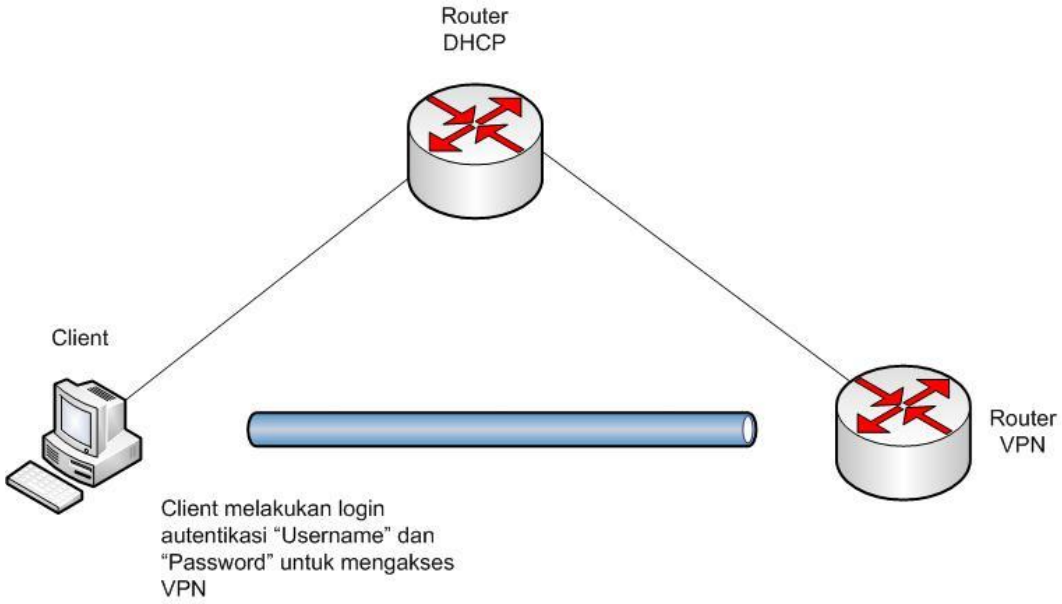
BAB IV
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Tahap implementasi merupakan tahap penerapan dari analisis yang sudah dilakukan. Penerapan metode yang digunakan pada penelitian, selanjutnya akan di implementasi pada setiap perangkat yang terlibat untuk menghasilkan sistem yang sesuai dengan tahap perencangan.

4.1.1 Implentasi VPN Server

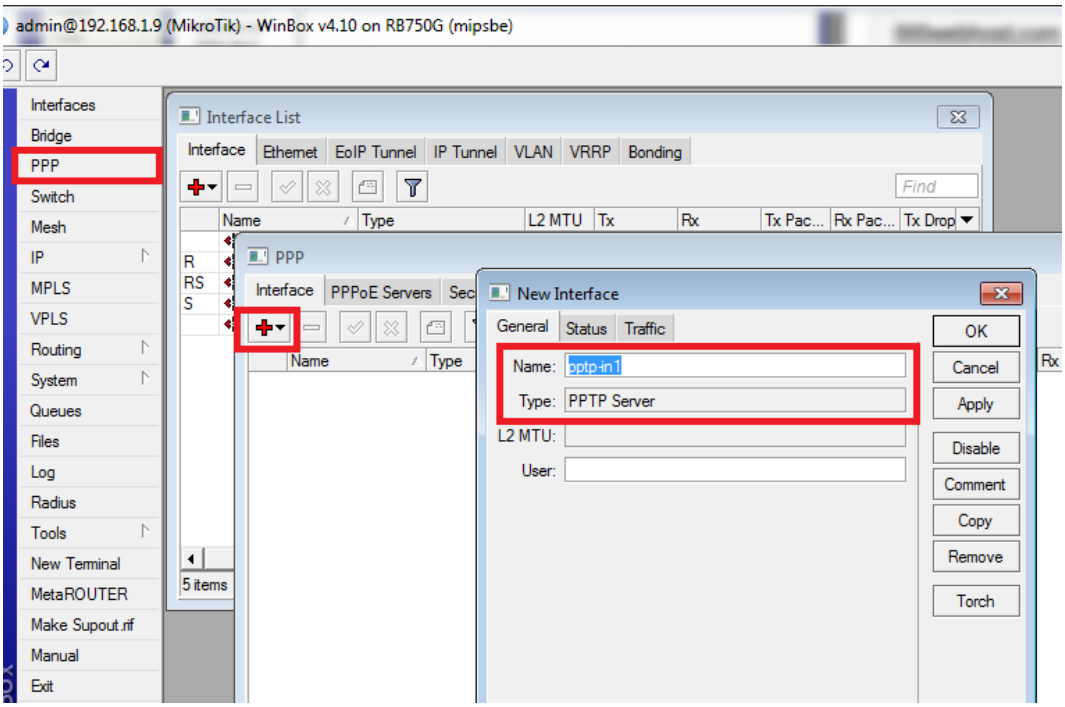
Pada pengimplementasian *VPN Server* menggunakan *mikrotik router OS*. VPN Server ini diimplementasikan dengan topologi jaringan pada gambar 4.1. Pada topologi tersebut setelah client mendapat ip dari router *DHCP*, client dapat melakukan koneksi terhadap VPN server dengan melakukan autentikasi *loginusername* dan *password*terlebih dahulu.



Gambar 4. 1 Topologi Jaringan VPN

4.1.1.1 Instalasi *VPN Server*

Instalasi *VPN Server* pada PT.Padjadjaran Mitra menggunakan protokol *PPTP* yang dimulai dengan mengaktikan service yang terdapat pada Mikrotik Router OS tersebut. Langkah-langkah untuk mengaktifkan service *VPN*adalah sebagai berikut :



Gambar 4. 2 Interface pptp

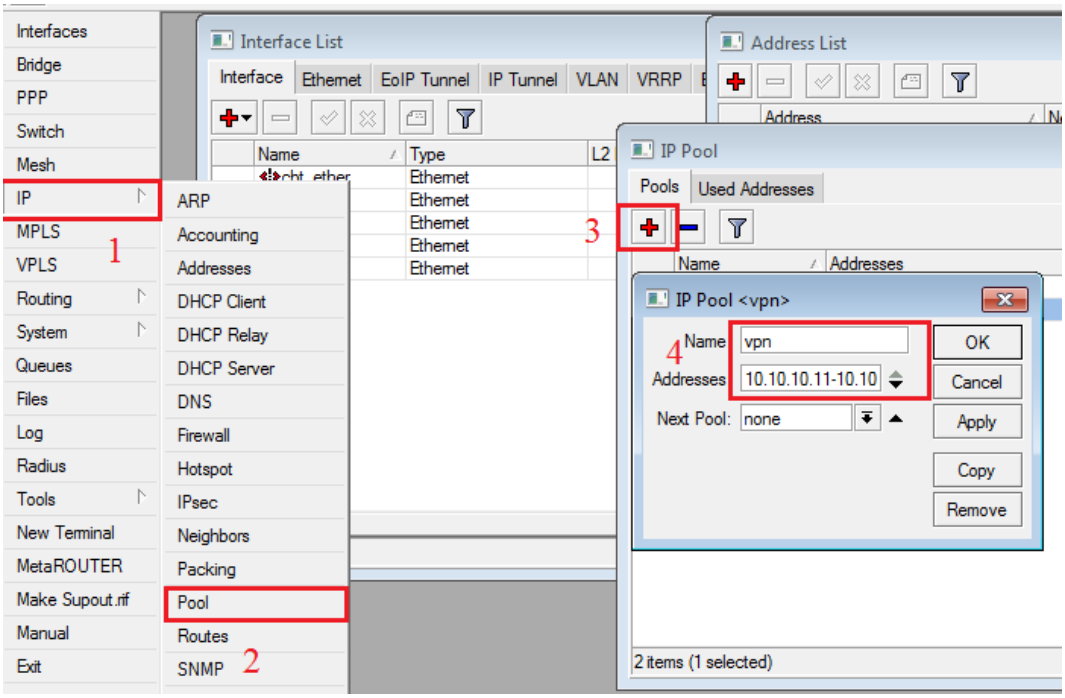
Tahap pertama, dengan membuat terlebih dahulu interface pptp dengan nama interace yang dapat dibuat sesuai keinginan admin.

1. Pilih menu PPP pada tampilan winbox sebelah kiri.
2. Tambahkan interace baru yang akan dibuat.
3. Beri nama PPTP interface yang telah dibuat sesui ketentuan admin.

4.1.1.2 Pembuatan IP Pool

*IP pool*atau sekelompok *IPAddress* dibuat untuk mengalokasikan sejumlah *ip addressVPN Client* per-*user* yang akan terkoneksi ke *VPN Server*. Selain menggunakan *IP Pool* , pemberian *ip address* per-user dapat dilakukan satu

persatu secara manual. Tetapi apabila jumlah client sudah cukup banyak maka cara ini lah yang paling efektif dilakukan.



