

“HANDS-ON SESSION BORDER GATEWAY PROTOCOL”

Ali AlSabeh, J. Crichigno
Department of Integrated Information Technology
University of South Carolina



NSF Award 1829698

“CyberTraining CIP: Cyberinfrastructure Expertise on High-throughput Networks for Big Science Data Transfers”

LAB SERIES: BORDER GATEWAY PROTOCOL

Lab Series: Border Gateway Protocol

- Lab 1: Introduction to Mininet
- Lab 2: Introduction to Free Range Routing (FRR)
- Lab 3: Introduction to BGP
- Lab 4: Configure and verify EBGP
- Lab 5: BGP Authentication
- Lab 6: Configure BGP with Default Route
- Lab 7: Using AS_PATH BGP Attribute
- Lab 8: Configuring IBGP and EBGP Sessions, Local Preference, and MED
- Lab 9: IBGP, Next Hop and Full Mesh Topology
- Lab 10: BGP Route Reflection
- Lab 11: Configuring Multiprotocol BGP
- **Lab 12: IP Spoofing and Mitigation Techniques**
- **Lab 13: BGP Hijacking**

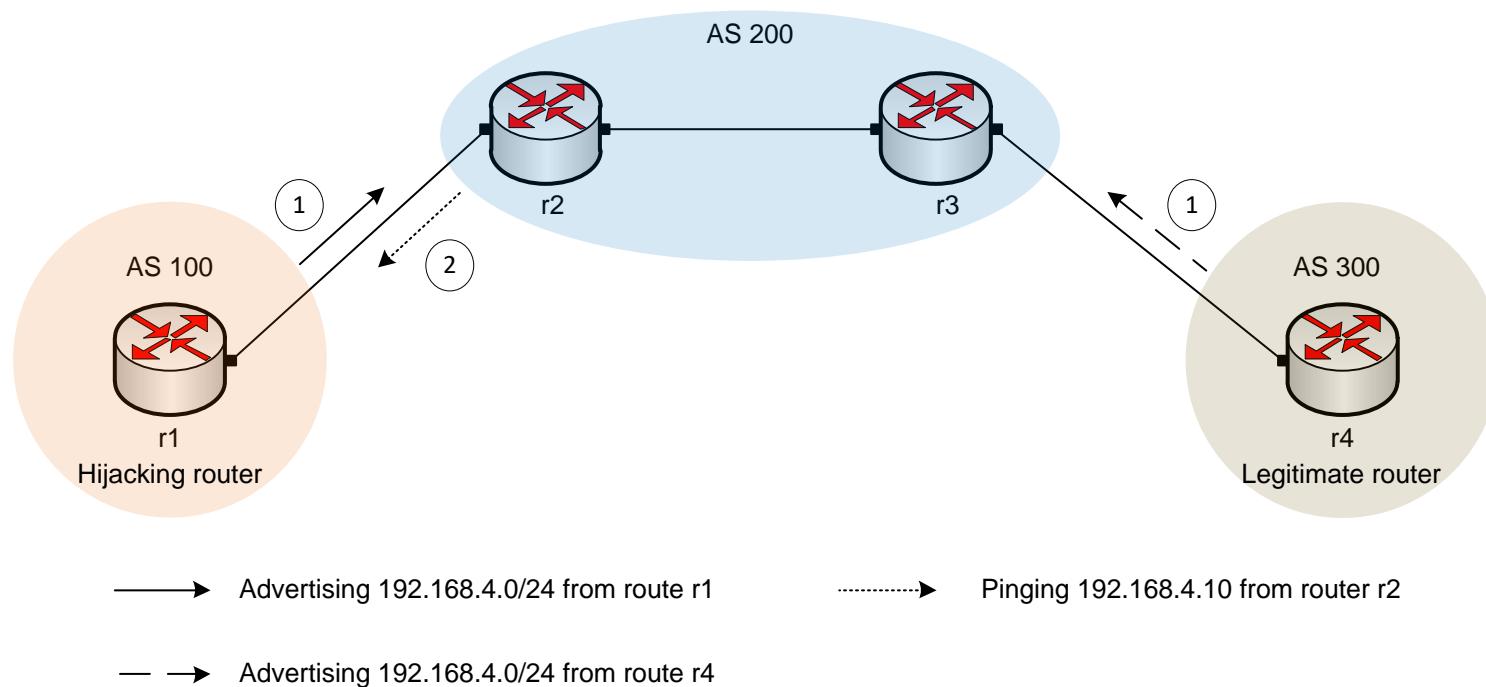
Organization of Lab Manuals

- Each lab starts with a section *Overview*
 - Objectives
 - Lab settings: passwords, device names
 - Roadmap: organization of the lab
- *Section 1*
 - Background information of the topic being covered (e.g., BGP hijacking)
 - Section 1 is optional (i.e., the reader can skip this section and move to lab directions)
- *Section 2... n*
 - Step-by-step directions

LAB 13: BGP HIJACKING

What is BGP Hijacking?

- BGP hijacking is when the attackers maliciously reroute Internet traffic
- It occurs when an unauthorized network originates IP prefix owned by other networks



BGP Hijacking Attacks

- Large scale BGP hijack out of India (2015)¹
 - 16,123 hijacked prefixes
- BGP hijack affected Amazon DNS (2018)²
 - 5 Amazon routes (prefixes) were affected
- Chinese Telecom performed a two hour BGP hijacking attack on European networks (2019)³
 - A significant portion of the traffic was routed through the Chinese Telecom infrastructure before reaching its destination
- Russian telecommunication provider rerouted traffic intended for several networks across the globe (2020)⁴
 - Over 8000 prefixes were rerouted from Cloudflare, Facebook, Google, Amazon, etc.

¹ Toonk, Andree "Large scale BGP hijack out of India". www.bgpmon.net/massive-route-leak-cause-internet-slowdown/

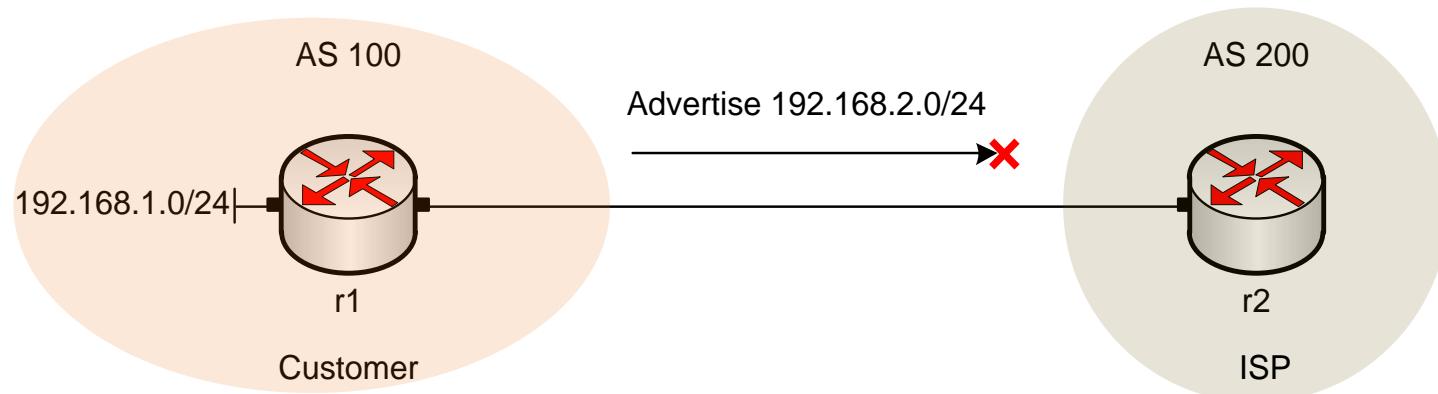
² Nichols Shaun, "AWS DNS network hijack turns MyEtherWallet into ThievesEtherWallet",

³ Linssen, R. H. H. G. M. "Vulnerability of DNS name servers against BGP hijacking." Bachelor's thesis, University of Twente, 2020

⁴ Improta Alessandro, Sani Luca "April Fools' BGP Hijack". <https://blog.catchpoint.com/2020/04/06/april-fools-bgp-hijack/>

