100

1

	SRC	DST
ETH	A:A:A	MAC ADDRESS OF IF1
IP	1.2.1.2	1.2.2.100

2

	SRC	DST
ETH	IF2	B:B:B
IP	~~1.2.1.2~~ 8.8.8.8	1.2.2.100

CHANGES AS WE CROSS LINKS

DOES NOT

END DST OF PACKET

NOT CHECKED BY DEFAULT!

B RESPONDS

3

	SRC	DST
ETH	B:B:B	IF2
IP	1.2.2.100	1.2.1.2

SRC ADDR IS USED FOR RESPONSE.

How do you get an IP address?

Getting an IP

Two ways to configure an IP for a host:

- Static configuration: manually specify IP address, mask, gateway, …

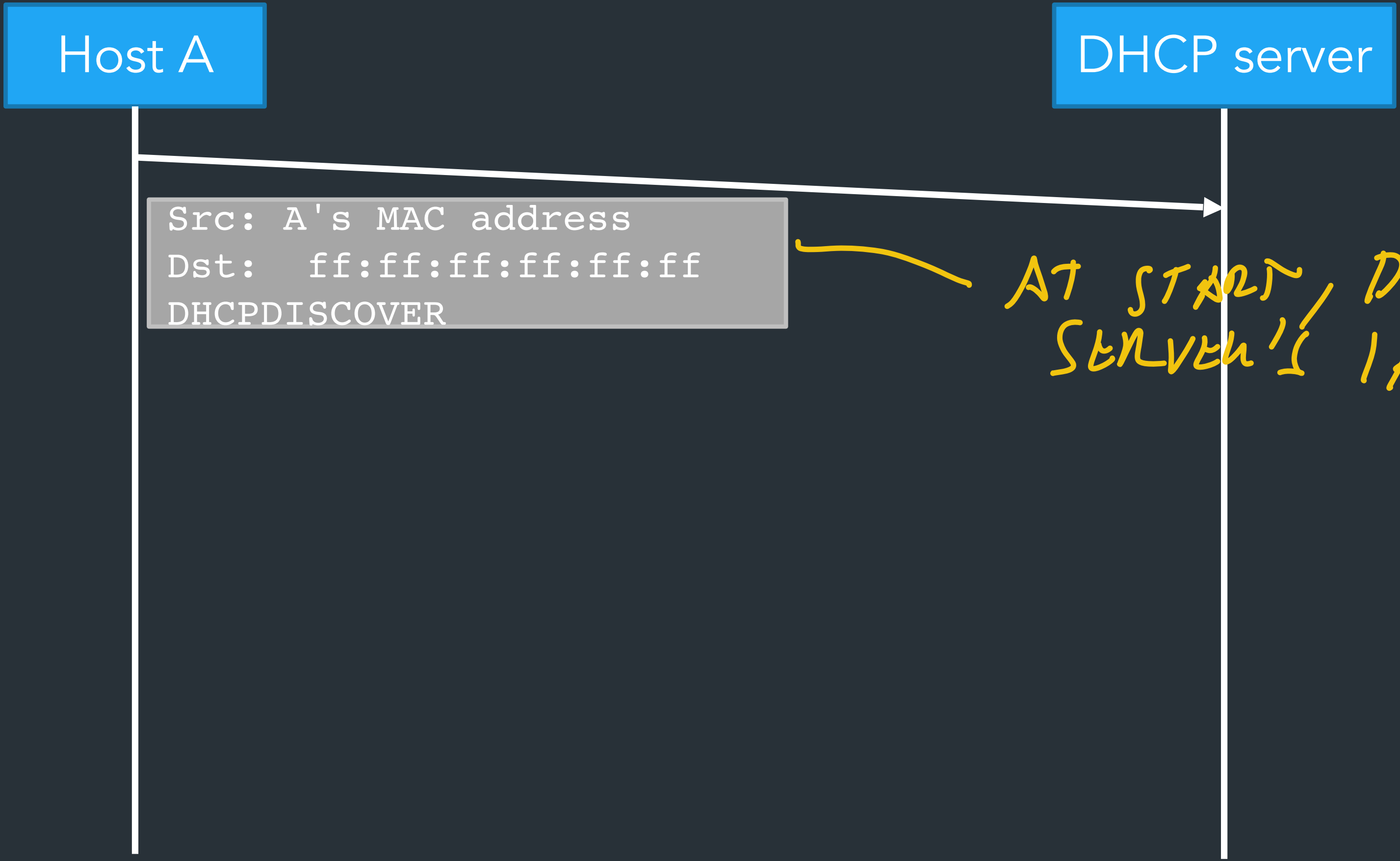
  => More common with network devices that don't change often