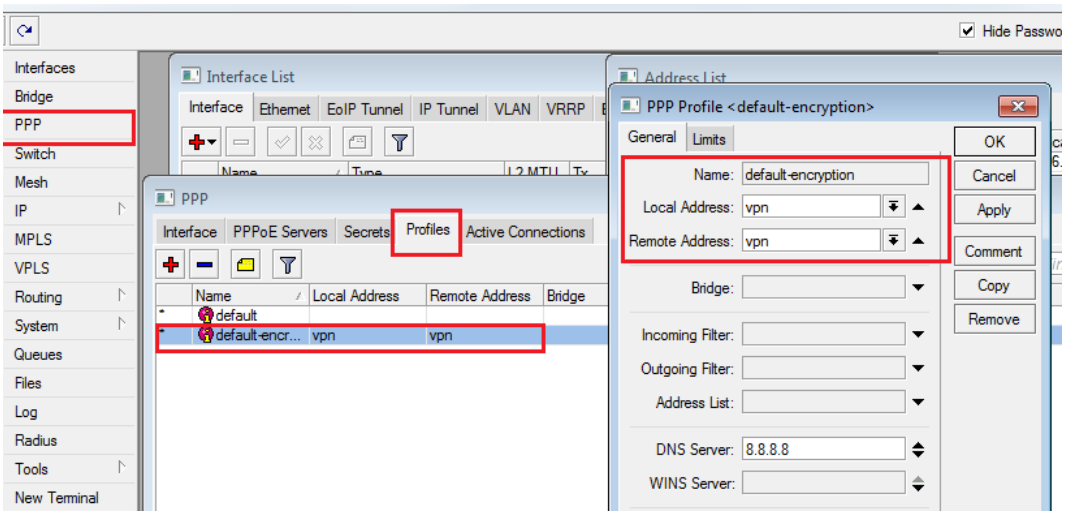
Gambar 4. 3 Ip Pool

Pilih menu IP > Pool untuk menambahkan ip pool yang akan ditentukan.

Buat ip pool baru dengan nama dan *address*yang sudah ditentukan.Misalkan 10.10.10.11-10.10.10.254 dan akan diberikan nama “vpn”.

4.1.1.3 Pembuatan *Profile PPTP-Server*

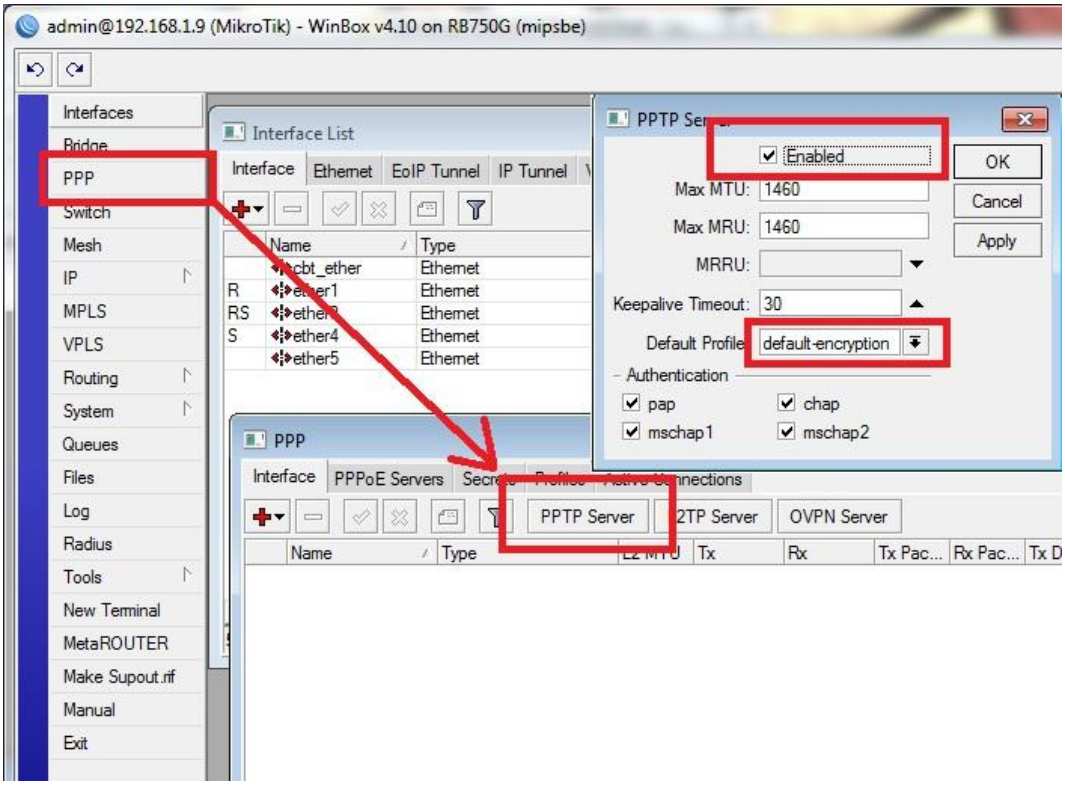
Profile pada *PPTP-Server* dibuat untuk pengaturan alamat *local address* dan *remote address* antara *VPN server* dengan *VPN client.*



Gambar 4. 4 Profile PPTP-Server

1. Klik menu PPP > Tab Profile
2. Buat profile yang diinginkan atau edit profile yang sudah ada dengan meng klik profile yang sudah ada pada tampilan *profile ppp*.
3. Tentukan local addres dan remote address yang akan diberikan dan *DNS* yang akan diberikan.

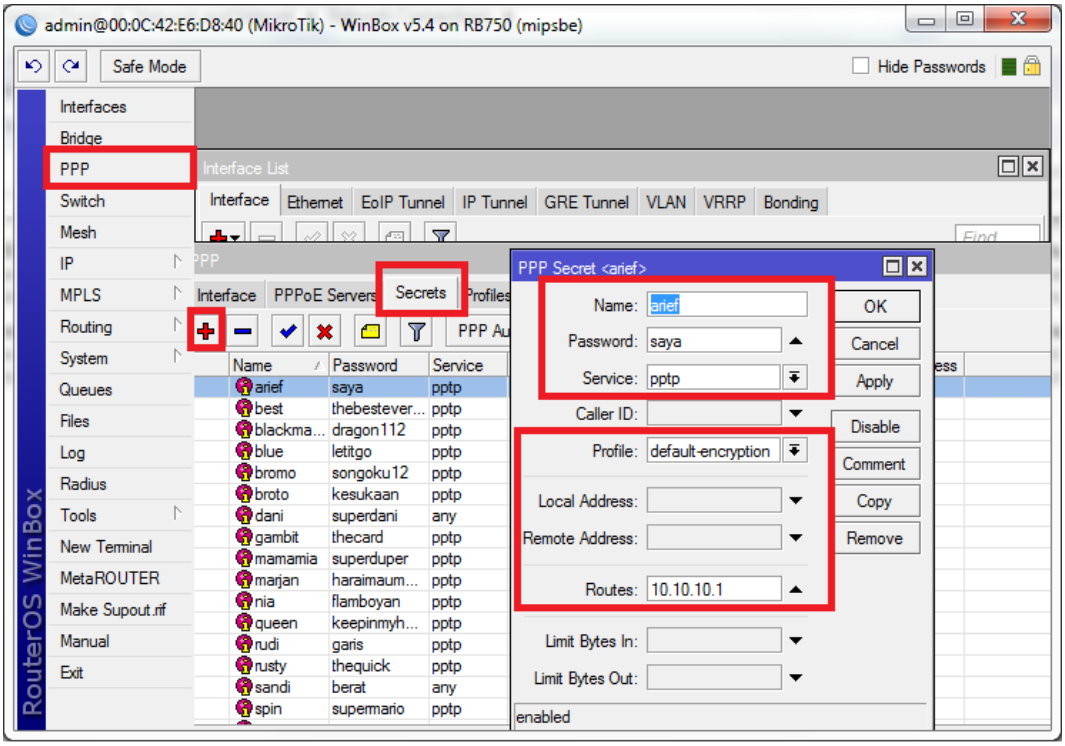
Setelah mengbuat profile *PPTP*, barulah selanjutnya mengaktifkan service PPTP pada mikrotik Router OS nya.



Gambar 4. 5 Service PPTP

4.1.1.4 Pembuatan User VPN

Setelah mengaktifkan service PPTP , selanjutnya membuat user login VPN untuk client yang akan mengakses VPN.

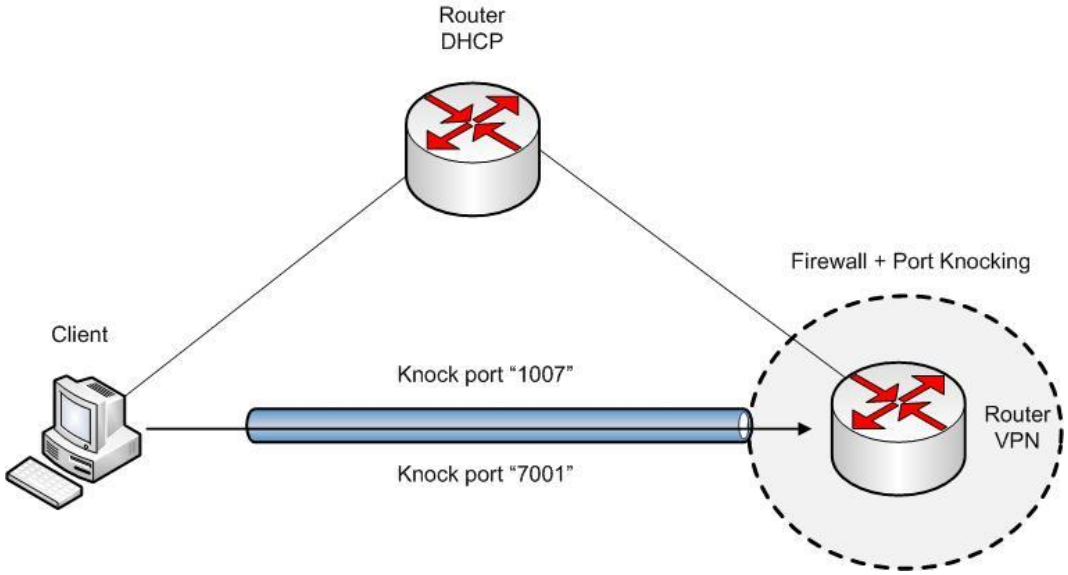


Gambar 4. 6 User *VPN*

1. Klik menu PPP > Tab Secret.
2. Tambahkan User baru.
3. Isikan User name dan Password yang dikehendaki.
4. Pilih service PPTP, dan sesuaikan profile yang dibuat.
5. Isikan Router gateway yang dituju.

