

Section 6: Mininet II

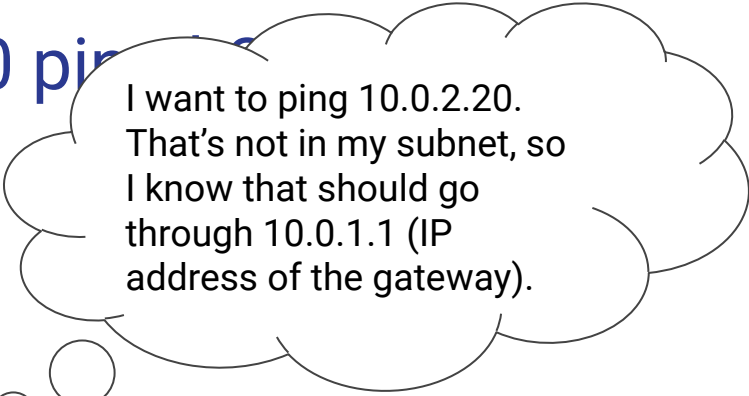
CSE 461 Computer Networks

Administrivia

- Project 2 is due Nov 22
- HW3 is due Nov 15



Part 4 - h10 ping



I want to ping 10.0.2.20.
That's not in my subnet, so
I know that should go
through 10.0.1.1 (IP
address of the gateway).

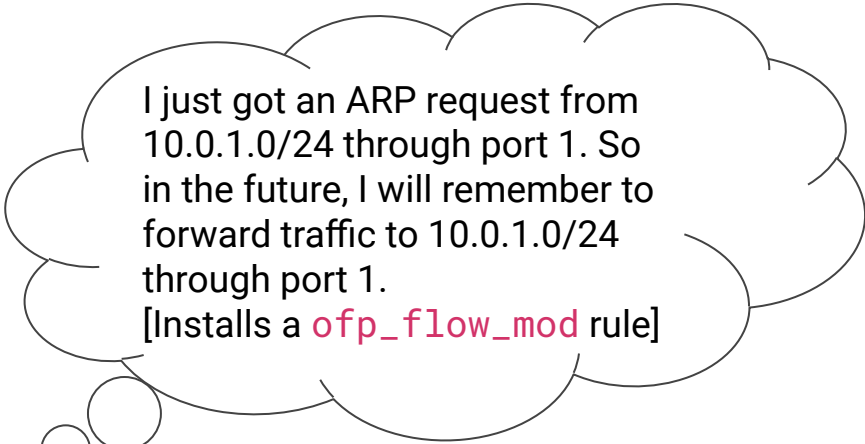
[h10@10.0.1.10/24]--{s1}--\

[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]

[h30@10.0.3.30/24]--{s3}--/

|
|
[hnotrust1@172.16.10.100/24]

Part 4 - h10 ping h20

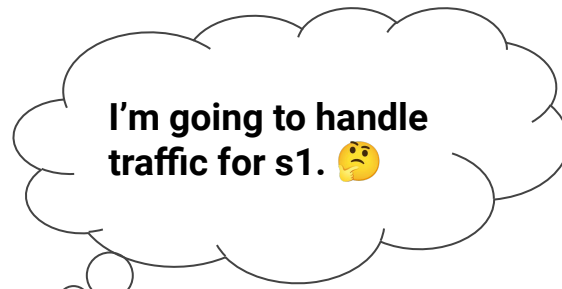


I just got an ARP request from 10.0.1.0/24 through port 1. So in the future, I will remember to forward traffic to 10.0.1.0/24 through port 1.

[Installs a `ofp_flow_mod` rule]

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```


Part 4 - h10 ping h20

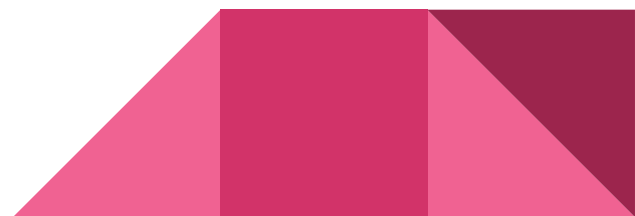


```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```

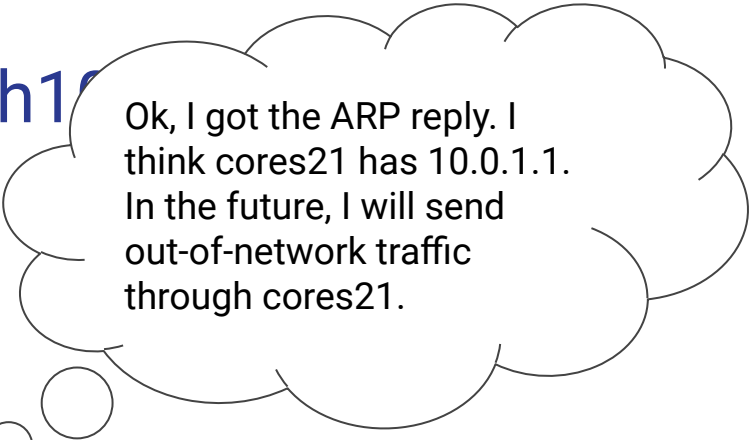
Part 4 - h10 ping h20

10.0.1.1 is at
de:ad:be:ef:ca:fe (I just
made that up, but I
replied so that's me
👉).

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
|  
[hnotrust1@172.16.10.100/24]
```