- Automatic: ask the network for an IP when you connect!

  => Most common for end hosts

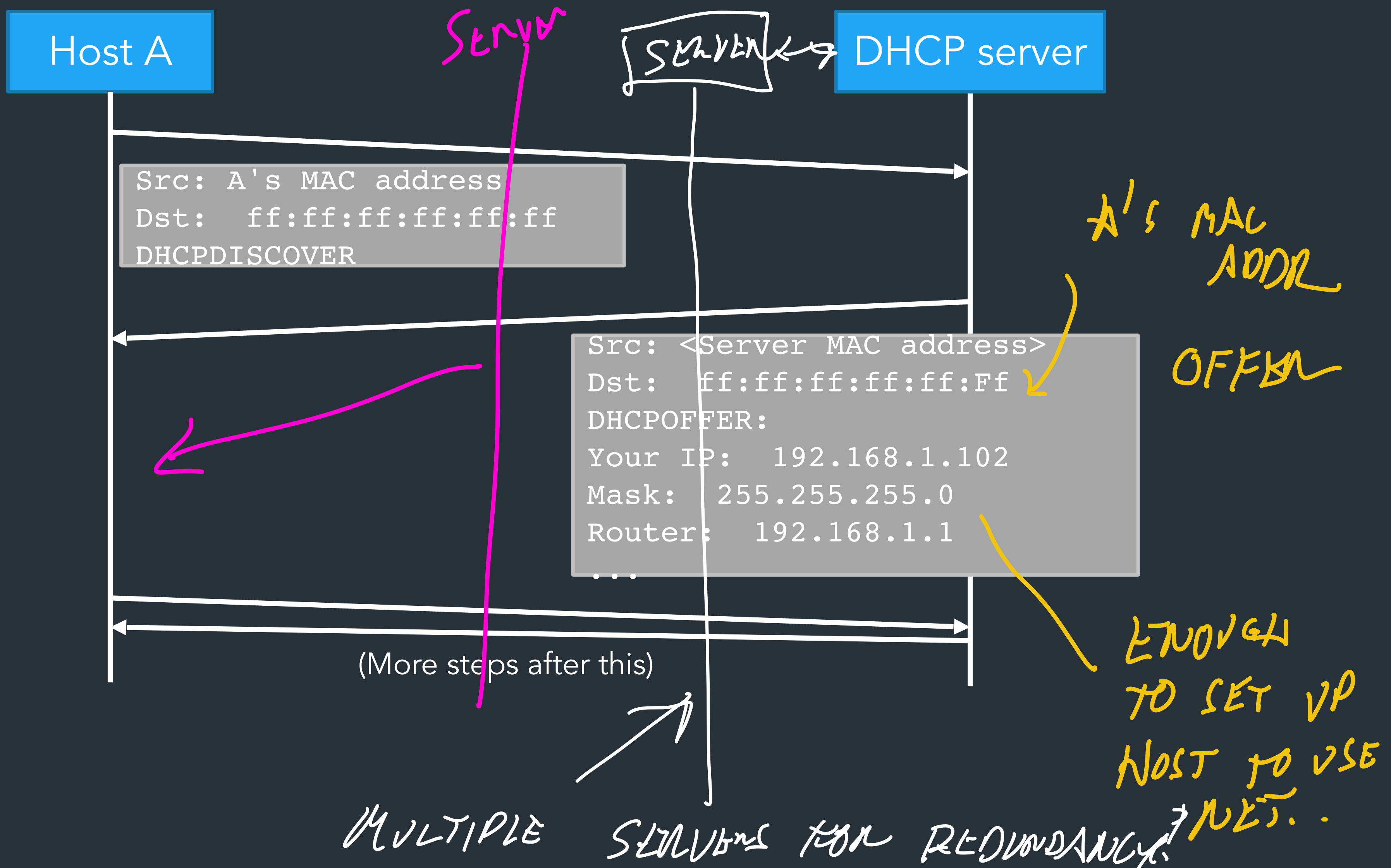
  => Dynamic Host Configuration Protocol (DHCP)