Using IP Prefix Filters to Mitigate BGP Hijacking

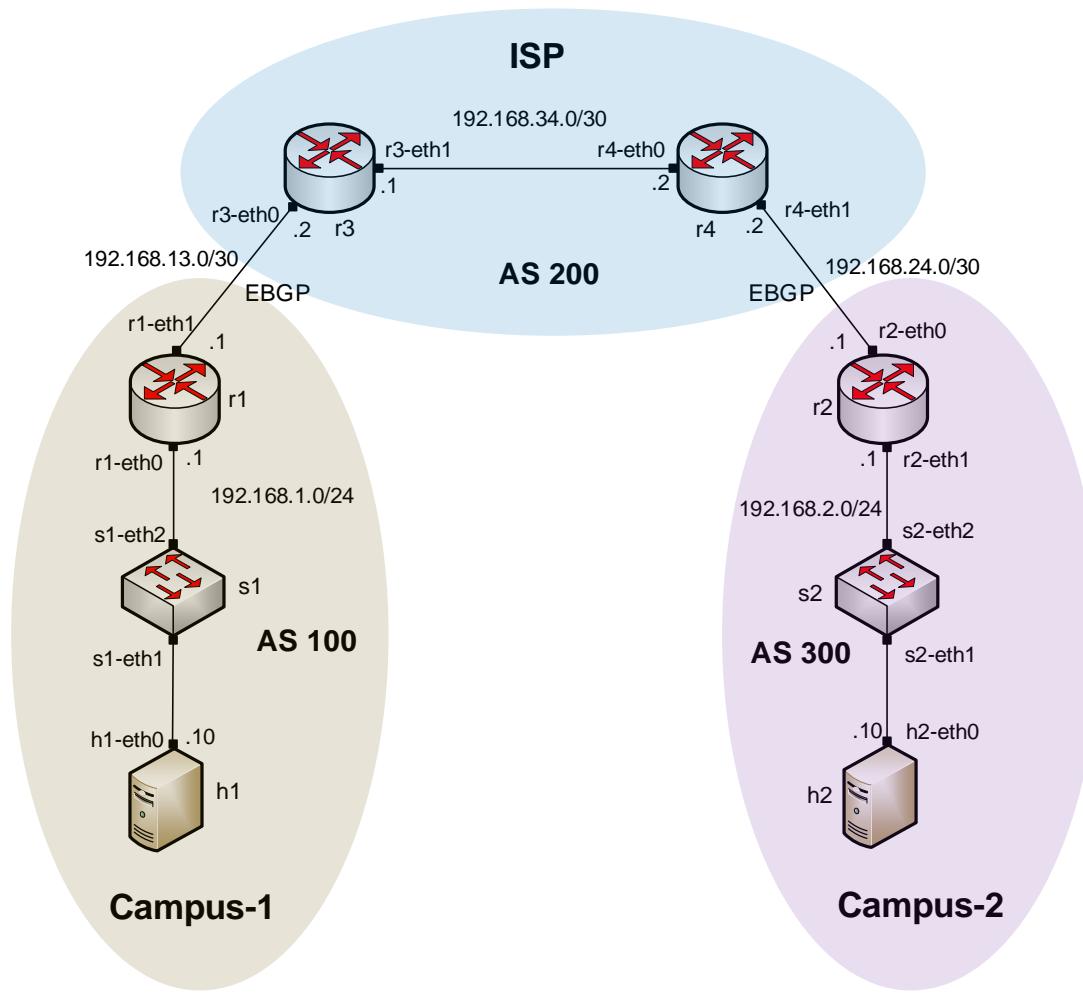
- A router can limit the number of BGP route advertisements by configuring IP prefix filters
- “Most important is to secure the inbound routing advertisements, particularly from customer networks, through the use of explicit prefix-level filters...”¹



