Implementasi Port-Knocking di Mikrotik dengan Menggunakan Komponen Delphi TcpClient

Agus Prihanto[1]
[1] Prodi D3 Manajemen Informatika, Jurusan Tekni Elektro, Universitas Negeri Surabaya, Surabaya.
E-mail: cogierb201@yahoo.com

***Abstrak* – Membuka service atau port pada device jaringan seperti PC Server, Router, dll akan mempermudah pekerjaan seorang administrator jaringan dalam melakukan administrasi sistem secara *remote* dimanapun mereka berada, namun disisi lain dapat menjadi ancaman yang serius karena orang yang tidak berkepentingan seperti hacker dapat menjadikan service tersebut sebagai pintu masuk ke jaringan kita. Solusi alternatifnya kita dapat menggunakan beberapa metode keamanan seperti membatasi ip yang dapat mengakses service tersebut, menggunakan VPN, memasang IDS, merubah port service diluar port default, menggunakan metode port knocking, dll. Dalam penelitian ini akan dikembangkan tool port knocking dengan bahasa pemrograman delphi dengan memanfaatkan komponen TcpClient.**

Hasil pengujian menunjukkan bahwa tool port-knocking yang telah dikembangkan mampu mengenerate script firewall filter rule yang dapat diexport ke dalam file rsc dan dapat diimport ke mikrotik. Hasil pengujian lain dengan nmap menunjukkan setelah knocking berhasil dilakukan, maka port 22, 23 dan 80 statusnya menjadi open, sedangkan pengujian dengan ping menunjukkan setelah knocking berhasil dilakukan, maka status ping menjadi *Reply* yang sebelumnya *Request Time Out.*

Kata Kunci : port knocking, mikrotik, firewall, TcpClient

I. PENDAHULUAN

Sudah menjadi sebuah hal umum, seorang administrator jaringan menyediakan service remote login untuk mengakses ke device jaringan yang mereka kelola seperti router, PC Server, dll guna mempermudah pekerjaan mereka dalam melakukan administrasi sistem dimanapun mereka berada[3].

Mikrotik sebagai salah satu dedicated router yang mempunyai banyak service *remote login* seperti ssh service (22), telnet (23), webfix (80), winbox (8291) merupakan port-port yang dapat digunakan untuk mengendalikan router tersebut. Hal ini cukup membantu administrator dalam mengelolah jaringan mereka, namun sekaligus dapat menjadi ancaman yang serius karena orang yang tidak berkepentingan seperti hacker dapat menjadikan service tersebut sebagai pintu masuk ke jaringan kita.

Solusi alternatifnya kita dapat menggunakan beberapa metode keamanan seperti membatasi ip yang dapat mengakses service tersebut, menggunakan VPN, memasang IDS, merubah port service diluar port default, menggunakan metode port knocking, dll.

Dalam penelitian ini akan dikembangkan tool port knocking dengan bahasa pemrograman delphi dengan memanfaatkan komponen TcpClient.

II. KAJIAN PUSTAKA

1. Port Knocking

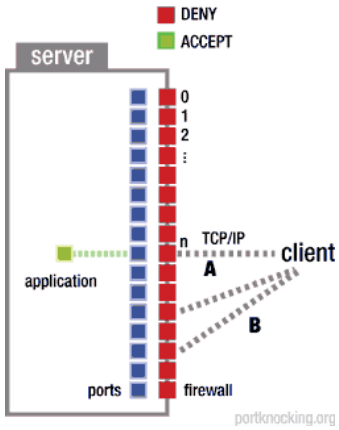
Port knocking adalah sebuah metode membuka port secara eksternal melalui firewall dengan cara melakukan usaha koneksi pada suatu port yang tertutup dengan urutan upaya koneksi yang telah ditentukan[1]. Dengan kata lain port knocking adalah sebuah metode untuk membangun sebuah komunikasi *host-to-host* dengan perangkat komputer yang tidak membuka port komunikasi apapun secara bebas. Port knocking diimplementasikan dengan mengkonfigurasi sebuah program kecil yang disebut daemon guna memonitor log firewall untuk permintaan koneksi dan menentukan apakah klien terdaftar pada alamat IP yang disetujui dan telah melakukan urutan ketukan yang benar. Jika jawabannya adalah ya, firewall akan membuka port yang terkait secara dinamis.