END HOSTS, HOME ROUTERS...



```
Src: A's MAC address
Dst:  ff:ff:ff:ff:ff:ff
DHCPDISCOVER
```

AT START, DON'T KNOW SERVER'S IP!



```
Src: A's MAC address
Dst:  ff:ff:ff:ff:ff:ff
DHCPDISCOVER
```

```
Src: <Server MAC address>
Dst:  ff:ff:ff:ff:ff:Ff
DHCPOFFER:
Your IP:  192.168.1.102
Mask:  255.255.255.0
Router:  192.168.1.1
...
```

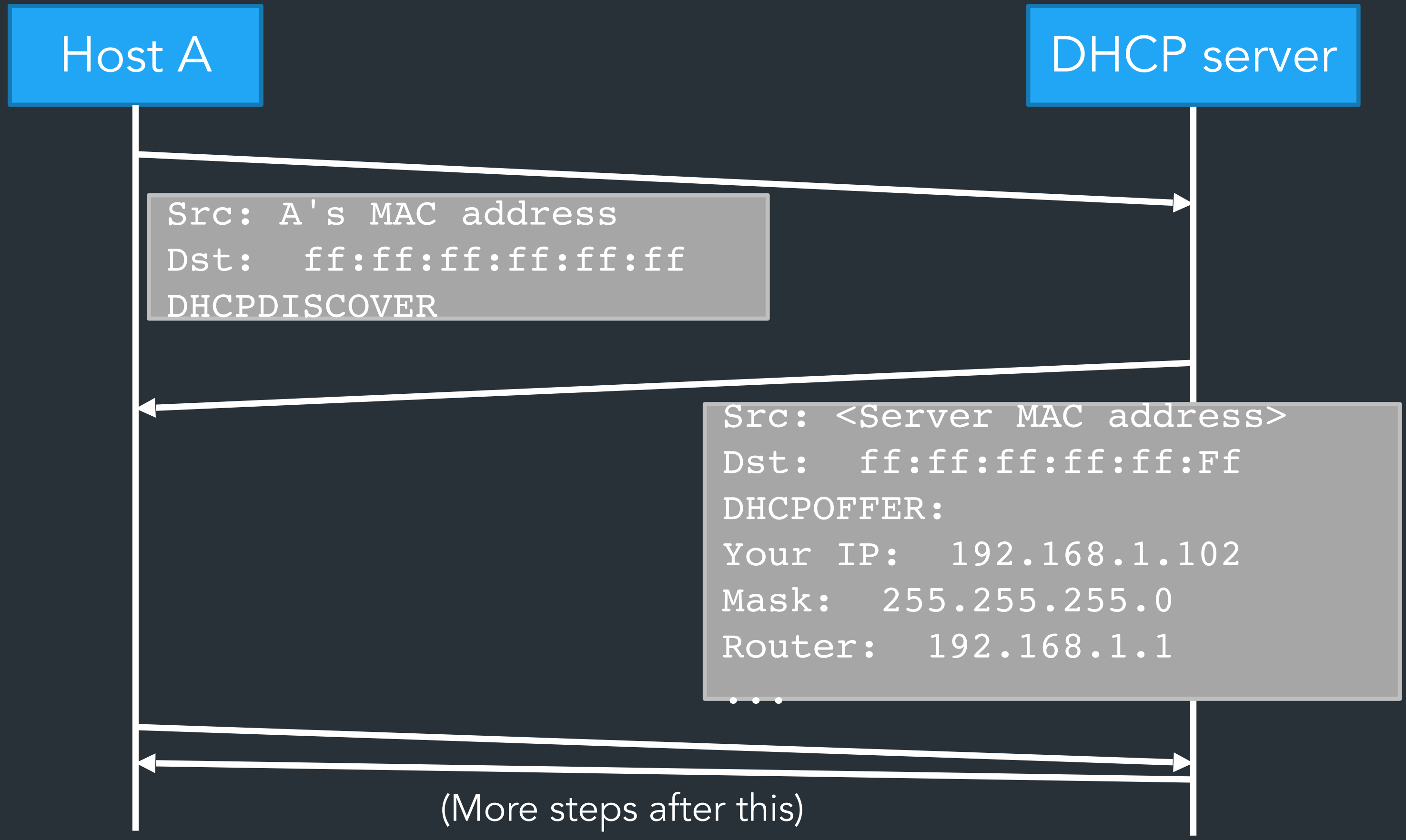
(More steps after this)