¹ “MANRS Implementation Guide”. <https://www.manrs.org/isps/guide/filtering/>

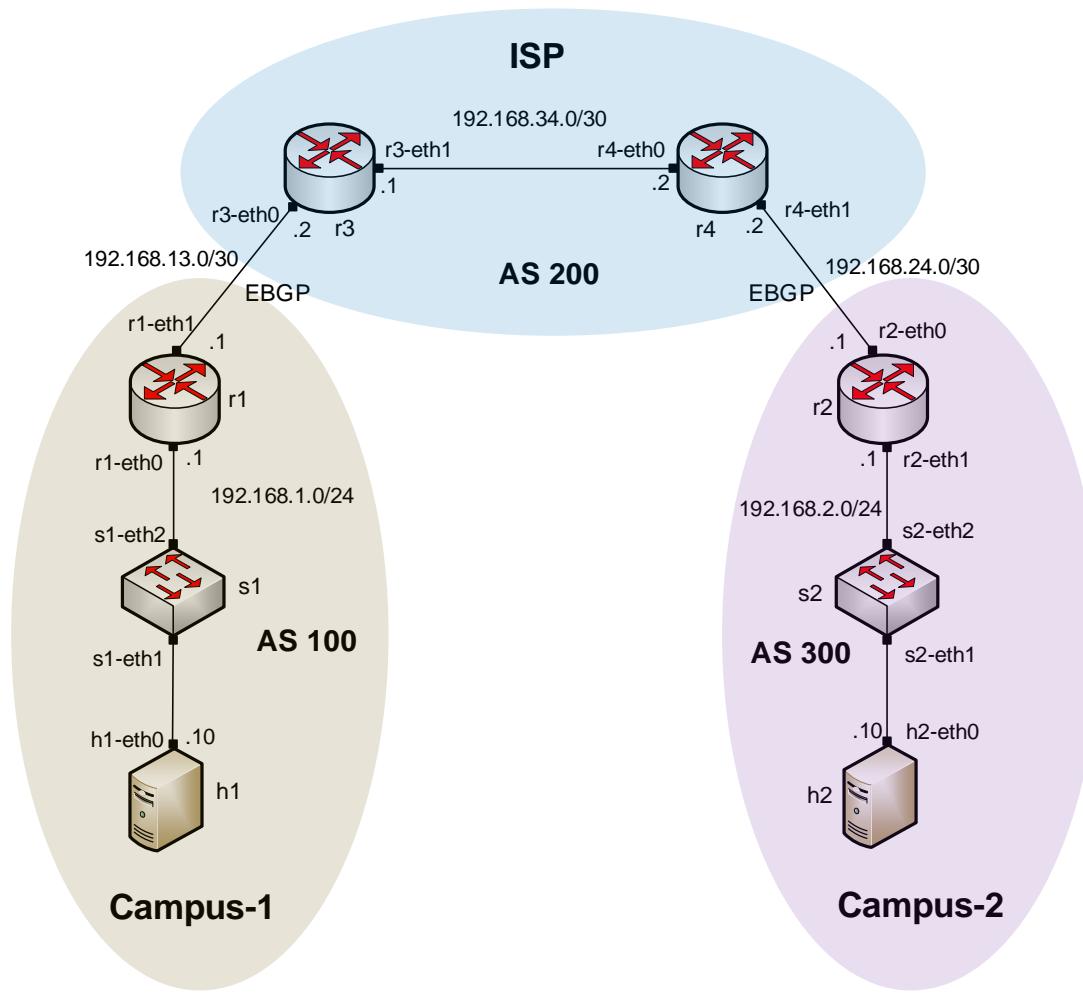
Lab 13 Topology

- Campus-1 hijacks Campus-2's prefixes



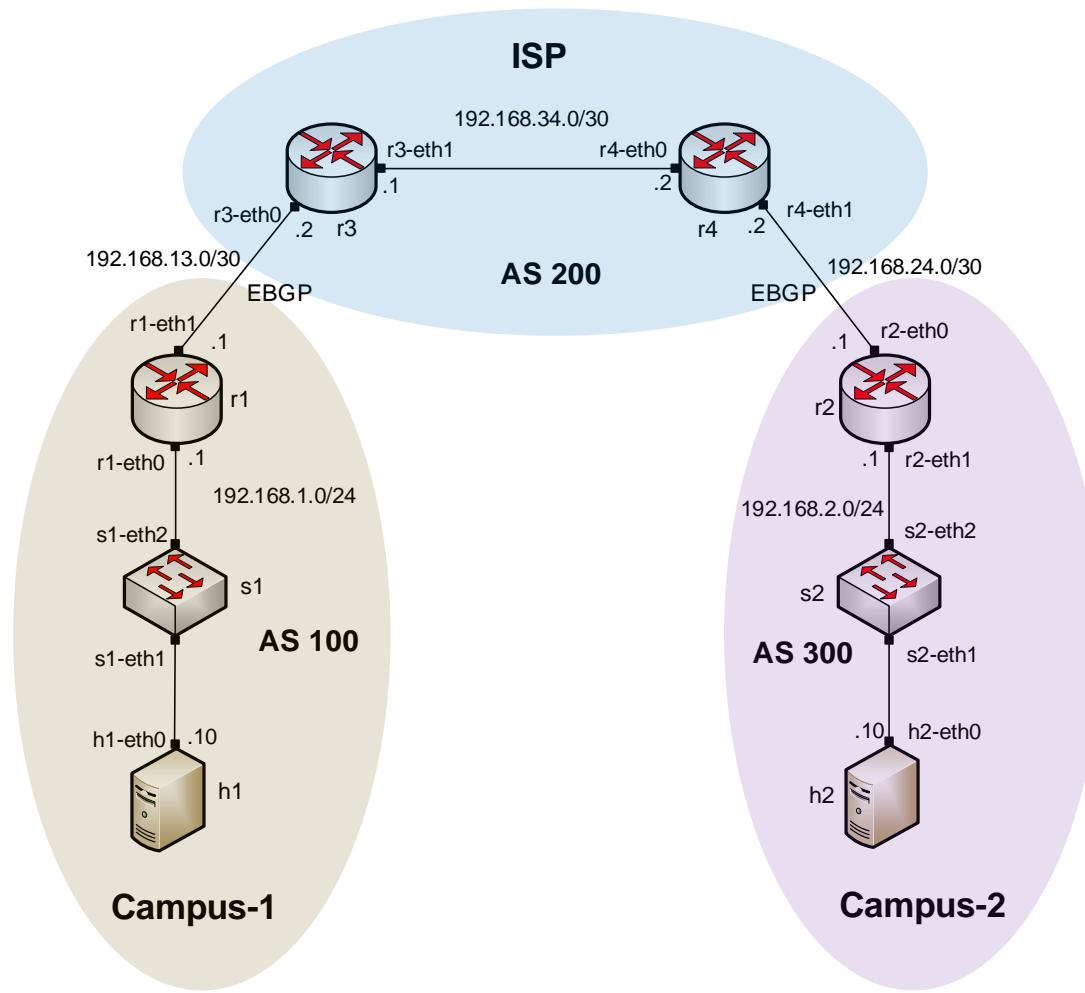
Lab 13 Topology

- Traffic to Campus-2 is redirected to Campus-1



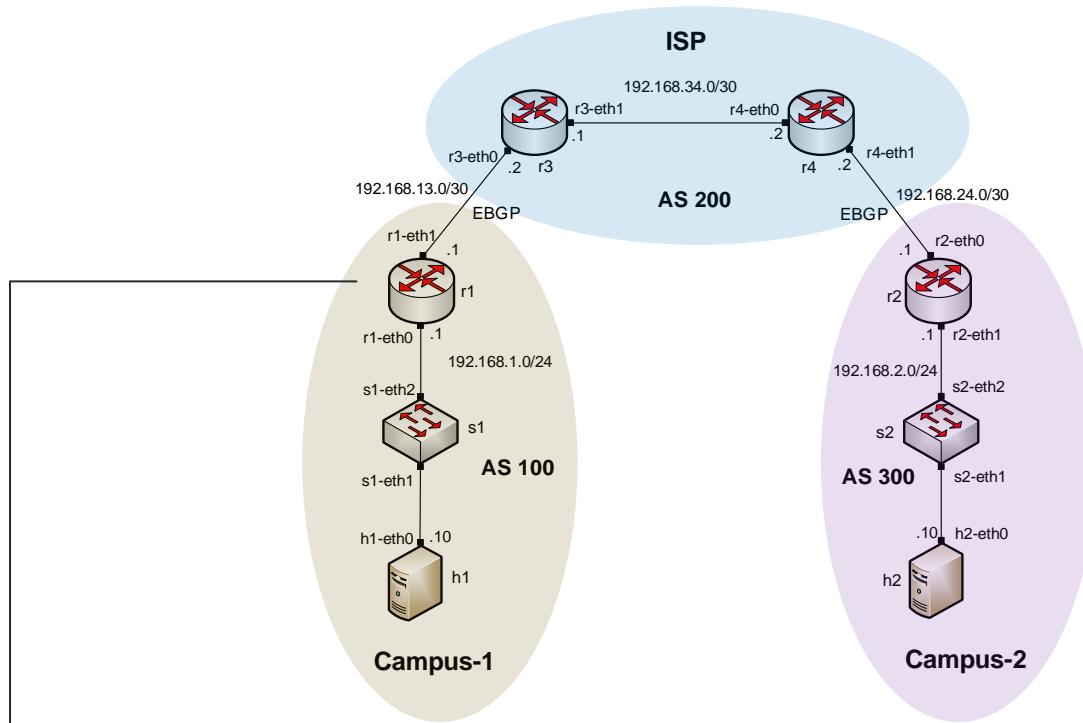
Lab 13 Topology

- The ISP configures IP prefix lists on both campuses to prevent BGP hijacking



Lab 13 Configuration

- Router r1 hijacks (advertises) the network 192.168.2.0/24



Router r1

```
frr-pc# configure terminal
frr-pc(config)# router bgp 100
frr-pc(config-router)# network 192.168.2.0/24
```

Lab 13 Configuration

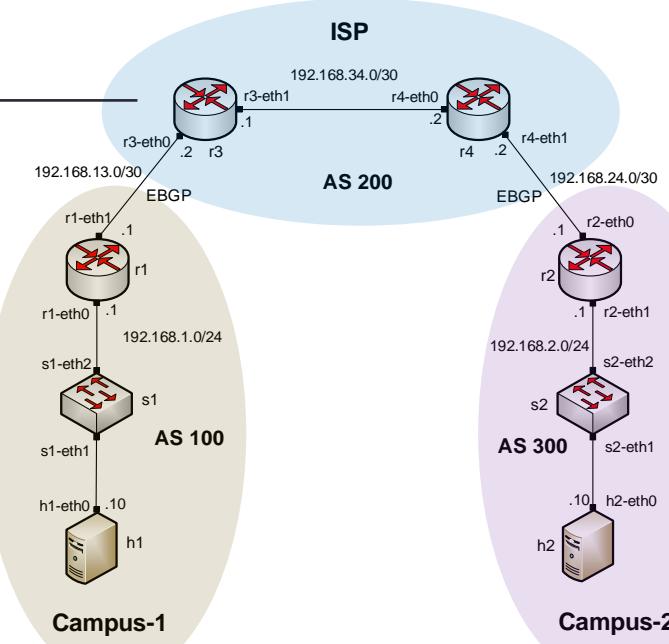
- Router r3 changes the next hop of Campus-2 (192.168.2.0/24)

Router r3

```
"Host: r3"
frr-pc# show ip bgp
BGP table version is 3, local router ID is 192.168.34.1, vrf id 0
Default local pref 100, local AS 200
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

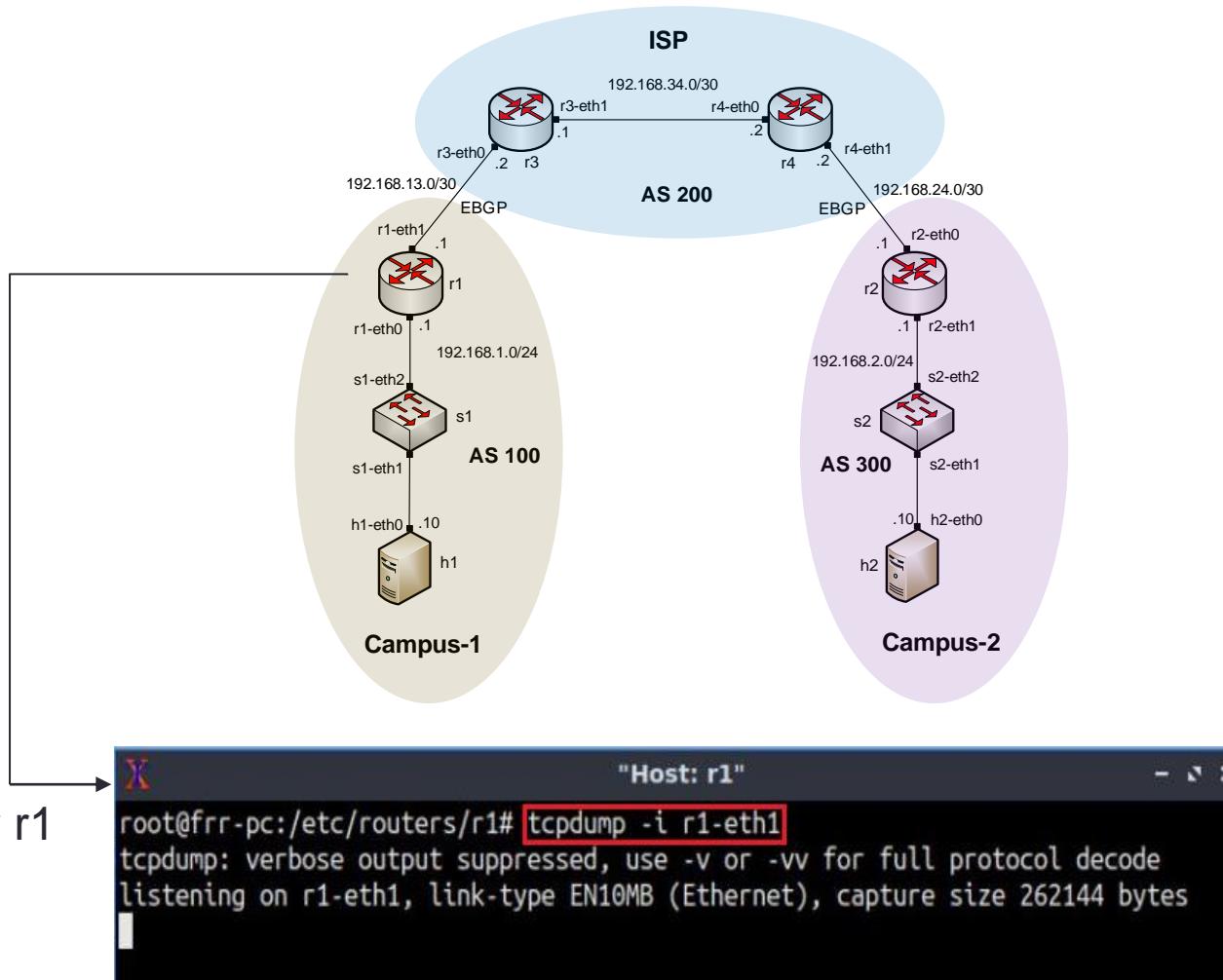
      Network          Next Hop            Metric LocPrf Weight Path
*-> 192.168.1.0/24  192.168.13.1        0        0 100 i
*-> 192.168.2.0/24  192.168.13.1        0        0 100 i
* i                 192.168.34.2       0     100        0 300 i

Displayed 2 routes and 3 total paths
frr-pc#
```