Gambar 1 : analogi port knocking dengan kunci brangkas dengan angka kombinasi

Tujuan utama dari port knocking adalah mencegah penyerang dari pemindai sistem yang mudah dieksplotasi seperti SSH dengan melakukan port scanning. Jika penyerang mengirimkan urutan ketukan yang salah, port yang dilindungi tidak akan muncul atau terbuka[3].

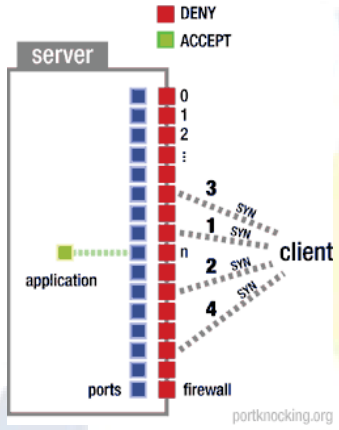
Berikut adalah 4 tahapan port scanning[5] :

Tahap 1



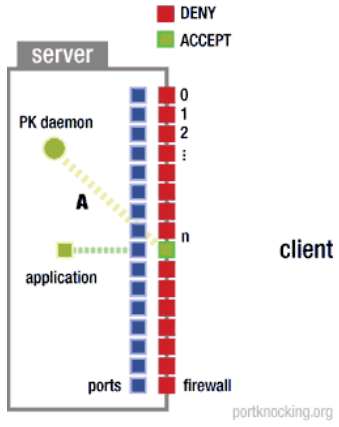
Gambar 2 : (A) client tidak dapat tersambung dengan aplikasi di server yang listen di port n, (B) client tidak dapat membangun koneksi ke beberapa port

Tahap 2



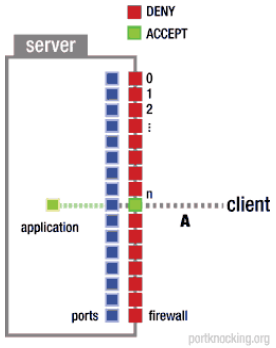
Gambar 3 : (1,2,3,4) client mencoba mengirim paket SYN ke server dengan urutan tertentu tetapi tidak menerima ACK dari server.

Tahap 3



Gambar 4 : Jika urutan port packet SYN yang dikirim sesuai rule server, maka server akan membuka port untuk client tersebut.

Tahap 4



Gambar 5 : Client dapat tersambung ke port n dan diijinkan untuk melakukan authentikasi ke aplikasi.

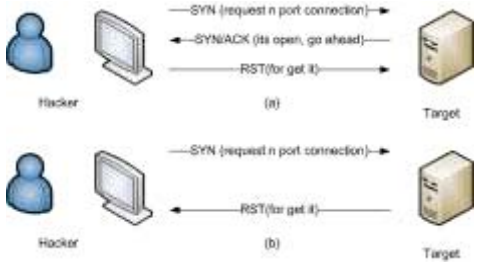
2. Scanning

Scanning adalah kegiatan yang dilakukan hacker untuk menentukan aktif tidaknya host target dalam jaringan. Hasil scanning dapat berupa IP Address, sistem operasi, service maupun aplikasi yang dijalankan. Setelah mengetahui port-port yang active, para hacker biasanya akan melancarkan serangan *brute-force attack* pada port tersebut guna mendapatkan password yang valid. Dengan password tersebut, hacker melakukan login dan mendapatkan akses full terhadap komputer korban dan dapat melakukan hal-hal yang merugikan[2][7].

3. Nmap

Nmap merupakan utilitas berlisensi *open source* yang berfungsi untuk *discovery* jaringan dan *audit* keamanan. Nmap menggunakan *raw IP packet* untuk menentukan apakah host tersedia pada jaringan, service apa yang sedang berjalan di host (nama service dan versi), sistem operasi (versi OS) yang host jalankan, jenis firewall yang sedang digunakan dan banyak karakteristik lainnya. Nmap dirancang untuk memindai jaringan besar secara cepat, tetapi bekerja baik juga untuk host tunggal[2][4].