A's MAC ADDR

OFFER

ENOUGH TO SET UP HOST TO USE NET..

MULTIPLE SERVERS FOR REDUNDANCY?



```
Src: A's MAC address
Dst:  ff:ff:ff:ff:ff:ff
DHCPDISCOVER
```

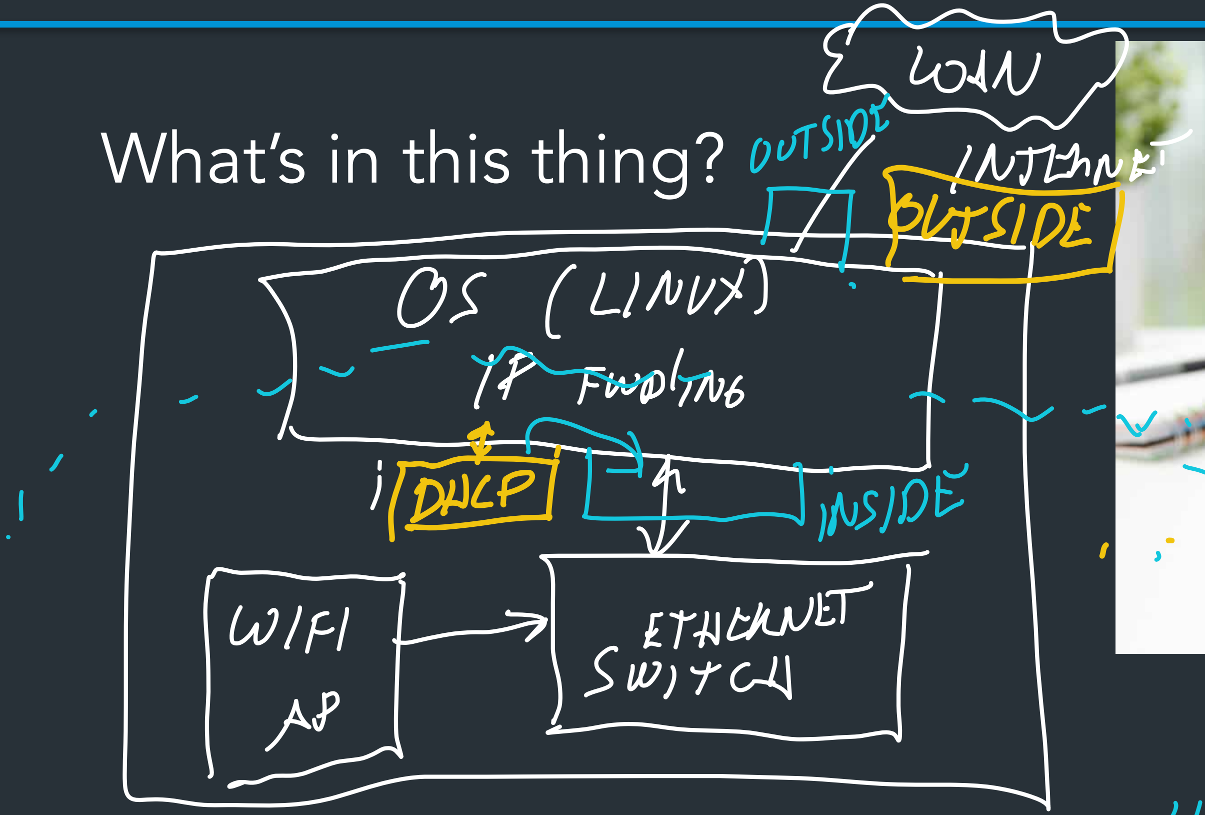
```
Src: <Server MAC address>
Dst:  ff:ff:ff:ff:ff:Ff
DHCPOFFER:
Your IP:  192.168.1.102
Mask:  255.255.255.0
Router:  192.168.1.1
...
```