4.1.2 Implementasi *Port Knocking*

Port knocking diimplementasikan menggunakan topologi jaringan pada gambar 4.7 ada 2 *port* yang digunakan untuk membuka akses VPN *Server*.Pada topologi tersebut terdapat seorang *client*yang hendak mengakses VPN *Server*dengan melakukan *knocking port*terlebih dahulu sesuai dengan port yang telah ditentukan agar dapat terkoneksi dengan VPN *Server.*



Gambar 4. 7 Topologi Jaringan VPN dengan *Port Knocking*

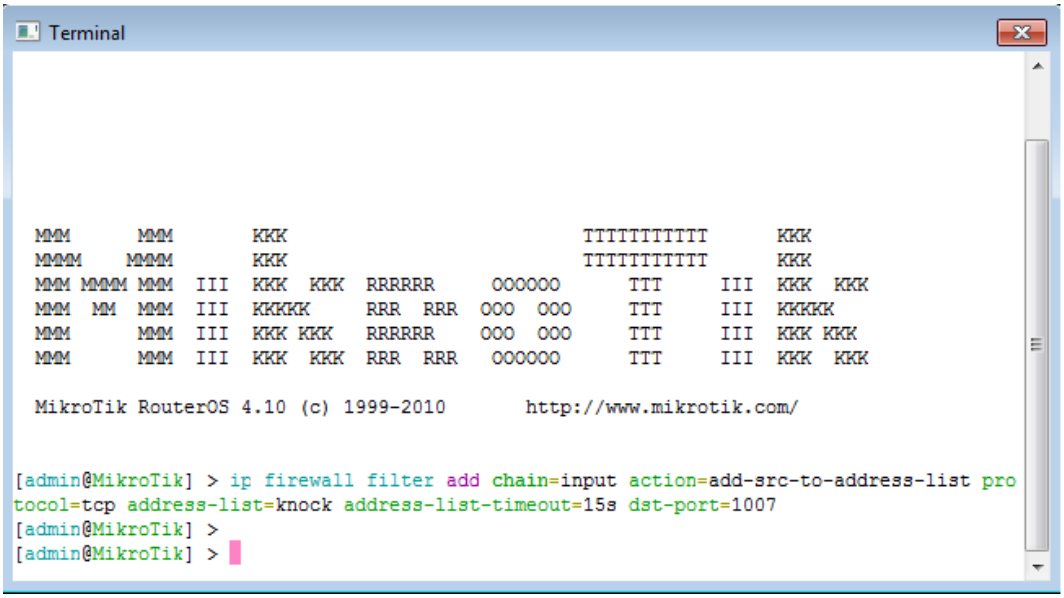
4.1.2.1 Konfigurasi Port Knocking

Selanjutnya tahap konfigurasi *port knocking* pada VPN *Server* untuk mem-*filter* Client yang *Authorize*untuk mengakses VPN *Server*PT.Padjadjaran Mitra. Berikut adalah langkah-langkah dalam konfigurasi *port knocking*pada mikrotik *RouterOS.*

1. Membuat rule pada terminal , dengan perintah

```
ip firewall filter add chain=input action=add-src-to-address-list
protocol=tcp address-list=knock address-list-timeout=15s dst-port=1007
```

Pembuatan rule yang pertama mempunyai arti apabila ada suatu client yang mengetuk *port*1007dengan *protocol tcp* maka ip dari client tersebut akan dimasukan kedalam kelompok *address-list"knock"*selama 15 detik.

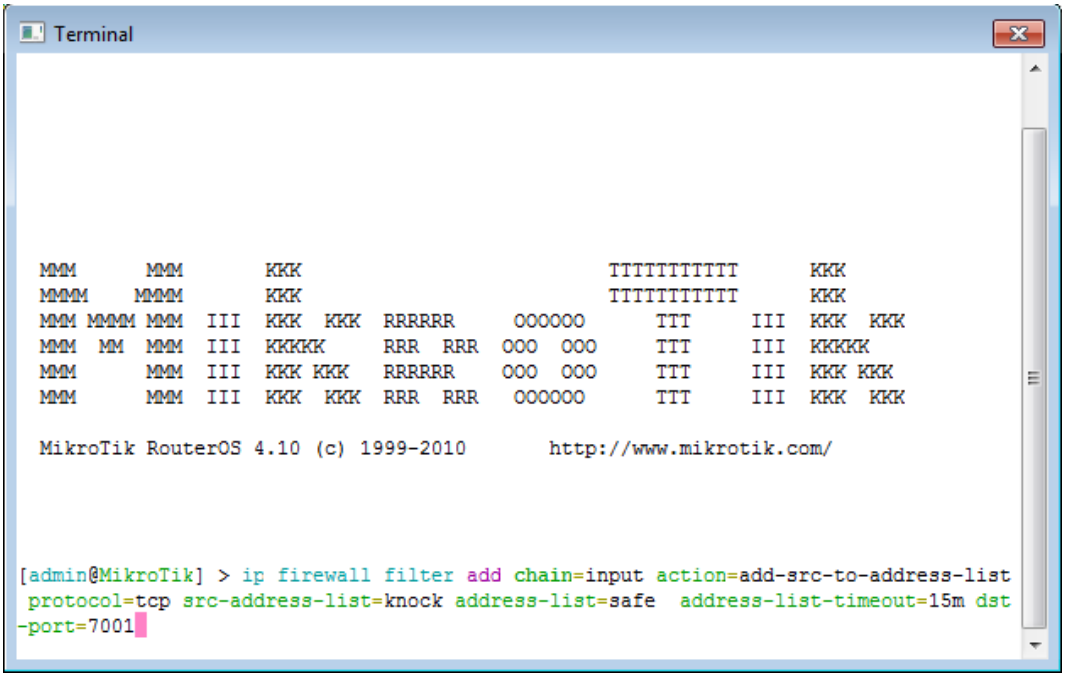


Gambar 4. 8 Rule 1

2. Membuat rule ke 2 pada terminal mikrotik, dengan perintah

ip firewall filter add chain=input action=add-src-to-address-list protocol=tcp src-address-list=knock address-list=safe address-list-timeout=15m dst-port=7001

Pembuatan rule yang ke dua mempunya arti apabila suatu ip client yang berada dalam kelompok *address-list* "*knock*" yang mengetuk *port* 7001 dengan protocol *tcp* , maka ip client tersebut akan dimasukan kedalam kelompok *address-list* "*safe*" selama 15 menit. Apabila dalam 15 menit ip tersebut tidak melakukan koneksi terhadap VPN *Server*, maka ip client yang berada dalam kelompok *address-list "safe"* akan dihapus.

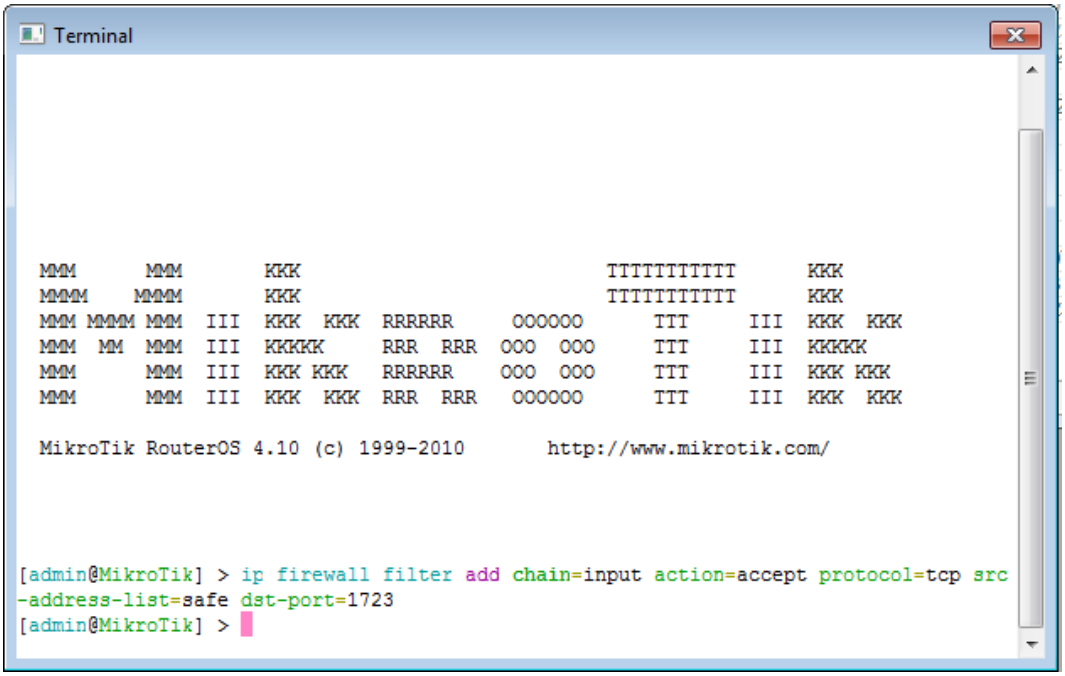


Gambar 4. 9 Rule 2

2. Membuat rule 3 pada terminal mikrotik, dengan perintah

ip firewall filter add chain=input action=accept protocol=tcp src-address-list=safe dst-port=1723

Pembuatan rule yang ke 3 mempunya arti apabila suatu ip client yang berada dalam kelompok *address-list"safe"*yang mengakses *port* 1723 maka akan diijinkan untuk mengakses VPN *Server*.