Lab 13 Configuration

- Capture the packets on router r1, specifically at r1-eth1

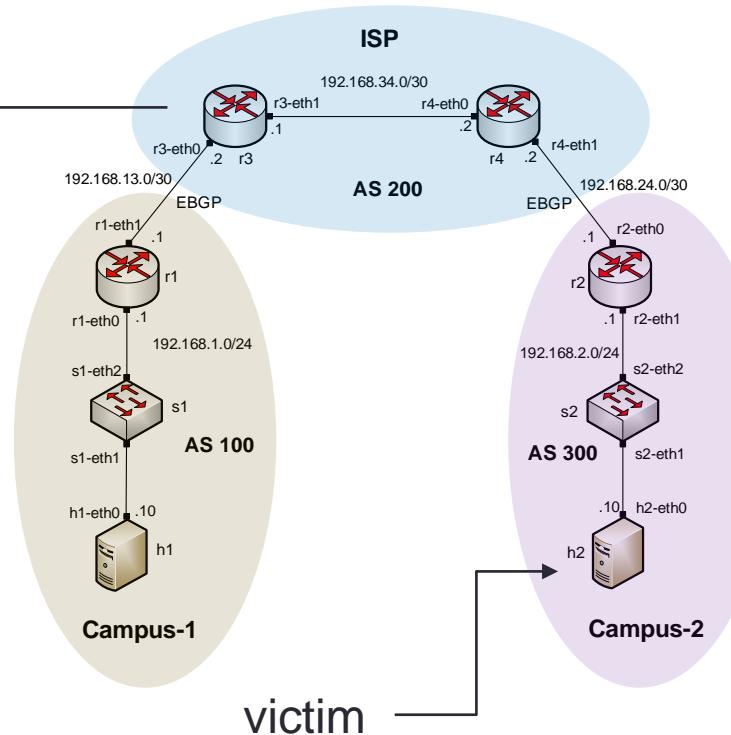


Lab 13 Configuration

- Ping the victim (192.168.2.10) from the ISP (router r3)

Router r3

```
"Host: r3"
frr-pc# ping 192.168.2.10
PING 192.168.2.10 (192.168.2.10) 56(84) bytes of data.
From 192.168.13.1 icmp_seq=1 Destination Net Unreachable
From 192.168.13.1 icmp_seq=2 Destination Net Unreachable
From 192.168.13.1 icmp_seq=3 Destination Net Unreachable
From 192.168.13.1 icmp_seq=4 Destination Net Unreachable
^C
--- 192.168.2.10 ping statistics ---
4 packets transmitted, 0 received, +4 errors, 100% packet loss, time 76ms
frr-pc#
```



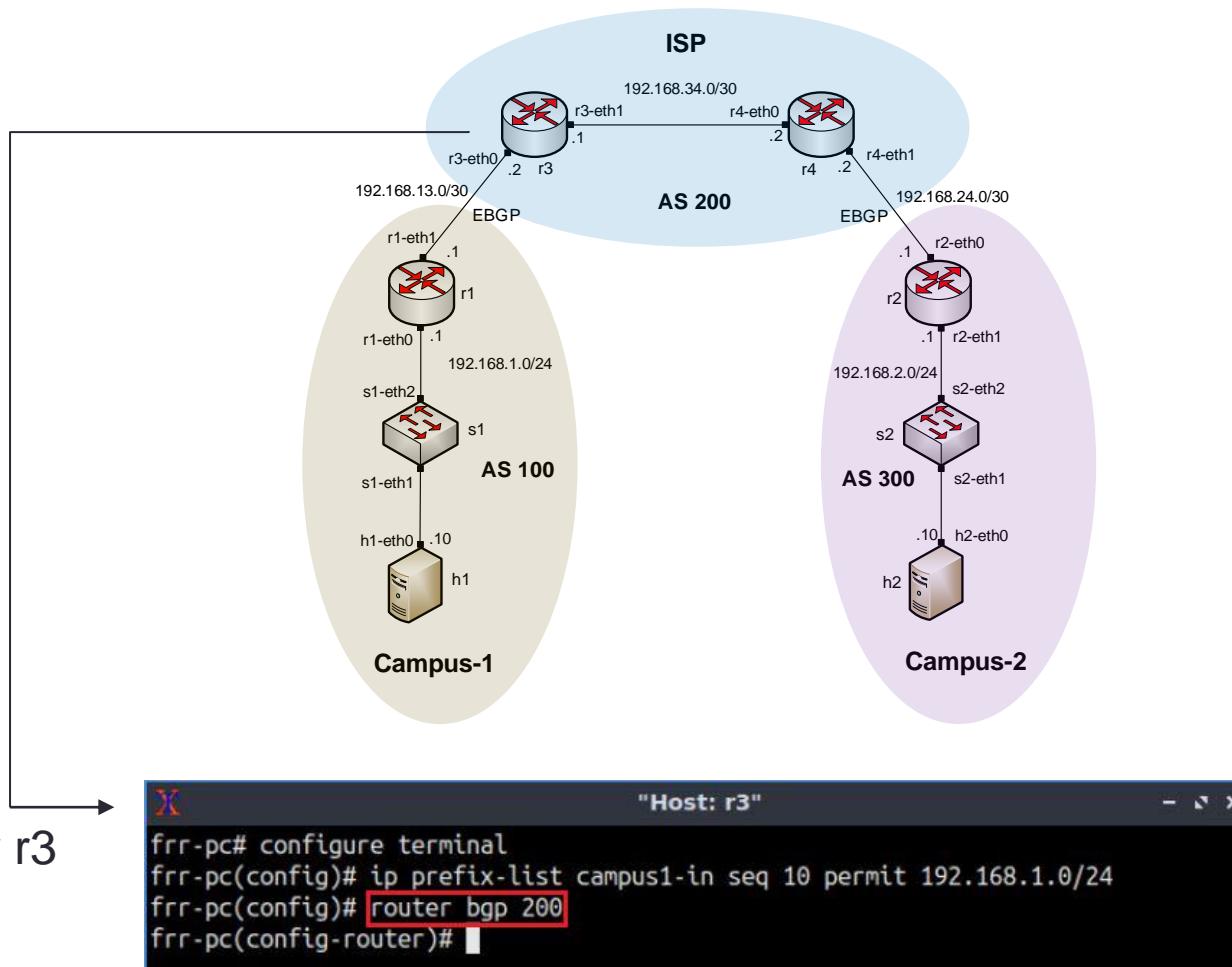
Lab 13 Configuration

- The traffic to network 192.168.2.10 will be rerouted to the hijacking router

```
X "Host: r1"
16:42:56.954553 IP 192.168.13.1 > 192.168.13.2: ICMP net 192.168.2.10 unreachable, length 92
16:42:57.978489 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 3, length 64
16:42:57.978522 IP 192.168.13.1 > 192.168.13.2: ICMP net 192.168.2.10 unreachable, length 92
16:42:59.002492 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 4, length 64
16:42:59.002529 IP 192.168.13.1 > 192.168.13.2: ICMP net 192.168.2.10 unreachable, length 92
16:43:00.026860 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 5, length 64
16:43:01.050427 ARP, Request who-has 192.168.13.1 tell 192.168.13.2, length 28
16:43:01.050605 ARP, Reply 192.168.13.1 is-at da:e0:e3:9f:dd:c9 (oui Unknown), length 28
16:43:01.050577 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 6, length 64
16:43:02.074485 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 7, length 64
16:43:03.098485 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 8, length 64
16:43:04.122487 IP 192.168.13.2 > 192.168.2.10: ICMP echo request, id 2045, seq 9, length 64
```