=> Again, host needs to use broadcast address. Why?

=> Problem?

A home router

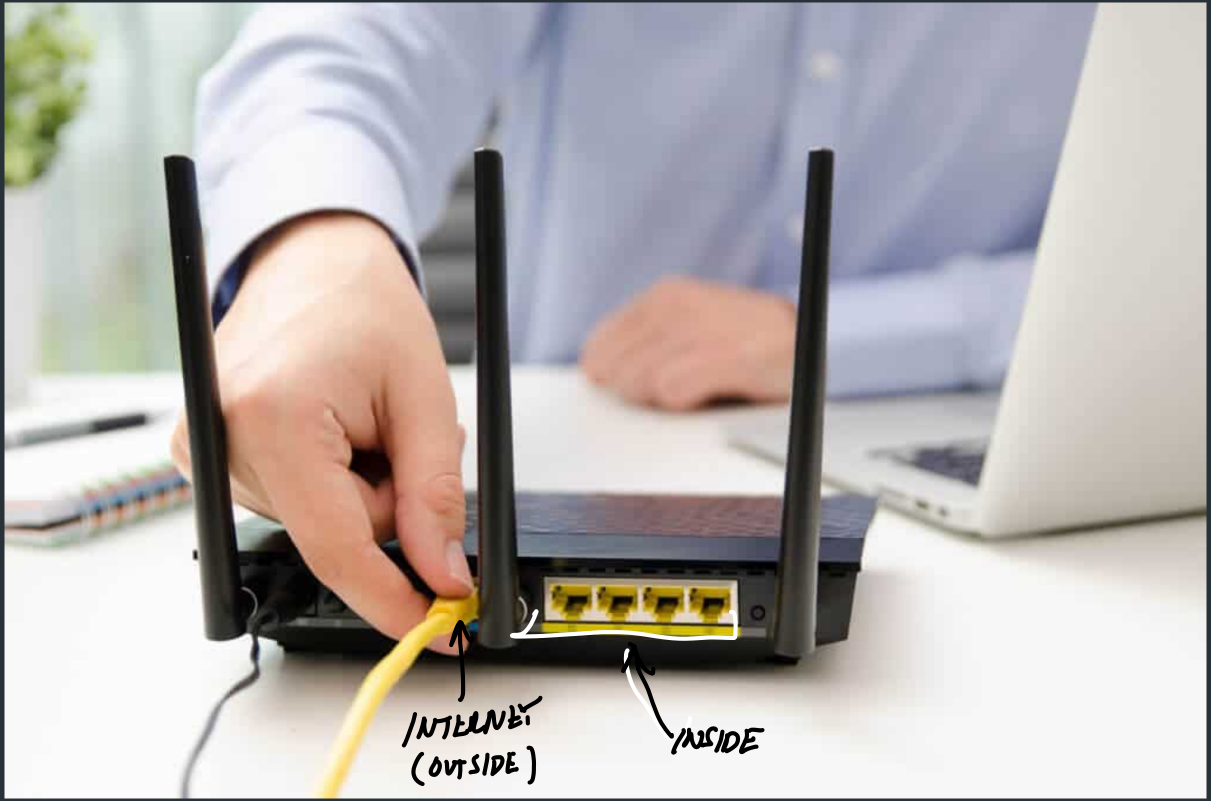
What's in this thing?