Nmap mengirimkan paket SYN kepada target pada proses port scanning untuk menemukan port mana yang terbuka. Jika port korban berbuka, maka nmap akan mendapatkan paket balasan berupa paket SYN dan ACK, bila port tertutup maka nmap akan menerima paket RST.



Gambar 6 : (a) kirim SYN paket ke port yang terbuka, (b) kirim SYN paket, target memberikan respon RST.

4. Brute Force

Brute-force attack adalah sebuah teknik serangan terhadap sebuah sistem keamanan komputer yang menggunakan percobaan terhadap semua kunci yang mungkin untuk memecahkan password, kunci, kode, atau sebuah kombinasi[2].

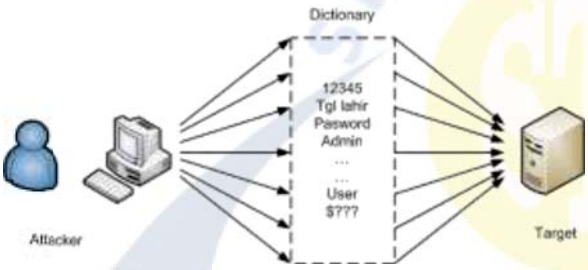
Dalam proses penebakan password, Brute-force attack dapat diimplementasikan dengan metode sebagai berikut :

a). *Brute Force Attack*

Metode ini diimplementasikan dengan cara menentukan *range of character set* dan mengkomputasikan setiap kemungkinan kombinasi karakter yang ada. Metode ini umumnya menggunakan kombinasi karakter yang terdiri dari huruf saja, huruf angka, huruf angka + *special character* atau setiap karakter pada table ASCII.

b). *Brute Force Attack with Dictionary*

Metode *Dictionary Attack* atau biasa disebut serangan kamus ini diimplementasikan dengan mengkomputasi secara berangsur-angsur setiap kata tunggal atau modifikasi kata dari sebuah kamus dan dicocokan dengan password pengguna tertentu.



Gambar 7 : Brute Force attack dengan kamus

4. Firewall

Firewall adalah perangkat yang berfungsi untuk memeriksa dan menentukan paket data yang dapat keluar atau masuk dari sebuah jaringan. Dengan kemampuan tersebut maka firewall berperan dalam melindungi jaringan dari serangan yang berasal dari jaringan luar (*outside network*). Firewall mengimplementasikan *packet filtering* dan dengan demikian menyediakan fungsi keamanan yang digunakan untuk mengelola aliran data ke, dari dan melalui router. Sebagai contoh, firewall difungsikan untuk melindungi jaringan lokal (LAN) dari kemungkinan serangan yang datang dari Internet. Selain untuk melindungi jaringan, firewall juga difungsikan untuk melindungi komputer user atau host (*host firewall*)[6].

Firewall digunakan sebagai sarana untuk mencegah atau meminimalkan risiko keamanan yang melekat dalam menghubungkan ke jaringan lain. Firewall jika dikonfigurasi dengan benar akan memainkan peran penting dalam penyebaran jaringan yang efisien dan infrastruktur yang aman . MikroTik RouterOS memiliki implementasi firewall yang sangat kuat dengan fitur-fitur sebagai berikut :

- stateful packet inspection
- layer-7 protocol detection
- peer-to-peer protocols filtering
- traffic classification by:
- source MAC address
- IP addresses (network or list) and address types (broadcast, local, multicast, unicast)
- port or port range
- IP protocols
- protocol options (ICMP type and code fields, TCP flags, IP options and MSS)
- interface the packet arrived from or left through
- internal flow and connection marks
- DSCP byte
- packet content
- rate at which packets arrive and sequence numbers
- packet size
- packet arrival time
- dll

Ada tiga chain filter yang telah ditetapkan pada RouterOS Mikrotik :

- *Input* - digunakan untuk memproses paket memasuki router melalui salah satu interface dengan alamat IP tujuan yang merupakan salah satu alamat router. Chain input berguna untuk membatasi akses konfigurasi terhadap Router Mikrotik.
- *Forward* - digunakan untuk proses paket data yang melewati router.
- *Output* - digunakan untuk proses paket data yang berasal dari router dan meninggalkan melalui salah satu interface.