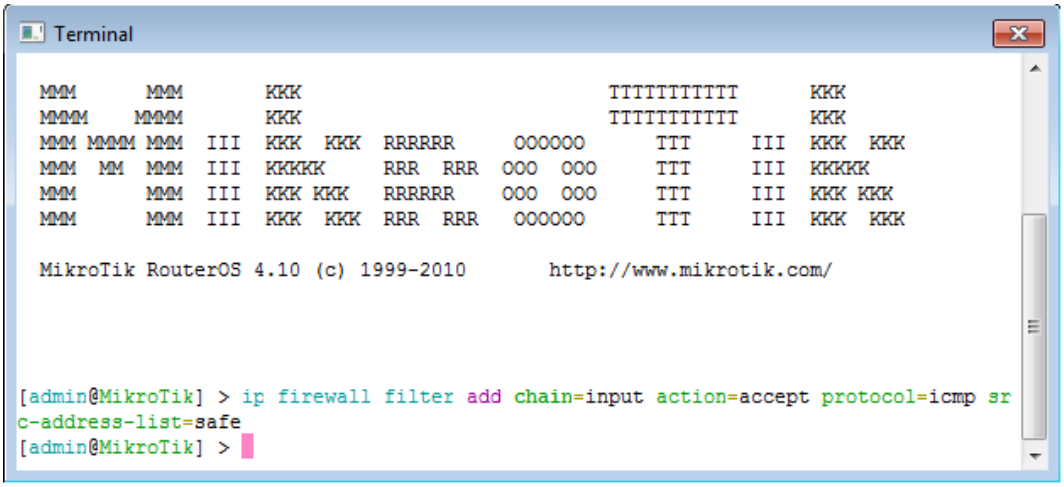


Gambar 4. 10 Rule 3

3. Membuat rule 4 pada terminal mikrotik, dengan perintah

ip firewall filter add chain=input action=accept protocol=icmp src-address-list=safe

Pada pembuatan rule yang ke 5 mempunya arti apabila suatu ip client yang berada dalam kelompok *address-list "safe"*diijinkan untuk mengakses *protocol icmp.*

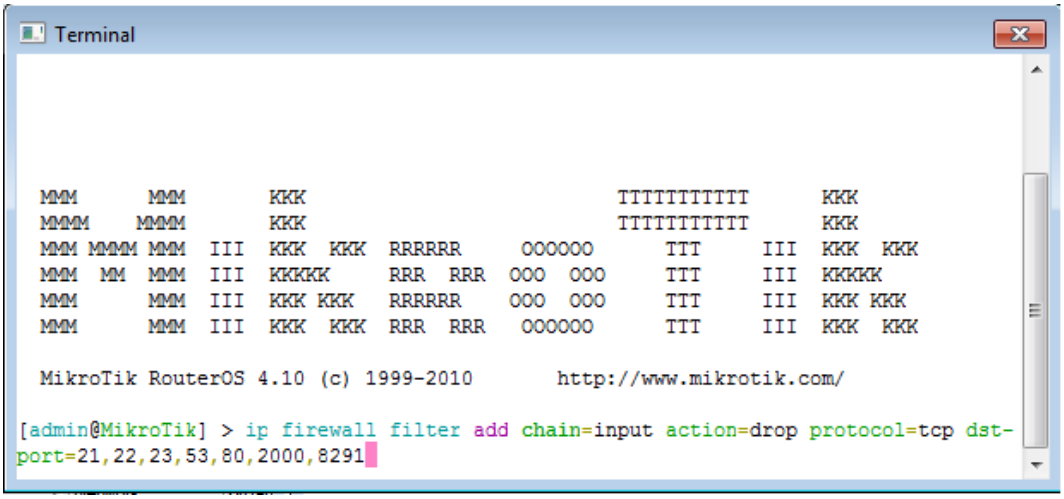


Gambar 4. 11 Rule 4

4. Membuat rule 6 pada terminal mikrotik, dengan perintah

ip firewall filter add chain=input action=drop protocol=tcp dst-port=21,22,23,53,80,2000,8291

Pada pembuatan rule yang ke 6 mempunya arti apabila suatu ip *client* yang mencoba mengakses port 21,22,23,53,80,2000,8291 akan di drop.

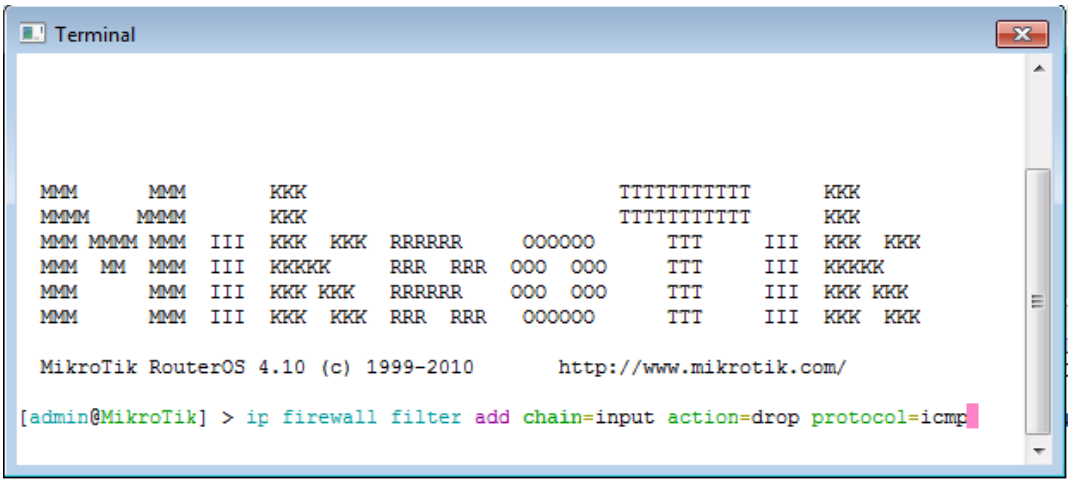


Gambar 4. 12 Rule 5

5. Membuat rule 7 pada terminal mikrotik, dengan perintah

ip firewall filter add chain=input action=drop protocol=icmp

Pada pembuatan rule yang ke 7 mempunya arti apabila suatu ip *client* yang tidak berada dalam kelompok *address-list "safe"*yang mencoba mengakses *protocol icmp* akan di drop



Gambar 4. 13 Rule 6

4.2 Pengujian

Pada tahap ini dilakukan pengujian terhadap VPN *Server* tanpa pengamanan *port knocking* dan yang sudah diberikan pengamanan *port knocking*. Tujuan dari pengujian ini adalah untuk mengetahui tingkat keamanan dari VPN *Server* PT.Padjadjaran Mitra.

Pengujian dilakukan dengan 3 bentuk serangan terhadap VPN *Server*. Bentuk penyerangan terhadap VPN Server PT.Padjadjaran Mitra diantaranya.

1. Pengujian Penyadapan
2. Pengujian *Port Scanning*
3. Pengujian *DDOS Attack*

4.2.1 Skenario Pengujian

Pada tahap pengujian dilakukan beberapa skenario untuk mengetahui tingkat keamanan VPN *Server* pada kondisi yang berbeda. Skenario pengujian yang dilakukan dijelaskan pada table 4.1 Skenario Pengujian VPN *Server*