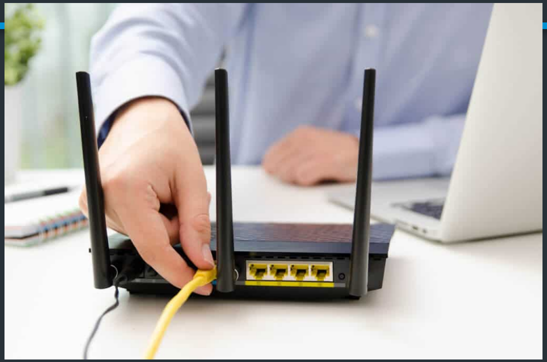
"INSIDE" "LOCAL NETWORK" LAN

192.168.1.0/24

Story time



About those home routers...

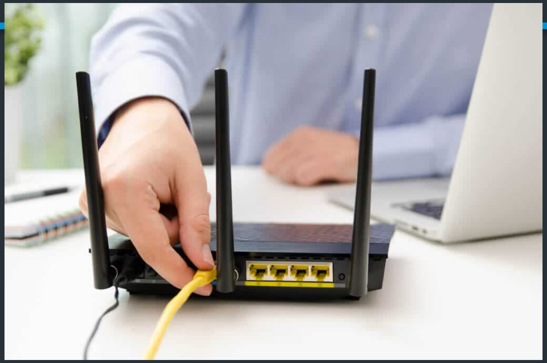


You get just one IP from your ISP...

=> Need to share IP among many devices on the same network!

YOU GET ONE IP ADDRESS FROM YOUR ISP.

About those home routers…



You get just one IP from your ISP…

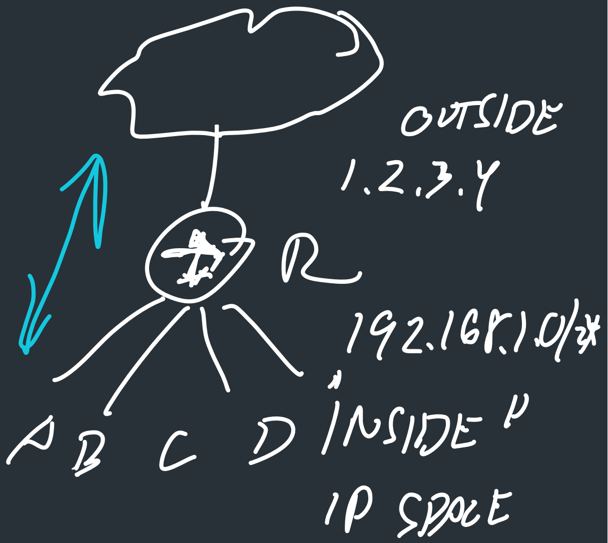
=> Need to share IP among many devices on the same network!

