

HW 5: Mininet Layer-3 Network

Brian Tokumoto

Task 1:

LAN	Needed Hosts	Prefix	Subnet mask	Net Addr	Smallest usable	Largest usable
A	>= 50	/26	255.255.255.192	20.10.172.128	20.10.172.129	20.10.172.190
B	>= 75	/25	255.255.255.128	20.10.172.0	20.10.172.1	20.10.172.126
C	>= 20	/27	255.255.255.224	20.10.172.192	20.10.172.193	20.10.172.222

Task 2:

- **Routers:** rA, rB, rC
- **Switches:** s1, s2, s3 for each LAN, plus s4 as backbone mesh switch
- **Hosts:**
 - LAN A: hA1 (20.10.172.129/26), hA2 (20.10.172.130/26), gw=20.10.172.129
 - LAN B: hB1 (20.10.172.1/25), hB2 (20.10.172.2/25), gw=20.10.172.1
 - LAN C: hC1 (20.10.172.193/27), hC2 (20.10.172.194/27), gw=20.10.172.193
- **Backbone interfaces** (on eth2):
 - rA: 20.10.100.1/24
 - rB: 20.10.100.2/24
 - rC: 20.10.100.3/24

The full Python script is [layer3_network_code.py](#).

After `net.start()`, we validated each host pair on the same LAN:

```
mininet> pingall
```

```
*** Ping: testing ping reachability
```

```
hA1 -> hA2 OK
```

```
hB1 -> hB2 OK
```

```
hC1 -> hC2 OK
```

```
mininet@mininet-vm:~$ rm layer3_network_code.py
mininet@mininet-vm:~$ sudo python3 layer3_network_code.py
**** Creating network
**** Adding routers
**** Adding hosts
**** Adding switches
**** Creating links
**** Building network
**** Configuring hosts
rA rB rC hA1 hA2 hB1 hB2 hC1 hC2
**** Starting controller

**** Starting 4 switches
s1 s2 s3 s4 ...

=== Testing intra-LAN connectivity only ===
hA1 -> hA2
hA2 -> hA1
**** Results: 0% dropped (2/2 received)
hB1 -> hB2
hB2 -> hB1
**** Results: 0% dropped (2/2 received)
hC1 -> hC2
hC2 -> hC1
**** Results: 0% dropped (2/2 received)
**** Starting CLI:
mininet> _
```

Task 3:

On each host we added two routes so that traffic to the other two LANs is sent to the local router:

On hA1/hA2:

```
route add -net 20.10.172.0 netmask 255.255.255.128 gw 20.10.172.129
```

```
route add -net 20.10.172.192 netmask 255.255.255.224 gw 20.10.172.129
```

On hB1/hB2:

```
route add -net 20.10.172.128 netmask 255.255.255.192 gw 20.10.172.1
```

```
route add -net 20.10.172.192 netmask 255.255.255.224 gw 20.10.172.1
```

On hC1/hC2:

```
route add -net 20.10.172.0 netmask 255.255.255.128 gw 20.10.172.193
```

```
route add -net 20.10.172.128 netmask 255.255.255.192 gw 20.10.172.193
```

After routes were in place, we tested:

```
mininet>
mininet> hA1 ping -c3 20.10.172.2
PING 20.10.172.2 (20.10.172.2) 56(84) bytes of data.
64 bytes from 20.10.172.2: icmp_seq=1 ttl=62 time=0.905 ms
64 bytes from 20.10.172.2: icmp_seq=2 ttl=62 time=0.099 ms
64 bytes from 20.10.172.2: icmp_seq=3 ttl=62 time=0.094 ms

--- 20.10.172.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2029ms
rtt min/avg/max/mdev = 0.094/0.366/0.905/0.381 ms
mininet> hA1 tracepath 20.10.172.2
 1?: [LOCALHOST] pmtu 1500
 1: ??? 0.059ms
 1: ??? 0.016ms
 2: ??? 0.023ms
 3: ??? 0.025ms reached
    Resume: pmtu 1500 hops 3 back 3
mininet> _

mininet> hC2 ping -c3 20.10.172.131
PING 20.10.172.131 (20.10.172.131) 56(84) bytes of data.
64 bytes from 20.10.172.131: icmp_seq=1 ttl=62 time=0.983 ms
64 bytes from 20.10.172.131: icmp_seq=2 ttl=62 time=0.210 ms
64 bytes from 20.10.172.131: icmp_seq=3 ttl=62 time=0.127 ms

--- 20.10.172.131 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2017ms
rtt min/avg/max/mdev = 0.127/0.440/0.983/0.385 ms
mininet> hC2 tracepath 20.10.172.131
 1?: [LOCALHOST] pmtu 1500
 1: ??? 1.720ms
 1: ??? 0.028ms
 2: ??? 1.411ms asymm 3
 3: ??? 0.029ms asymm 2
 4: ??? 0.826ms reached
    Resume: pmtu 1500 hops 4 back 3
mininet> _
```

Tests succeeded with 0% loss and correct (ish) three-hop paths.