Tabel 4. 1 Skenario Pengujian VPN *Server*

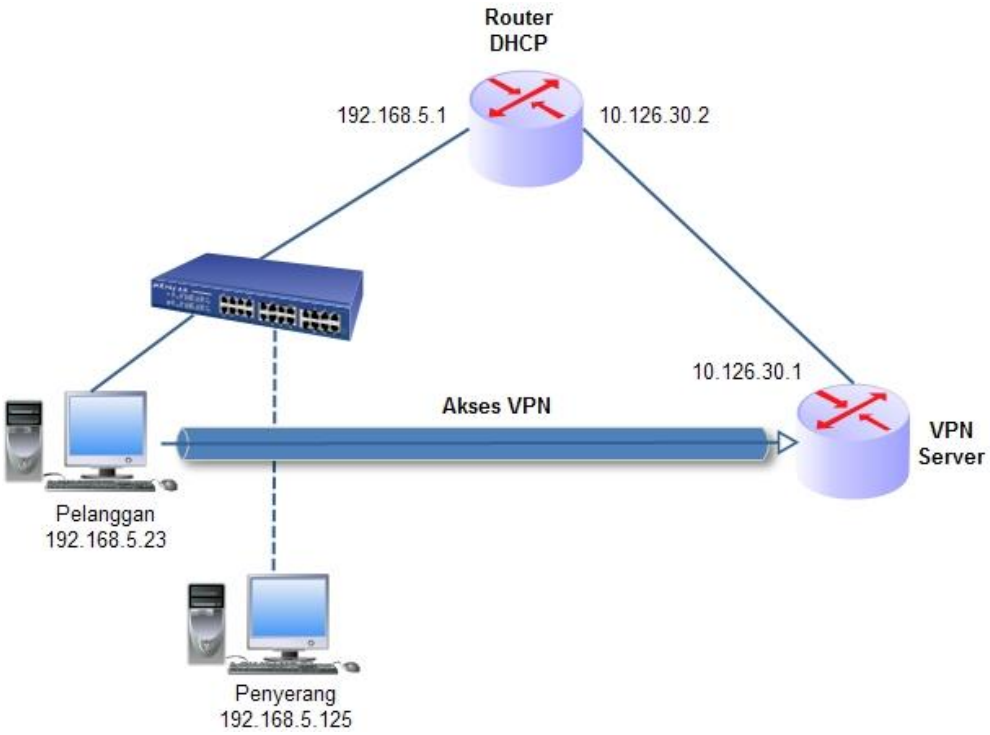
No	VPN *Server*	Waktu Penyerangan
1	Tidak menggunakan *Port Knocking*	Sebelum terkoneksi dengan VPN *Server*
2	Menggunakan *Port Knocking*	Sebelum melakukan *knocking port* terhadap VPN *Server*
3	Menggunakan Port Knocking	Sesudah melakukan*knocking port* terhadap VPN *Server*

4.2.2 Pengujian VPN *Server* tanpa menggunakan *Port Knocking*

Pada tahap ini pengujian dilakukan berdasarkan bentuk penyerangan terhadap VPN *Server*PT.Padjadjaran Mitra.

4.2.2.1 Pengujian Penyadapan

Pada tahap pengujian ini dengan metode *Man In The Middle Attack*. Langkah pertama adalah dengan melakukan *arp spoofing* terhadap ip target untuk mengelabui agar seolah-olah ip penyerang itu adalah *gateway*. Ketika ip target menuju ip penyerang, maka dilakukan lah *Packet Sniffing* untuk mendapatkan *username* dan *password* yang digunakan target untuk terkoneksi dengan VPN *Server*.



Gambar 4. 14 Arsitektur Pengujian Penyadapan

Pengujian ini dilakukan dengan menggunakan Linux Backtrack dan menggunakan tools arpspoofing , wireshark, genkeys dan asleap. Berikut adalah langkah pengujian nya

1. Aktifkan fitur ip forwarding dari kernel. Fitur ini digunakan agar paket-paket data tetap diteruskan ke tujuan semula ketika dilakukan arp spoofing.