Common to create a "private" IP range used within local network

=> Routers need to do extra work to share public IP among private IPs

=> Network Address Translation (NAT)

(A form of connection multiplexing)



Private IPs (RFC1918)

Some IP ranges are reserved:

USED FOR INTERNAL STUFF

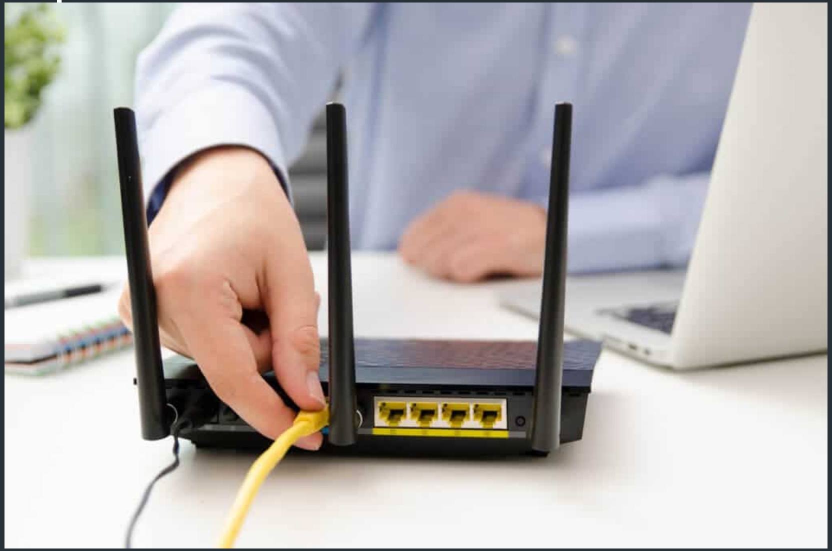
- HOME NETWORKS
- DOCKER

Prefix	Use
127.0.0.0/8	"Loopback" address—always for current host
10.0.0.0/8	Reserved for private internal networks (RFC1918)
192.168.0.0/16	
172.16.0.0/12	DOCKER

- Many networks will use these blocks internally

Network Address Translation

- What happens when hosts need to share an IP address?
- How to map private IP space to public IPs?

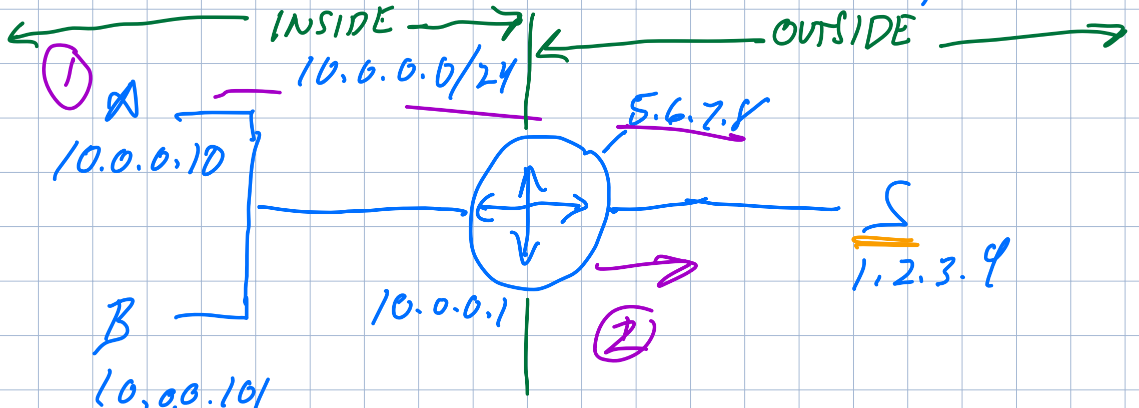


Network Address Translation (NAT)

- Despite CIDR, it's still difficult to allocate addresses (2^{32} is only 4 billion)
- NAT "hides" entire network behind one address
- Hosts are given private addresses
- Routers map outgoing packets to a free address/port
- Router reverse maps incoming packets
- Problems?