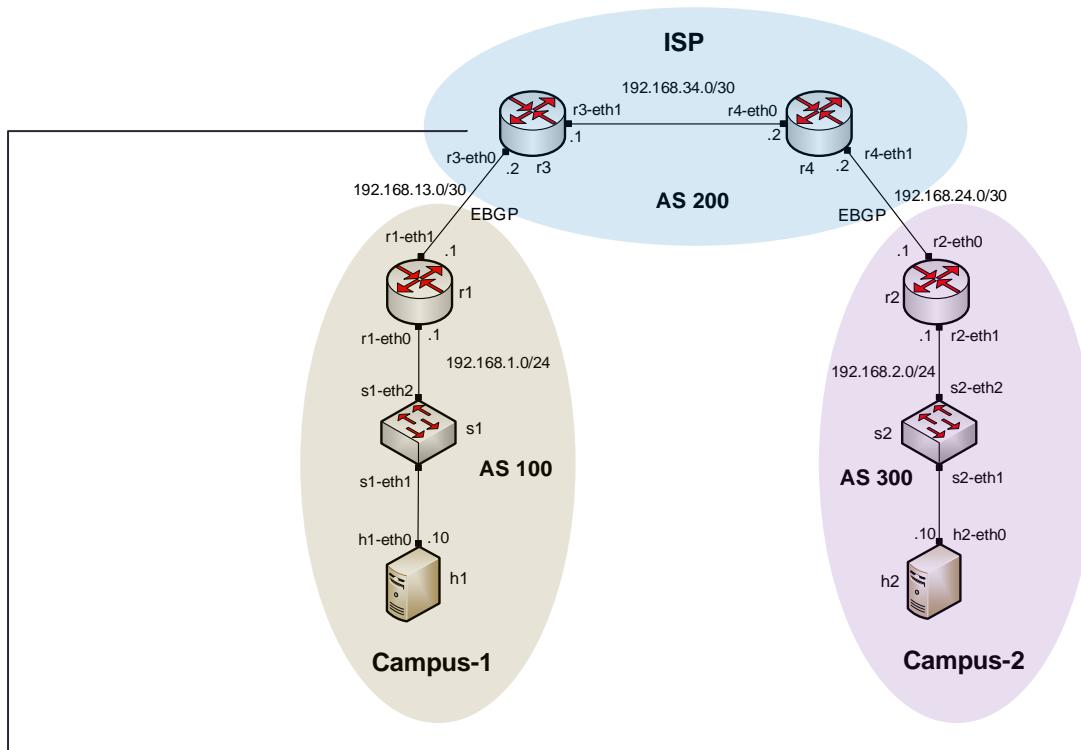
Lab 13 Configuration

- Configure IP prefix list on the ISP (router r3)



Lab 13 Configuration

- Apply the prefix list to router r3 neighbor



Router r3

```
"Host: r3"
frr-pc# configure terminal
frr-pc(config)# ip prefix-list campus1-in seq 10 permit 192.168.1.0/24
frr-pc(config)# router bgp 200
frr-pc(config-router)# neighbor 192.168.13.1 prefix-list campus1-in in
frr-pc(config-router)#

```

Lab 13 Configuration

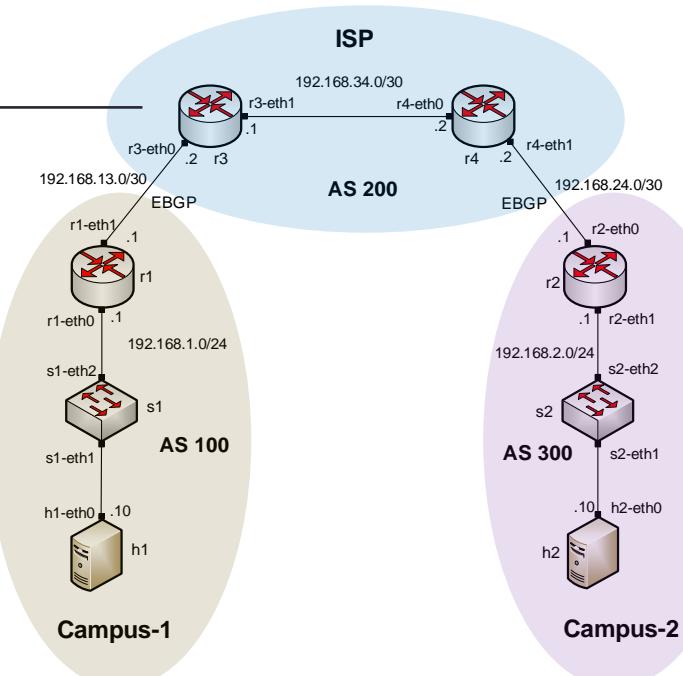
- Router r3 readjusts its BGP table back to normal

Router r3

```
"Host: r3"
frr-pc# show ip bgp
BGP table version is 4, local router ID is 192.168.34.1, vrf id 0
Default local pref 100, local AS 200
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

      Network          Next Hop            Metric LocPrf Weight Path
*-> 192.168.1.0/24  192.168.13.1        0        0 100 i
*>i192.168.2.0/24  192.168.34.2        0       100    0 300 i

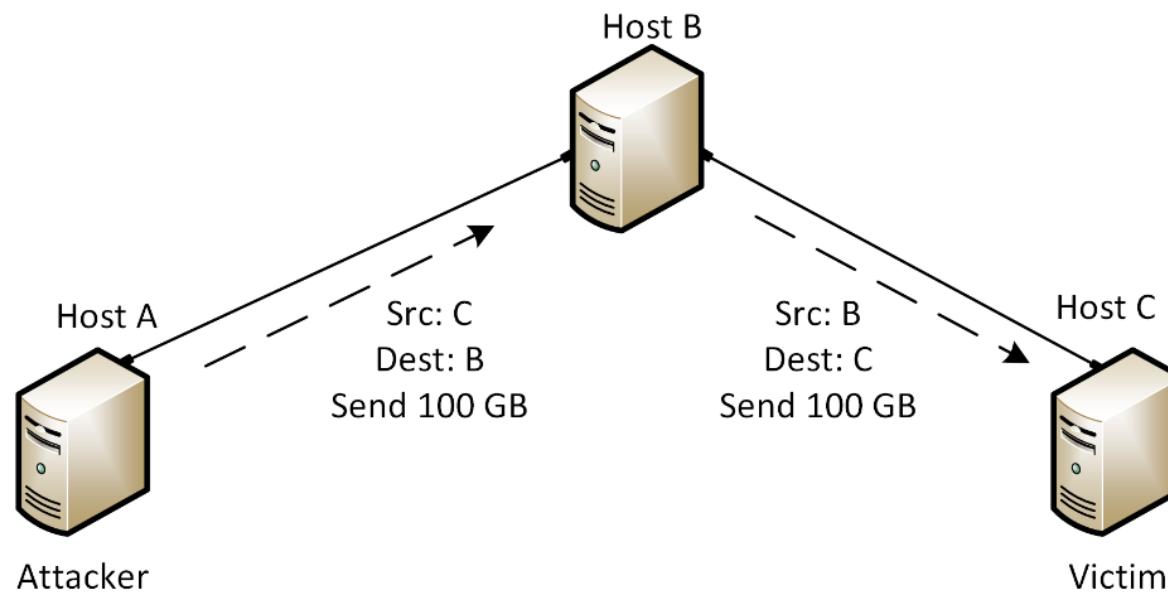
Displayed 2 routes and 2 total paths
frr-pc#
```



LAB 12: IP SPOOFING AND MITIGATION TECHNIQUES

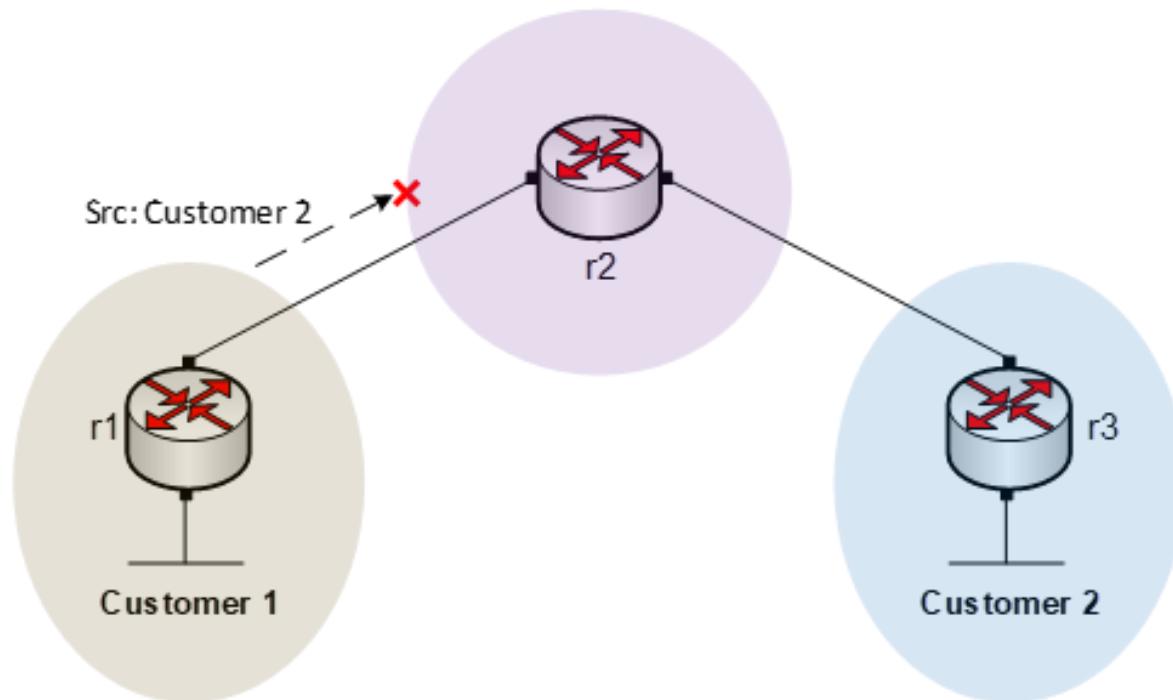
What is IP Spoofing?