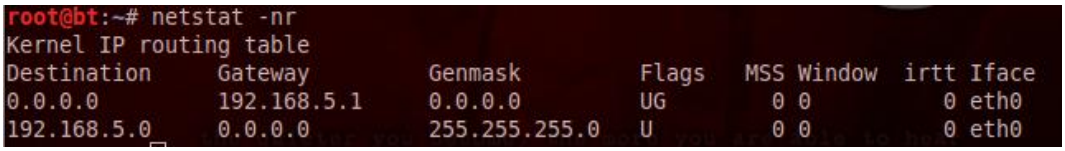
```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
root@bt:~# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Gambar 4. 15 *ip forwarding* kernel

2. Ketahui terlebih dahulu alamat *ip host, guest* dan *gateway* dengan perintah *ifconfig*, serta *netstat -nr*.

netstat –nr

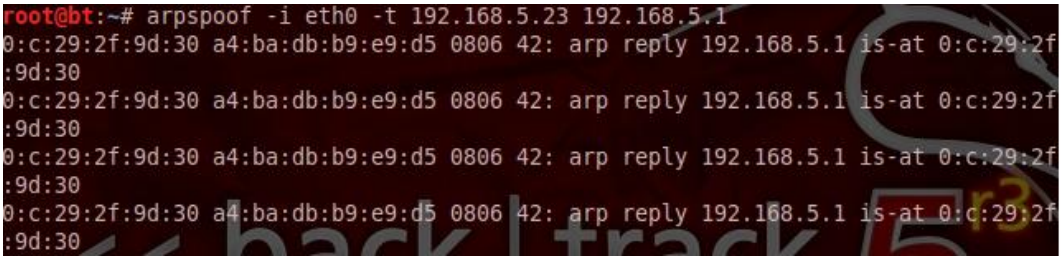


Gambar 4. 16 Netstat

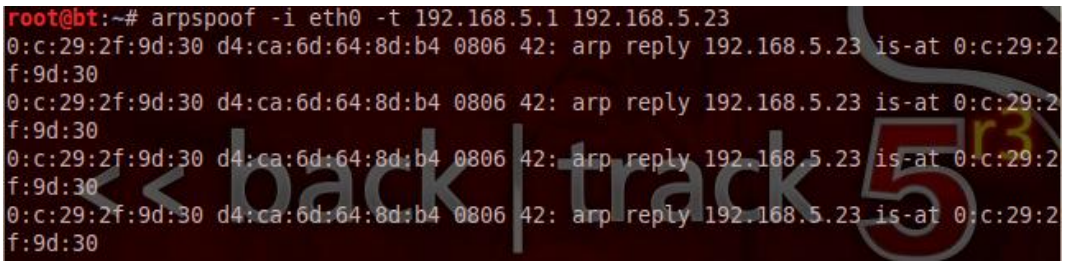
3. Lakukan arpspoof antara komputer target dengan gateway.

arpspoof –i eth0 –t 192.168.5.23 192.168.5.1

arpspoof –i eth0 –t 192.168.5.1 192.168.5.23

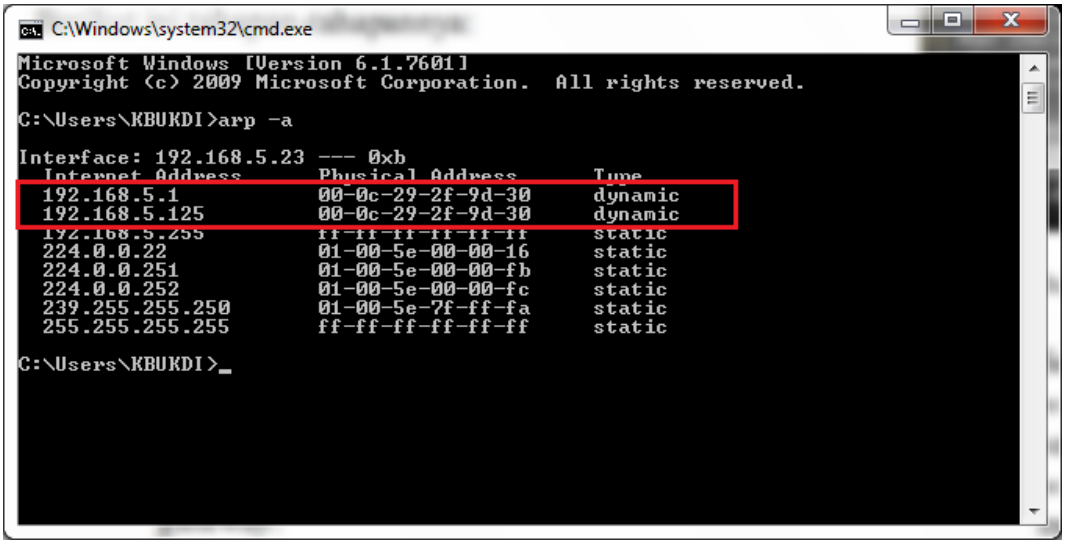


Gambar 4. 17 arpspoofing target-gateway



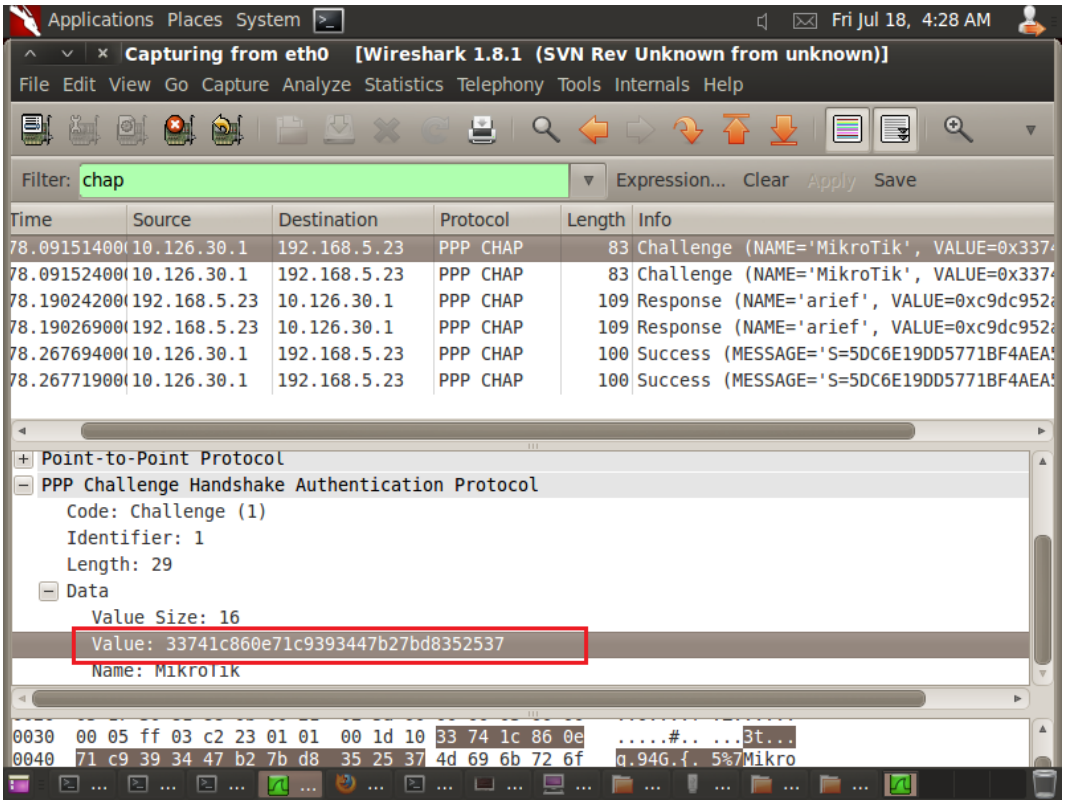
Gambar 4. 18 arpspoofing gateway-target

Jika arpspoofing berhasil dilakukan, maka apabila dicek menggunakan terminal, mac address gateway (192.168.5.1) akan sama dengan mac address penyerang(192.168.5.125).



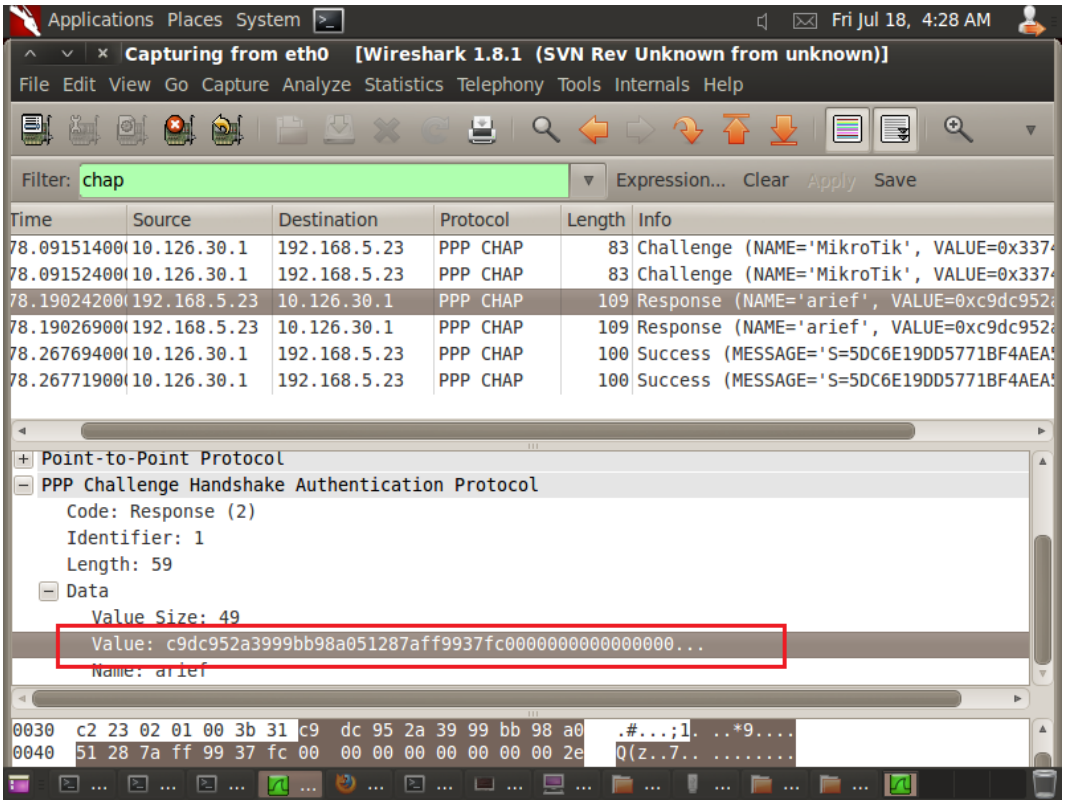
Gambar 4. 19 Mac Address setelah arpspoofing

4. Capture paket dengan menggunakan *wireshark.* Dan filter paket dengan keyword “CHAP”



Gambar 4. 20 PPP Challenge

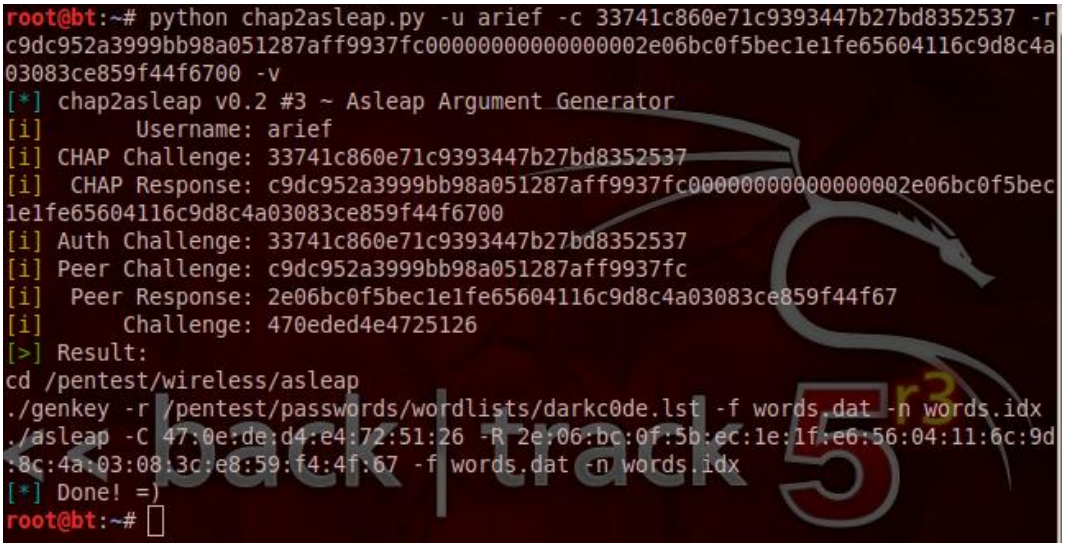
Setelah muncul hasil capture dengan hasil filter “CHAP”, copy *valuePPP CHAP Challenge* tersebut pada gambar 4.21. Copy value *PPP CHAP Response* seperti pada gambar 4.22.



Gambar 4. 21 PPP Response

5. Masukan kedua nilai yang sudah dicopy seperti pada langkah no 4 beserta username dalam argument script chap2asleap.py

python chap2asleap.py –u arief –c 33741c860e71c9393447b27bd8352537–r c9dc952a3999bb98a051287aff9937fc00000000000000002e06bc0f5bec1e1fe65604116c9d8c4a03083ce859f44f6700 -v

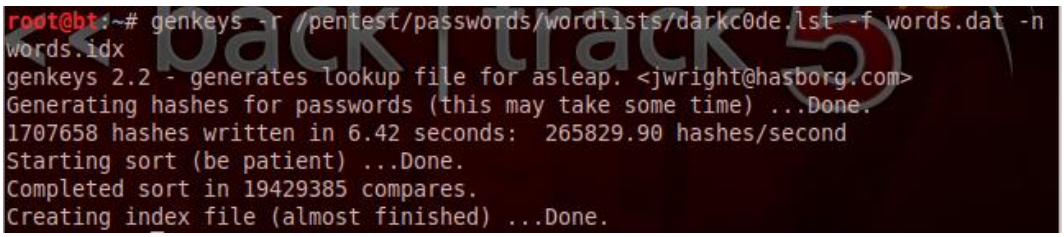


Gambar 4. 22 chap2asleap.py

Script pada langkah nomor 5, bertujuan untuk mengubah format CHAP menjadi format yang dapat dipahami oleh ASLEAP dan digunakan untuk men-*generatewordlist* .

6. Setelah *wordlist* di *generate*, load *wordlist dictionary* ke *genkeys* untuk dibuat NT hashnya dan dibuat indexnya.

genkeys –r /pentest/passwords/wordlists/darkc0de.lst –f word.dat –n words.idx



Gambar 4. 23 Genkeys

Genkeys menghasilkan word.dat yang merupakan file binary NTHash, dan words.idx yang berupa index look up table.

7. Setelah file words.dat dan words.idx dibuat, selanjutnya kedua file tersebut digunakan sebagai argument untuk program ASLEAP.

Asleap –C 47:0e:de:d4:e4:72:51:26 –R 2e:06:bc:0f:5b:ec:1e:1f:6e:56:04:11:6c:9d:8c:4a:03:08:3c:e8:59:f4:4f:67 –f words.dat –n words.idx

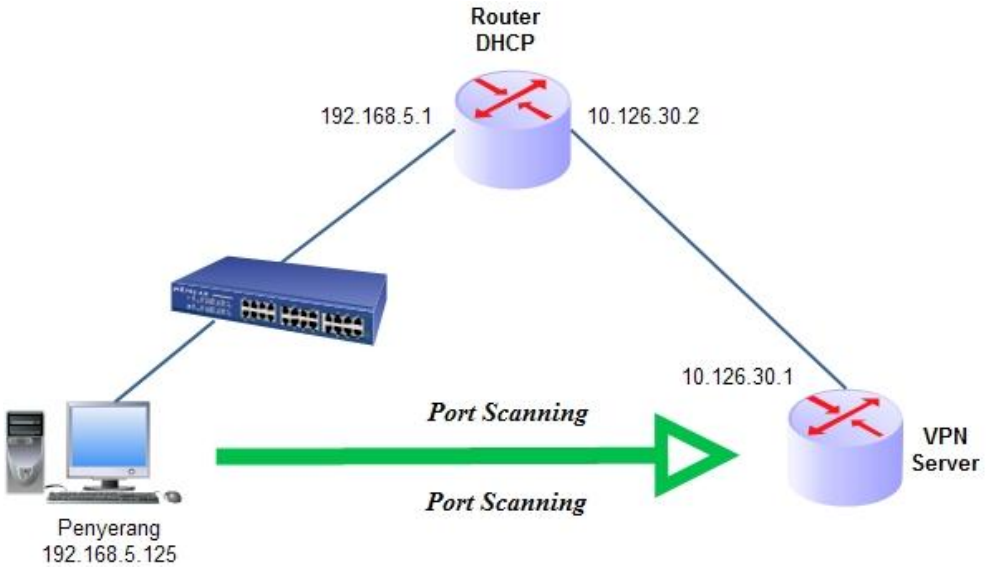


Gambar 4. 24 asleap

Pada Gambar 4.24 password ditemukan dengan BruteForce terhadap Hash, dan dibandingkan dengan file lookup.

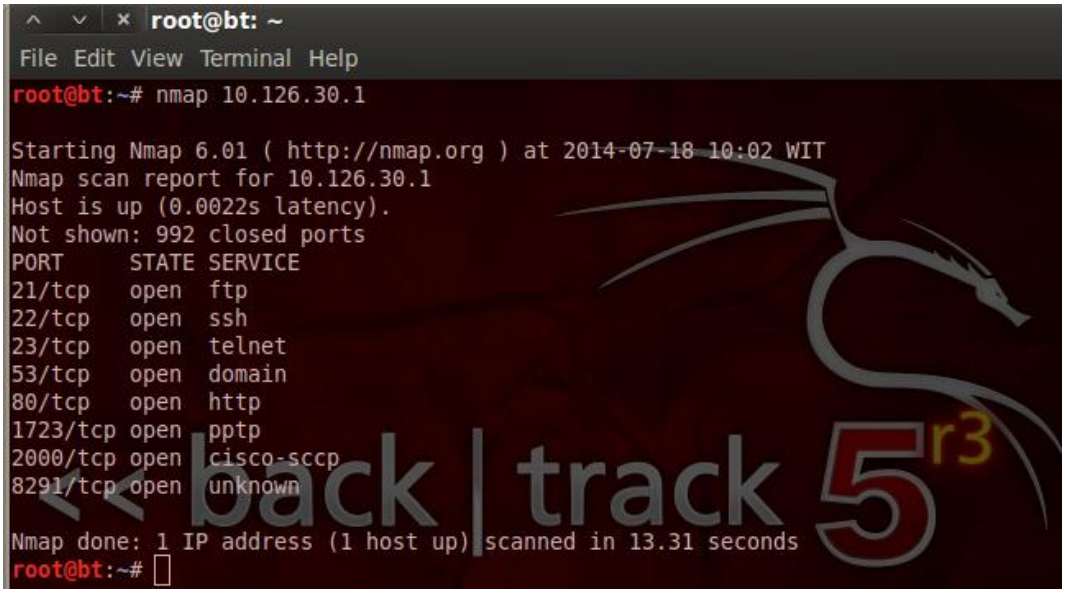
4.2.2.2 Pengujian Port Scanning