NAT Example

HOW NAT WORKS (IN GENERAL)



① INSIDE SRC	DST		OUTSIDE SRC	DST
10.0.0.1:5000	1.2.3.4:80 TCP	→	5.6.7.8:8888	1.2.3.4:80

① PACKET FROM A

② ROUTER TRANSLATES

ROUTER STORES:

10.0.0.1:5000 ⇒ 5.6.7.8:8888

↑ INSIDE IP

↑ OUTSIDE

↑ PORT THE ROUTER PICKS

③ RESPONSE FROM S

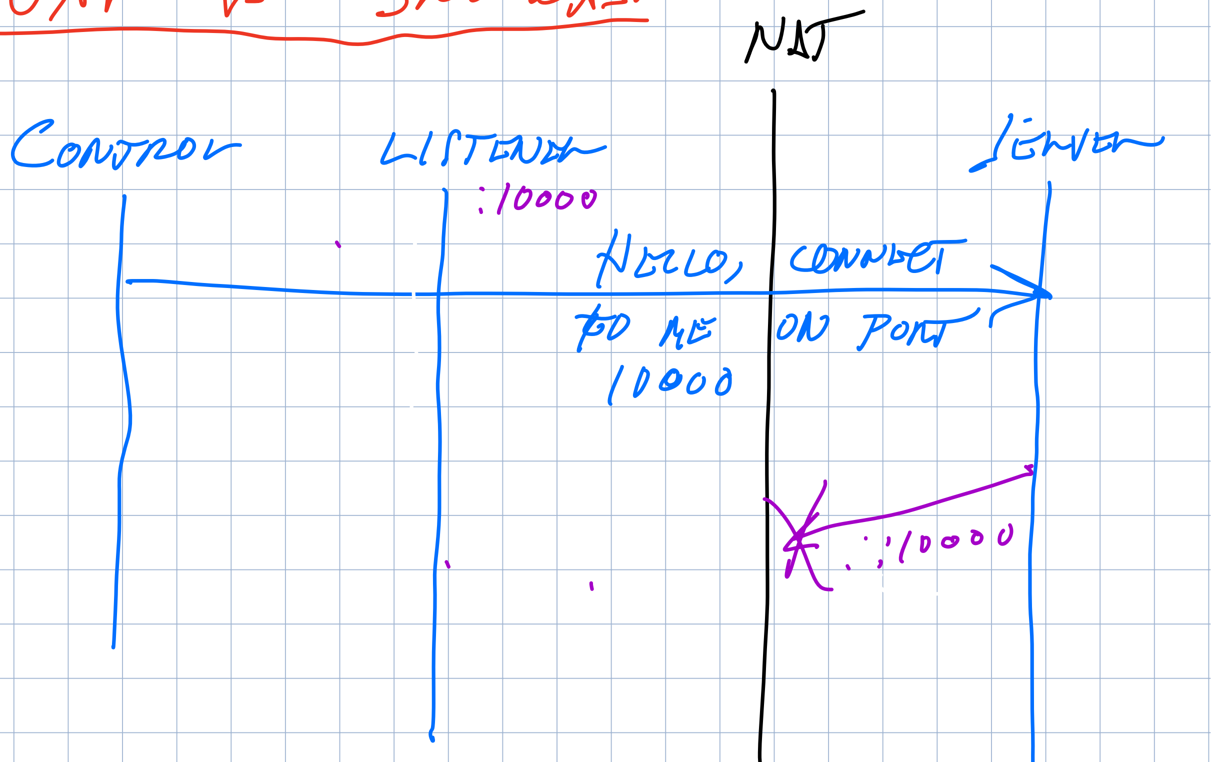
SRC	DST
1.2.3.4:80	5.6.7.8:8888
↓ NAT	
1.2.3.4:80	10.0.0.10:5000

ROUTER USES PORT NUMBERS TO "MULTIPLEX" CONNECTIONS TO ONE IP

END TO END CONNECTIVITY IS BROKEN!

- OUTSIDE HOST CAN'T CONNECT UNLESS INSIDE HOST STARTED A CONNECTION

NAT vs. SNOWCAST



- FTP
- VoIP
- GAMES.

NAT Traversal

Various methods, depending on the type of NAT

Examples:

- ICE: Interactive Connectivity Establishment (RFC8445)
- STUN: Session Traversal Utilities for NAT (RFC5389)

One idea: connect to external server via UDP, it tells you the address/port