- It is the process of originating IP packets with source addresses other than those assigned to the origin host.
- IP spoofing can be exploited in several ways, mainly to launch Denial of Service (DoS) attacks.



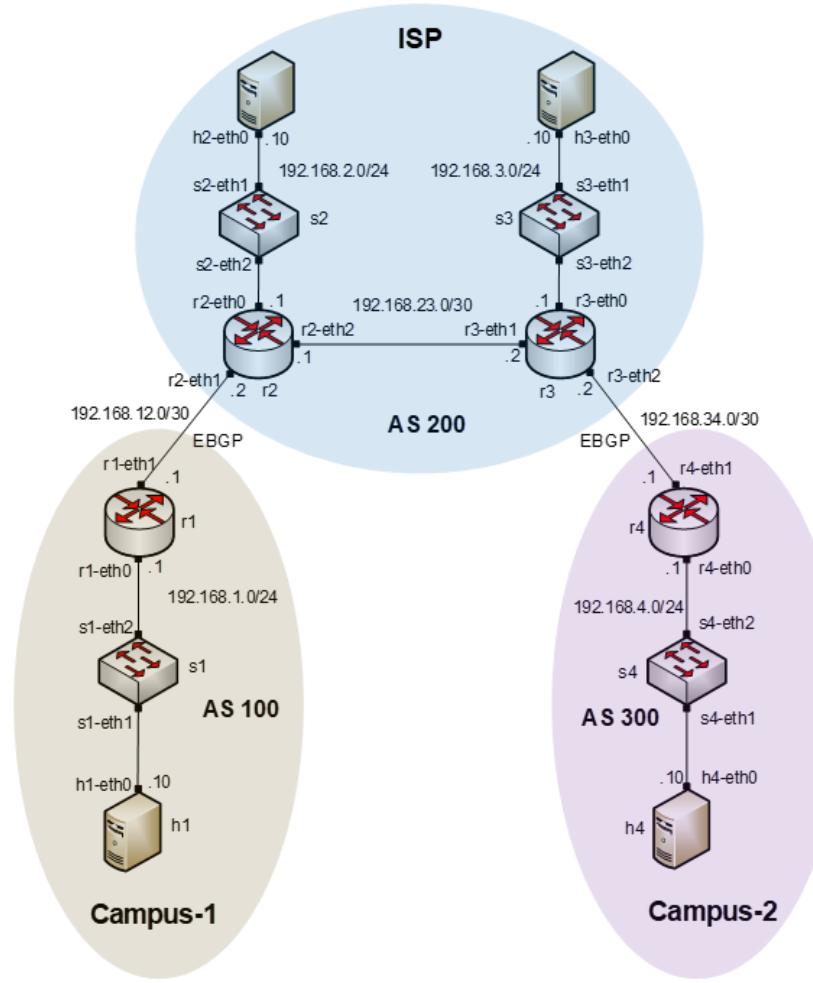
Anti-Spoofing Techniques – Route Filtering

- Route filtering is a method for selectively identifying routes that are advertised or received from neighbor routers.
- It can be used to manipulate traffic flows, reduce memory utilization, or to improve security



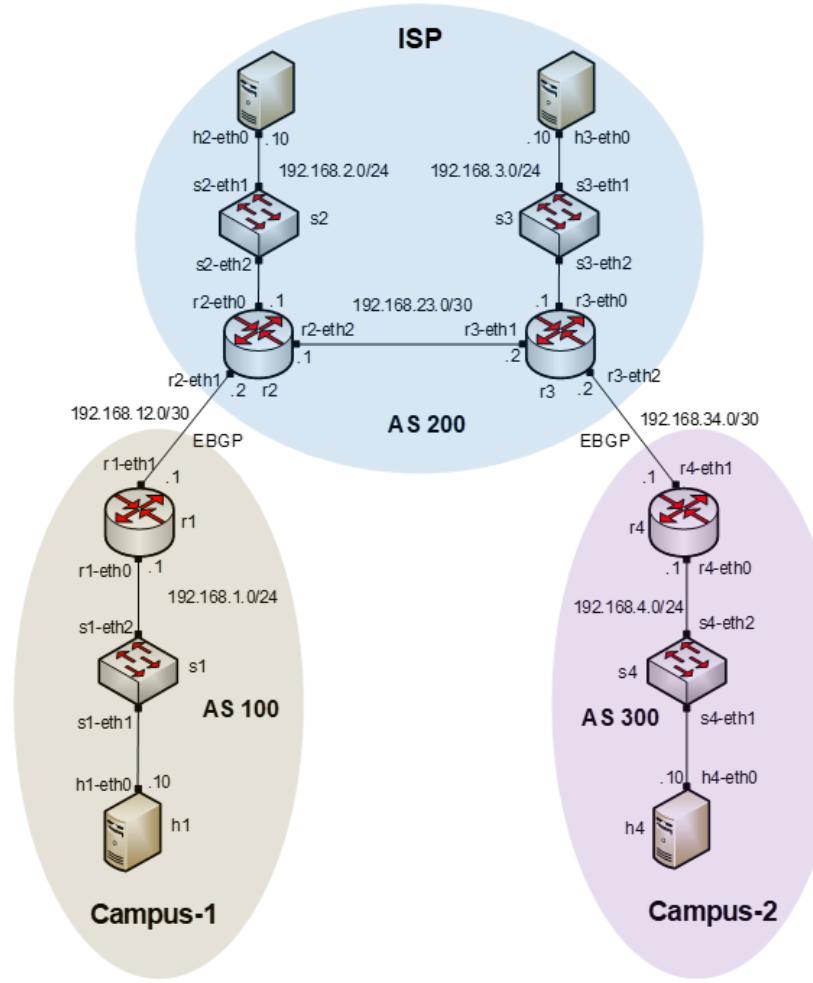
Lab 12 Topology

- Host h1 in Campus-1 spoofs the IP address of host h4 in Campus-2 and launches DoS attack.



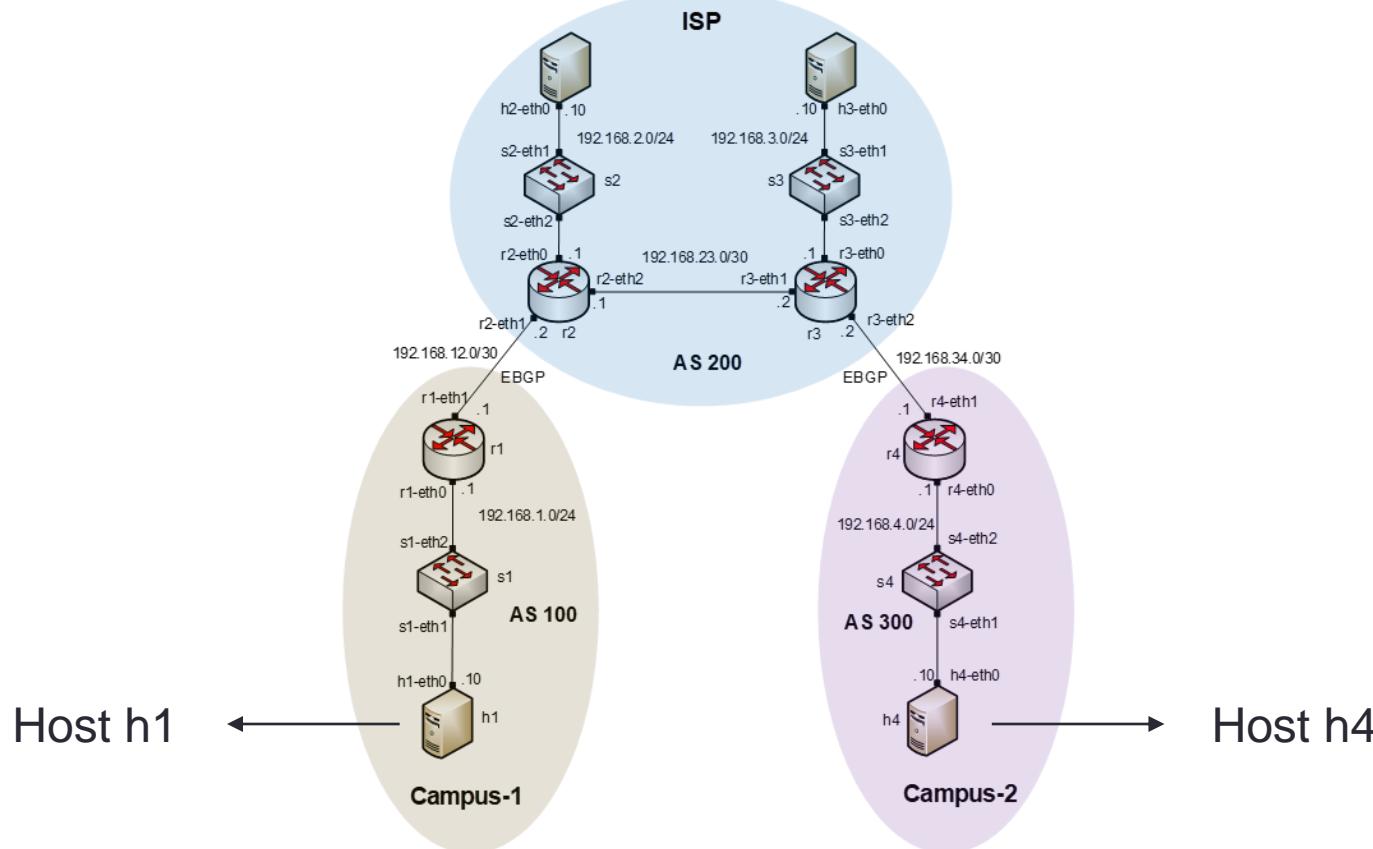
Lab 12 Topology

- The Internet Service Provider (ISP) applies the appropriate route filters to prevent IP spoofing