Part 4 - h10



Ok, I got the ARP reply. I think cores21 has 10.0.1.1. In the future, I will send out-of-network traffic through cores21.

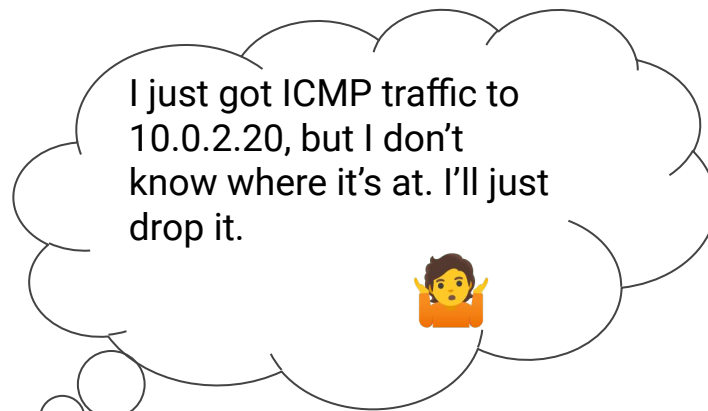
```
[h10@10.0.1.10/24]--{s1}--\
```

```
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]
```

```
[h30@10.0.3.30/24]--{s3}--/
```

```
      |  
      |  
[hnotrust1@172.16.10.100/24]
```

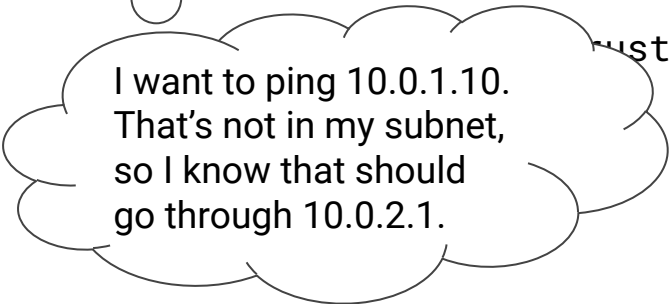

Part 4 - h10 ping h20



```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```


Part 4 - h20 ping h10

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/ |  
                               |  
                               [rust1@172.16.10.100/24]
```



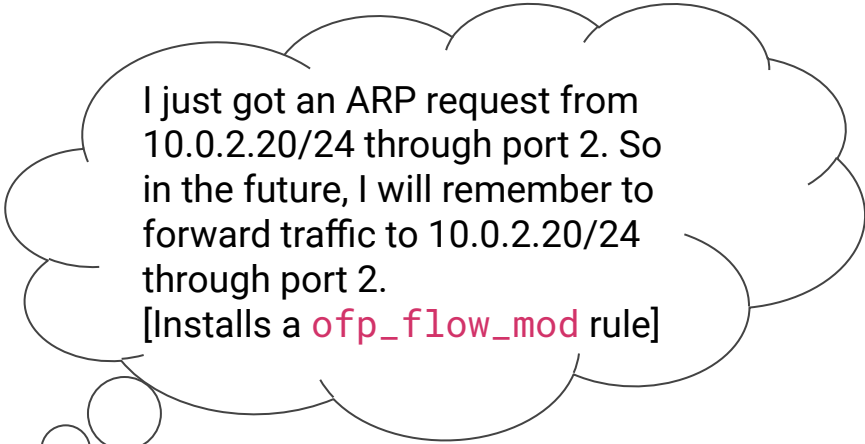
I want to ping 10.0.1.10.
That's not in my subnet,
so I know that should
go through 10.0.2.1.