Selanjutnya tahap pengujian *Port Scanning* terhadap VPN *Server*. Langkah pertama dengan melakukan *port scanning* dengan menggunakan *nmap* pada ip VPN Server dengan men-*scan* satu-persatu *port* yang terbuka.



Gambar 4. 25 Arsitektur Pengujian Port Scanning

nmap 10.126.30.1

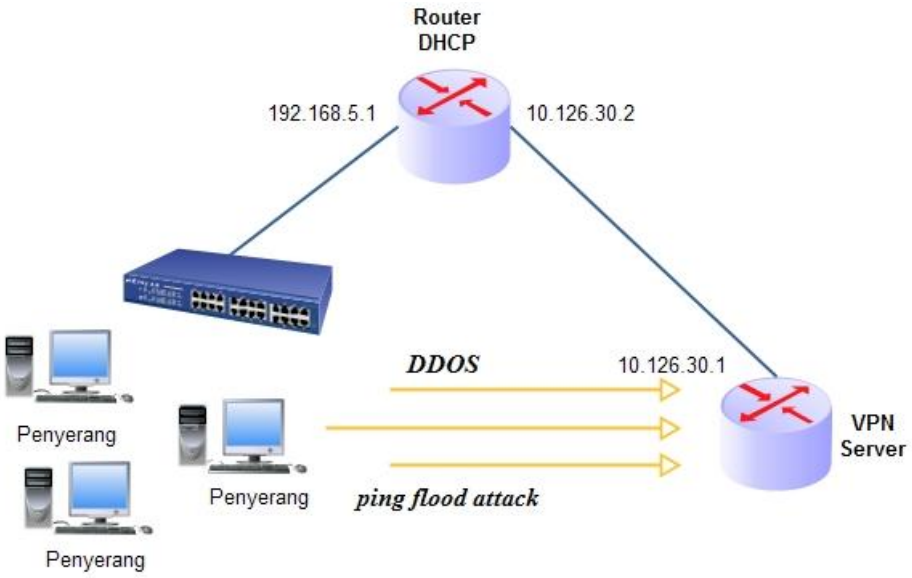


Gambar 4. 26 Port Scanning

Setelah dilakukan *Port Scanning* terhadap VPN *Server* yang belum menggunakan metode Port Knocking terlihat beberapa port-port yang terbuka.

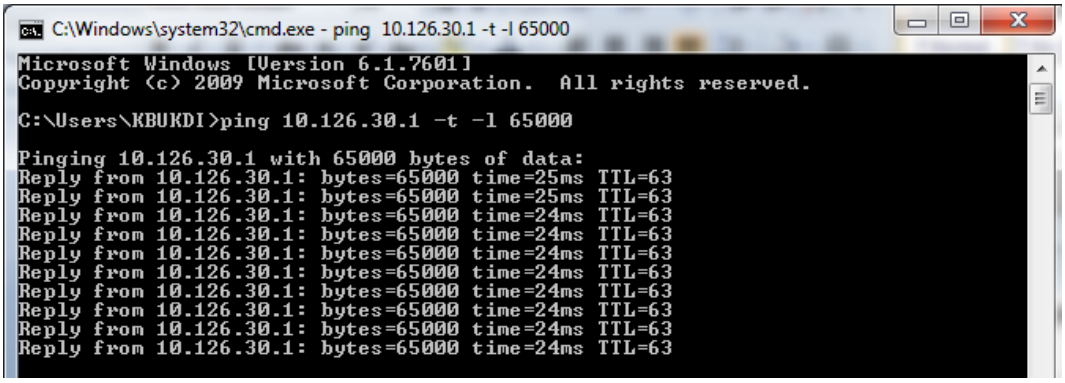
4.2.2.3 Pengujian DDOS Attack

Dan tahap berikutnya pengujian DDOS Attack terhadap VPN Server. Langkap pertama dengan melakukan pengiriman paket secara besar terhadap ip VPN Server.



Gambar 4. 27 Arsitektur DDOS Attack

ping 10.126.30.1 –t –l 65000



Gambar 4. 28 DDOS attack

Dengan dilakukan nya serangan *DDOS Attack* terhadap VPN *Server* membuat traffic lalu lintas data menjadi padat.

	Name	Type	L2 MTU	Tx	Rx	Tx Pac...	Rx Pac...
	cbt_ether	Ethernet	1524	0 bps	0 bps	0	0
	ether1	Ethernet	1524	364 bps	0 bps	1	0
S	ether3	Ethernet	1524	0 bps	0 bps	0	0
S	ether4	Ethernet	1524	0 bps	0 bps	0	0
R	ether5	Ethernet	1524	1091.1 kbps	1059.7 kbps	91	92

Gambar 4. 29 Traffic ethernet 5

4.2.3 Pengujian *VPN Server* dengan *Port Knocking*

Pengujian *VPNServer* yang sudah diamankan dengan menggunakan metode *Port Knocking* menggunakan metode dan tools yang sama dengan pengujian VPN Server tanpa menggunakan Port Knocking pada sub bab 4.2.2. Yang membedakan pada bagian ini adalah sebelum memulai pengaksesan *VPN Server* dilakukan pengetukan *port-port* tertentu yang sudah ditentukan sebelumnya.

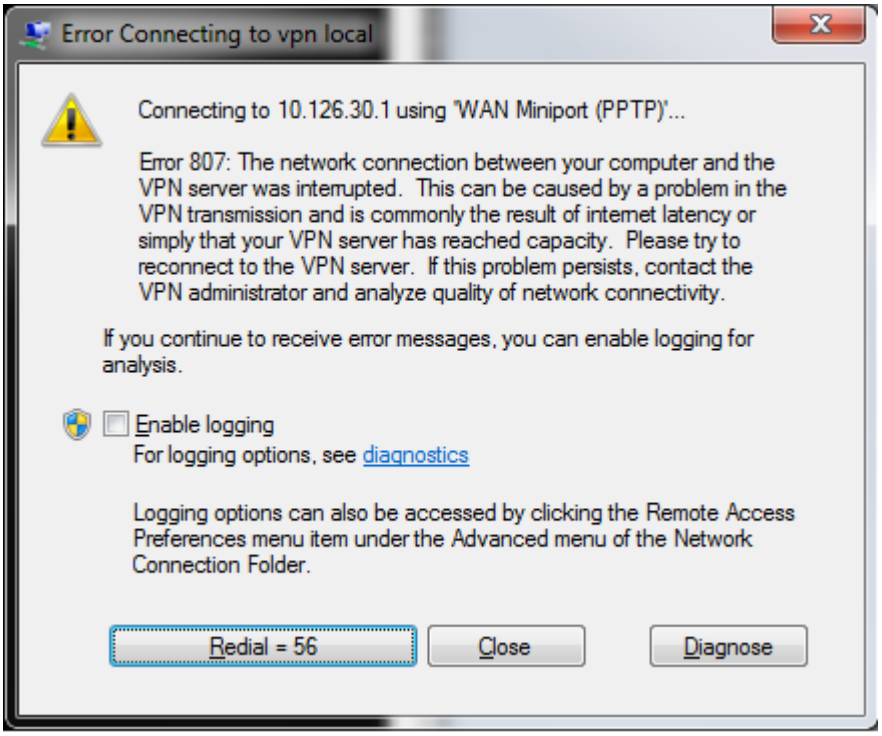
Karena metode *port knocking* pada *VPN Server* telah diaktifkan, maka pada skenario ini pengujian dilakukan untuk menangkal serangan-serangan terhadap VPN Server.

4.2.3.1 Pengujian sebelum melakukan knocking terhadap *VPN Server*

Setelah diaktifkan metode *port knocking* pada *VPN Server* akan dilakukan beberapa pengujian untuk membuktikan penerapan metode *port knocking*. Dan pada pengujian kali ini dilakukan sebelum melakukan ketukan atau *knocking* terhadap *VPN Server*.

4.2.3.1.1 Pengujian terhadap serangan penyadapan

Karena pada sub bab 4.2.2 telah dijelaskan *VPN Server* telah dapat disadap oleh pihak yang tidak bertanggung jawab, sehingga setelah diaktifkan nya metode *port knocking* pada saat melakukan koneksi terhadap *VPN Server*, username dan password yang didapat tidak diperkenankan terkoneksi pada *VPN Server* sebelum melakukan knocking terlebih dahulu.



Gambar 4. 30 Connection Interrupted

4.2.3.1.2 Pengujian terhadap port scanning

Pada pengujian ini, setelah diaktifkannya metode *port knocking* pada *VPN Server* sehingga menyebabkan nya port-port yang sebelumnya terbuka menjadi tertutup atau menjadi ter*filter*. Ketika dilakukan *nmap* terhadap *VPN server* akan diperoleh hasil sebagai berikut.