Lab 12 Configuration

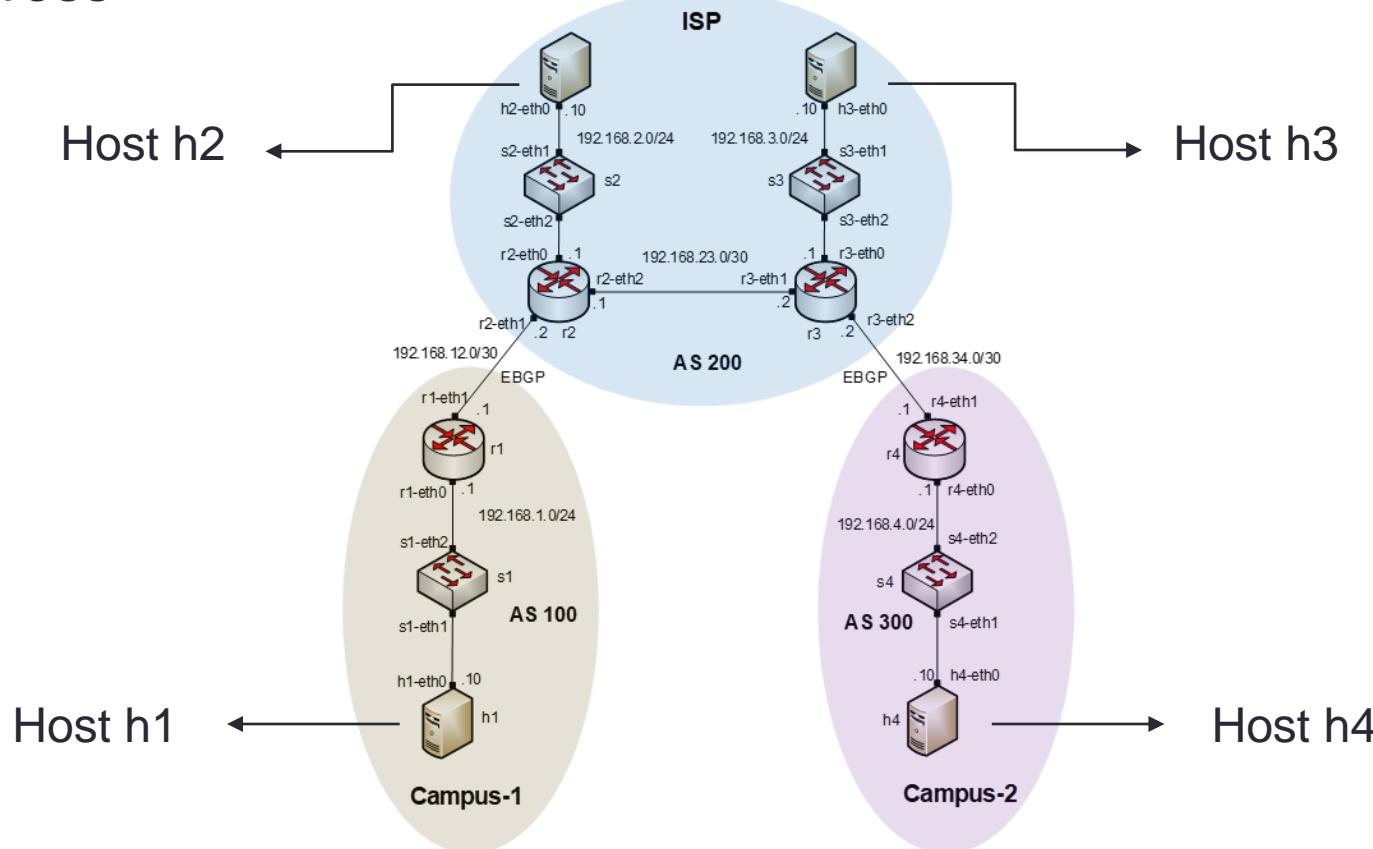
- Host h1 spoofs the IP address of host h4



```
X "Host: h1"
root@frr-pc:~# ifconfig lo 192.168.4.10
root@frr-pc:~# "
```

Lab 12 Configuration

- Host h1 pings hosts h2 and h3 using the spoofed source IP address



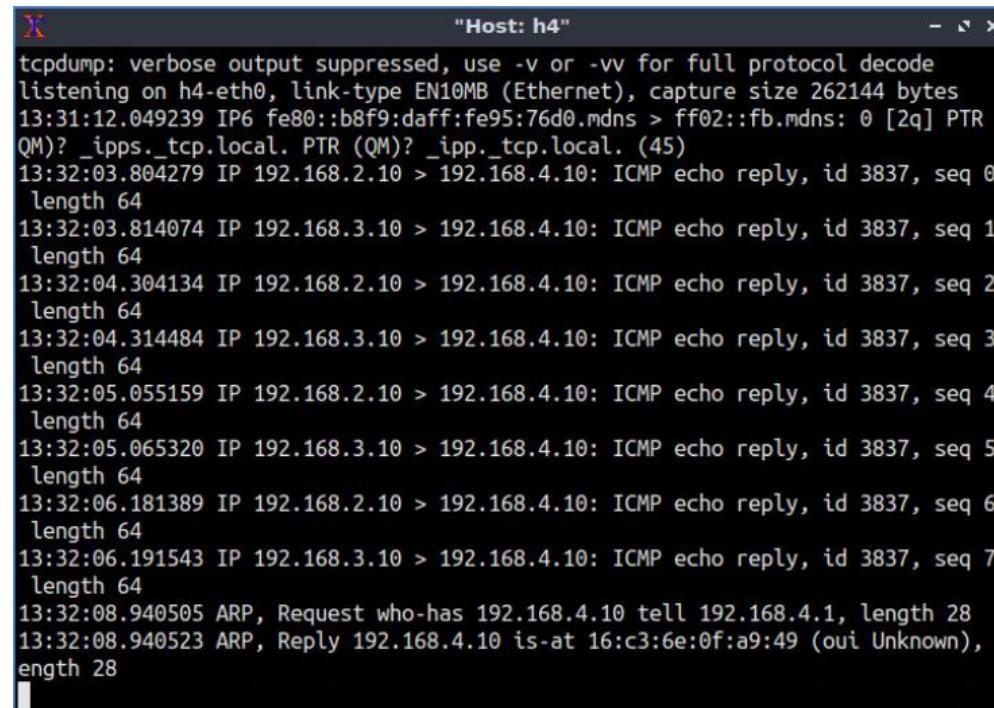
```

root@frr-pc:~# fping --src 192.168.4.10 192.168.2.10 192.168.3.10
192.168.2.10 is unreachable
192.168.3.10 is unreachable
root@frr-pc:~#