Part 4 - h20 ping h10

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/ |  
                               |  
                               [hnotrust1@172.16.10.100/24]
```

ARP REQUEST:
Who is 10.0.2.1?
Tell 10.0.2.20

Part 4 - h20 ping h10



I just got an ARP request from 10.0.2.20/24 through port 2. So in the future, I will remember to forward traffic to 10.0.2.20/24 through port 2.

[Installs a `ofp_flow_mod` rule]

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```


Part 4 - h20 ping h10

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/ |  
                               |  
                               1@172.16.10.100/24]
```

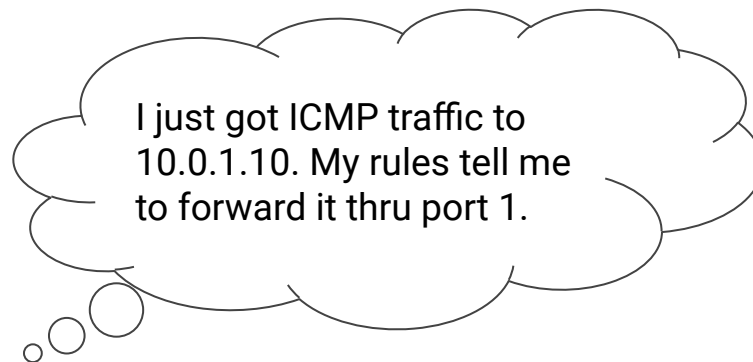
Ok, I got the ARP reply. I think cores21 has 10.0.2.1. In the future, I will send out-of-network traffic through cores21.

Part 4 - h20 ping h10

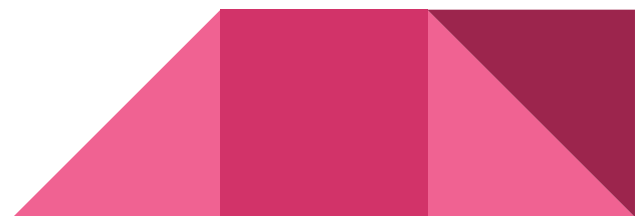
```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/ |  
                               |  
                               | hnotrust1@172.16.10.100/24]
```

Ping 10.0.1.10

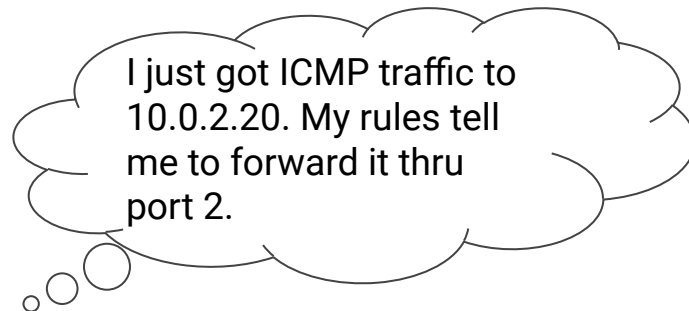
Part 4 - h20 ping h10



```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```



Part 4 - h20 ping h10



```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/  
|  
[hnotrust1@172.16.10.100/24]
```

Part 4 - h20 ping h10

```
[h10@10.0.1.10/24]--{s1}--\  
[h20@10.0.2.20/24]--{s2}--{cores21}--{dcs31}--[serv1@10.0.4.10/24]  
[h30@10.0.3.30/24]--{s3}--/ |  
                                |  
                                trust1@172.16.10.100/24]
```



Part 4 Summary

- `cores21` will respond to all ARP **requests**, claiming to be every sX, so it can forward all the IP/ICMP traffic.
- Once `cores21` knows where each host is, it will install a rule to forward IP traffic to that host through that port. (But **don't install duplicate rules**, b/c we don't want the rule table to grow with pings.)
- Therefore, pings to a host will always fail until `cores21` hears from that host.
- What will the output of `pingall` look like? What if we run `pingall` again?





Questions !

Consider the following forwarding table with 2 rules. For a packet with destination address 192.24.29.32, what would its next hop be?

Prefix	Next Hop
192.24.0.0/18	D
192.24.12.0/22	B



Consider the following forwarding table with 2 rules. For a packet with destination address 192.24.29.32, what would its next hop be?

Prefix	Next Hop
192.24.0.0/18	D
192.24.12.0/22	B

Route to D: 192.00011000.00xxxxxx.xxxxxxxx

Route to B: 192.00011000.000011xx.xxxxxxxx

Packet: 192.00011000.00011101.00100000

Hop D



Practice Problems | Subnetting

Suppose you are given the prefix 192.168.0.0/16. You are asked to split the prefix into exactly 4 equal subnets, labelled 1-4. Give the prefix of each subnet. Under which subnet would the address 192.168.151.12 fall?



To split the prefix into 4 equal subnets, we'll need to borrow 2 bits from the host portion of the address to uniquely identify 4 different subnets with the same prefix. Hence, the first subnet will fix its 17th and 18th network address bit (from the left) to be 00, second will be 01, third will be 10, fourth will be 11:

Subnet 1: 192.168.0.0/18

Subnet 2: 192.168.64.0/18

Subnet 3: 192.168.128.0/18

Subnet 4: 192.168.192.0/18

Looking at the different subnets, 192.168.151.12 would exist under subnet 3