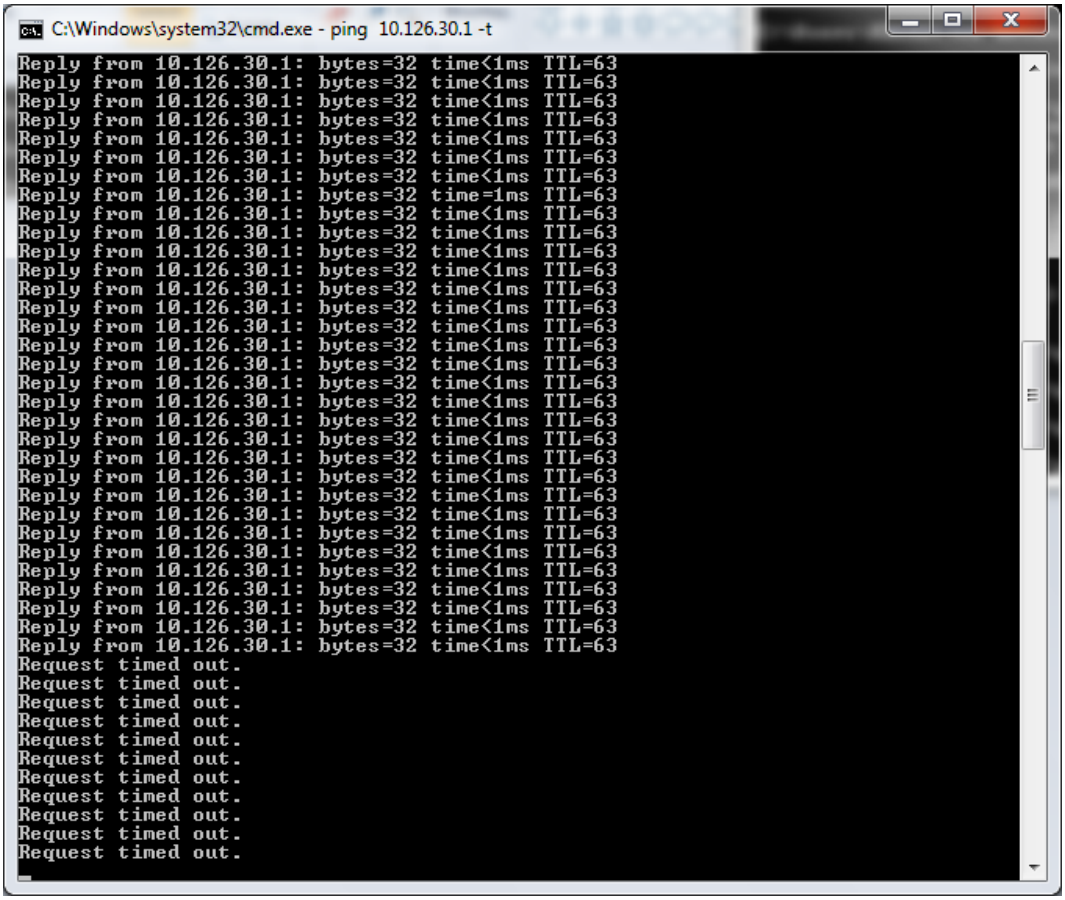


Gambar 4. 31 Filtered scan

Didapati port-port yang sebelumnya dalam keadan terbuka atau open, setelah penerapan metode *port knocking* status nya berubah menjadi *filtered.*

4.2.3.1.3 Pengujian terhadap *DDOS Attack*

Pada dasarnya *ddos attack* yang menyerang *VPN server* menggunakan protokol *icmp* dengan melakukan ping. sehingga setelah diaktifkan nya metode *port knocking*. Akses atau perintah ping yang akan menuju ke server akan ditolak oleh server *VPN Server*, sehingga hasil ping yang didapati pun seperti pada gambar 4.33 dibawah ini.



Gambar 4. 32 *DDOS Failed*

Pada hasil pengujian diatas setelah diaktifkan metode port knocking pada *VPN Server*, ping yang sebelumnya bisa dilakukant terhadap server menjadi *Request Time Out*

4.2.3.2 Pengujian setelah melakukan knocking terhadap *VPN Server*

Setalah proses pengujian yang telah dijelaskan pada sub bab 4.2.2, maka selanjutnya untuk meningkatkan kemanan pada *VPN server* ini, dengan ditambahkan 2 lapis keamanan pada *VPN server* ini agar pengamanan pada *VPN server* ini lebih aman. Sehingga ketika tahap sebelumya banyak penyerang atau penyadap dapat langsung menyerang target VPN Server, dengan menguggnakan metode pengamanan berlapis dengan port knocking para penyerang pun lebih sulit untuk mengakses *VPN Server*.

Setelah sebelumnya dilakukan pengujian akses *VPN Server* sebelum melakukan pengetukan atau *knocking* , selanjutnya kan dilakukan pengujian akses VPN Server dengan melakukan pengetukan atau knocking terhadap VPN Server terlebih dahulu.

1. Lakukan pengetukan port pada VPN server yang dituju.
   knock.exe 10.126.30.1 1007:tcp



Gambar 4. 33 knocking port 1007

2. Lakukan ketukan yang kedua terhadap VPN server yang dituju.
   knock.exe 10.126.30.1 7001:tcp



Gambar 4. 34 knocking port 7001

Dan setalah melakukan *knocking* terhadap *port* yang sudah ditentukan, user atau admin tersebut sudah bisa langsung mengakses *VPN server* tersebut.

4.2.4 Hasil Pengujian

Hasil pengujian pada penelitian ini diperoleh berdasarkan pengujian yang telah dilakukan berdasarkan skenario pada tabel 4.2 Skenario Pengujian *VPN Server*. Hasil pengujian terdiri dari pengujian

Tabel 4. 2 Hasil Pengujian

No	Pengujian	*VPN Server*tanpa *Port Knocking*	*VPN Server* dengan *Port Knocking*	
			Sebelum melakukan *knocking port*	Sesudah melakukan *knocking port*
1	Penyadapan	Berhasil Login VPN	Gagal login *VPN*	Berhasil Login *VPN*
2	*Port Scanning*	Berhasil mengetahui *port* yang terbuka	*Port* menjadi tertutup	*Port* menjadi terbuka kembali
3	*DDOS Attack*	Berhasil melakukan *flooding attack*	*Flooding attack* gagal dilakukan	Ping bisa kembali dilakukan