```

Lab 12 Configuration

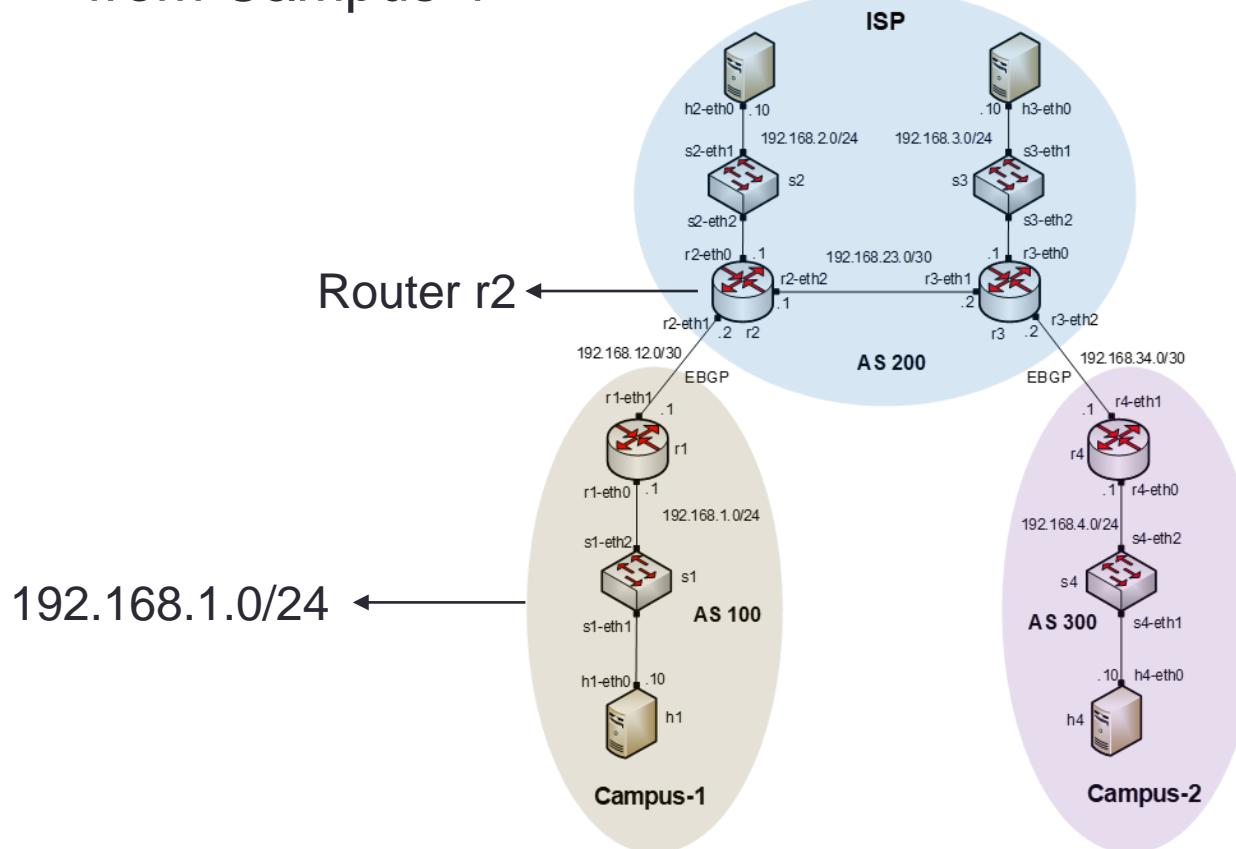
- Host h4 receives a reply messages from 192.168.2.10 and 192.168.3.10



```
"Host: h4"
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h4-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:31:12.049239 IP6 fe80::b8f9:daff:fe95:76d0.mdns > ff02::fb.mdns: 0 [2q] PTR (QM)? _ipps._tcp.local. PTR (QM)? _ipp._tcp.local. (45)
13:32:03.804279 IP 192.168.2.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 0, length 64
13:32:03.814074 IP 192.168.3.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 1, length 64
13:32:04.304134 IP 192.168.2.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 2, length 64
13:32:04.314484 IP 192.168.3.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 3, length 64
13:32:05.055159 IP 192.168.2.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 4, length 64
13:32:05.065320 IP 192.168.3.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 5, length 64
13:32:06.181389 IP 192.168.2.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 6, length 64
13:32:06.191543 IP 192.168.3.10 > 192.168.4.10: ICMP echo reply, id 3837, seq 7, length 64
13:32:08.940505 ARP, Request who-has 192.168.4.10 tell 192.168.4.1, length 28
13:32:08.940523 ARP, Reply 192.168.4.10 is-at 16:c3:6e:0f:a9:49 (oui Unknown), length 28
```

Lab 12 Configuration

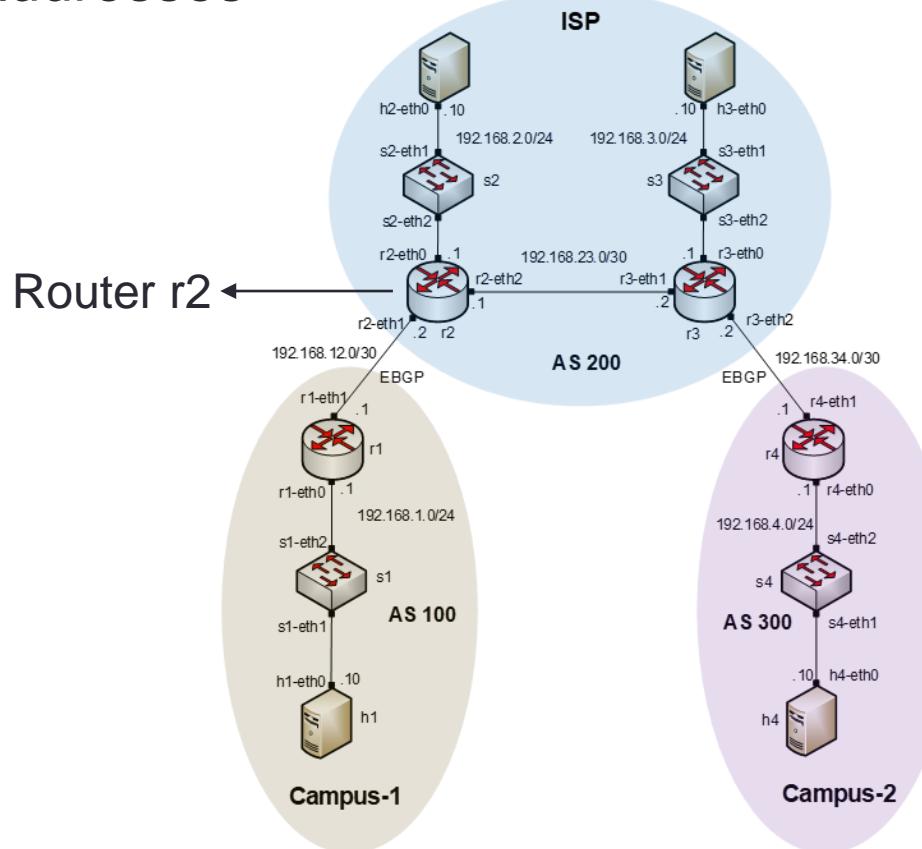
- Apply a filter on router r2 to accept IP source 192.168.1.0/24 from Campus-1



```
"Host: r2"
frr-pc# exit
root@frr-pc:/etc/routers/r2# iptables -A FORWARD -s 192.168.1.0/24 -i r2-eth1 -j ACCEPT
root@frr-pc:/etc/routers/r2# "
```

Lab 12 Configuration

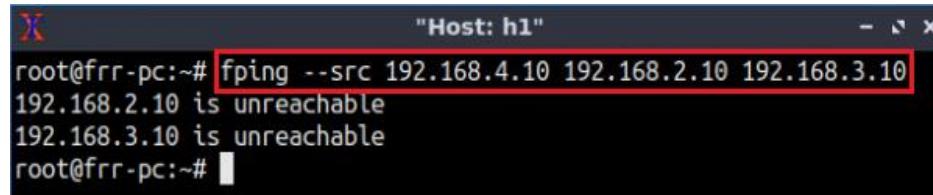
- Apply another route filter on router r2 to reject all other IP source addresses



```
"Host: r2"
frr-pc# exit
root@frr-pc:/etc/routers/r2# iptables -A FORWARD -s 192.168.1.0/24 -i r2-eth1 -j ACCEPT
root@frr-pc:/etc/routers/r2# iptables -A FORWARD -s 0/0 -i r2-eth1 -j DROP
root@frr-pc:/etc/routers/r2#
```

Lab 12 Configuration

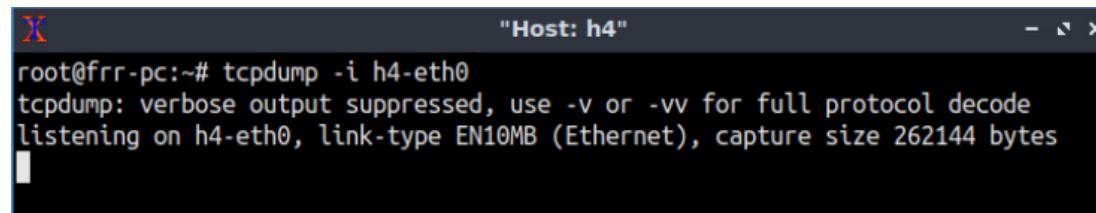
- Perform DoS attack by host h1 on host h4



A terminal window titled "Host: h1". The command entered is "fping --src 192.168.4.10 192.168.2.10 192.168.3.10". The output shows two hosts as unreachable: "192.168.2.10 is unreachable" and "192.168.3.10 is unreachable".

```
root@frrr-pc:~# fping --src 192.168.4.10 192.168.2.10 192.168.3.10
192.168.2.10 is unreachable
192.168.3.10 is unreachable
root@frrr-pc:~#
```

- Capture the network traffic on host h4



A terminal window titled "Host: h4". The command entered is "tcpdump -i h4-eth0". The output shows the command was run successfully and provides information about the interface: "tcpdump: verbose output suppressed, use -v or -vv for full protocol decode", "listening on h4-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes".

```
root@frrr-pc:~# tcpdump -i h4-eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h4-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes

```