Pada konfigurasi firewall mikrotik ada beberapa pilihan Action, diantaranya :

- *accept* : paket diterima dan tidak melanjutkan membaca baris berikutnya
- *add dst to address list* : menambahkan destination ip ke address-list
- *add srv to address list* : menambahkan source ip ke address-list
- *drop* : menolak paket secara diam-diam (tidak mengirimkan pesan penolakan ICMP)
- *reject* : menolak paket dan mengirimkan pesan penolakan ICMP
- *tarpit* : menolak, tetapi tetap menjaga TCP connection yang masuk (membalas dengan SYN/ACK untuk paket TCP SYN yang masuk)
- *passthrough* : mengabaikan rule ini dan menuju ke rule selanjutnya
- *log* : menambahkan informasi paket data ke log
- *dll*

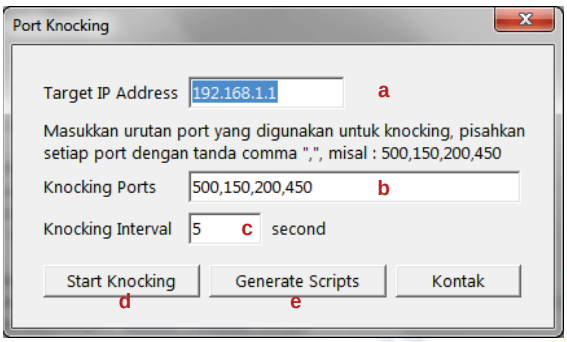
III. DESIGN DAN IMPLEMENTASI

Dalam implementasi pembuatan tool port knocking dengan komponen delphi TcpClient ini menggunakan tahapan sebagai berikut :

1. Pembuatan Aplikasi

Aplikasi ini mempunyai 2 fungsi utama yaitu sebagai tool untuk mengirimkan packet knocking ke router mikrotik dan sebagai generator script firewall filter untuk menjalankan daemon port knocking diserver.

Berikut adalah design interface aplikasi packet knocking



Gambar 8 : interface port knocking

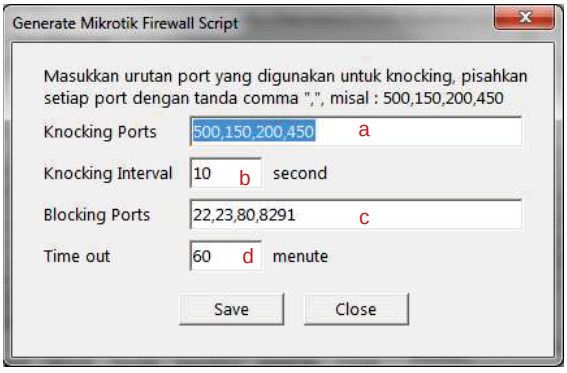
Keterangan :
(a) target ip router mikrotik
(b) urutan port yang digunakan untuk knocking
(c) interval waktu diantara pengiriman packet knocking setiap portnya
(d) memulai menjalankan pengiriman packet
(e) generate script firewall filter input

Berikut adalah procedure script untuk mengirimkan packet knocking :

```
procedure sendknock(vip,vport:string);
begin
mytcpclient.Close;
mytcpclient.RemoteHost:=vip;
mytcpclient.RemotePort:=vport;
mytcpclient.Open;
beep;
end;
```

Gambar 9 : procedure sendknock

Berikut adalah design interface generate script firewall input port knocking



Gambar 10 : interface generate script

Keterangan :
(a) urutan port yang digunakan untuk knocking
(b) interval waktu diantara pengiriman packet knocking setiap portnya
(c) port yang akan diblock/didrop di service mikrotik
(d) batas waktu kita diijinkan akses service mikrotik setelah berhasil knocking.

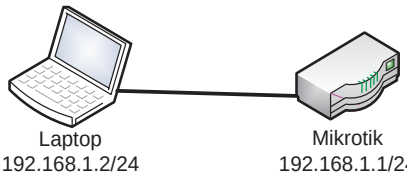
Berikut adalah procedure generate scipt :

```
procedure generatescript;
var scripts,ports : TStringList;
    i : byte;
begin
scripts:=TStringList.Create;
ports  :=TStringList.Create;
ports.Delimiter:=',';
ports.DelimitedText:=edknockports.Text;

scripts.Add(format('/ip firewall
filter',[]));

i:=0;

scripts.Add(format('add action=add-src-to-
address-list address-list=whitelist%d
address-list-timeout=%ss chain=input
disabled=no dst-port=%s
protocol=tcp',[i,edinterval.text,ports[i]]));

for i:=1 to ports.Count-2 do
  scripts.Add(format('add action=add-src-to-
address-list address-list=whitelist%d
address-list-timeout=%ss chain=input
disabled=no dst-port=%s protocol=tcp src-
address-
list=whitelist%d',[i,edinterval.text,ports[i]
,i-1]));

scripts.Add(format('add action=add-src-to-
address-list address-list=secure address-
list-timeout=%sm chain=input disabled=no dst-
port=%s protocol=tcp src-address-
list=whitelist%d',[edtimeout.text,ports[i],i-
1]));

scripts.Add(format('add action=accept
chain=input disabled=no src-address-
list=secure',[]));

scripts.Add(format('add action=drop
chain=input disabled=no dst-port=%s
protocol=tcp',[edblockports.text]));

scripts.Add(format('add action=drop
chain=input disabled=no protocol=icmp',[]));

scripts.Add(format('',[]));
scripts.SaveToFile('knocking.rsc');

ShowMessage('Generate scripts "knocking.rsc"
success !!');
ports.Free;
scripts.Free;
end;
```

Gambar 11 : procedure generate script

2. Design Jaringan Ujicoba

Dalam ujicoba ini memakai router mikrotik RB750 dan 1 Laptop untuk menjalankan tool port knocking yang terhubung secara LAN.



Gambar 12 : Design jaringan ujicoba

IV. HASIL DAN PEMBAHASAN

1. Export dan Import generate script

Pengujian dilakukan dengan memasukkan parameter sebagai berikut :

- Knocking ports = 500,150,200,450
- Knocking interval = 10 second
- Blocking Ports = 22(ssh), 23(telnet), 80(webfig), 8291(winbox)
- Time out = 60 menute

Kemudin hasil export generate scriptnya disimpan dengan nama *knocking.rsc*.:

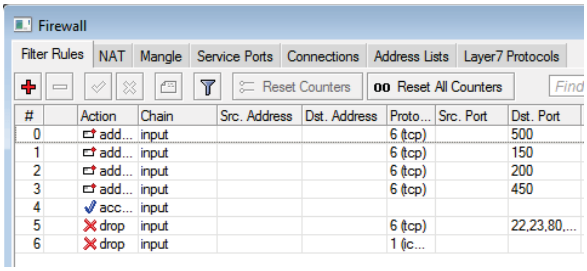
```
/ip firewall filter
add action=add-src-to-address-list address-
list=whitelist0 address-list-timeout=10s
chain=input disabled=no dst-port=500
protocol=tcp

add action=add-src-to-address-list address-
list=whitelist1 address-list-timeout=10s
chain=input disabled=no dst-port=150
protocol=tcp src-address-list=whitelist0

add action=add-src-to-address-list address-
list=whitelist2 address-list-timeout=10s
chain=input disabled=no dst-port=200
protocol=tcp src-address-list=whitelist1

add action=add-src-to-address-list address-
list=secure address-list-timeout=60m
chain=input disabled=no dst-port=450
protocol=tcp src-address-list=whitelist2

add action=accept chain=input disabled=no
src-address-list=secure

add action=drop chain=input disabled=no dst-
port=22,23,80,8291 protocol=tcp

add action=drop chain=input disabled=no
protocol=icmp
```

Gambar 13 : hasil generate script *knocking.rsc*

Scrip di atas kemudian diimport di mkrotik dengan perintah :

```
<admin@router>import knocking.rsc
```

Maka akan muncul rule pada bagian ip firewall filter di mikrotik sebagai berikut :



Gambar 14 : Firewall Filter Rule

2. Pengujian dengan nmap

Pengujian ini bertujuan untuk mengetahui port yang tersedia di router mikrotik baik sebelum knock dan sesudah knocking berhasil dilakukan.

a) Pengujian sebelum knocking berhasil

```
Nmap scan report for 192.168.1.1
Host is up (0.00023s latency).
Not shown: 95 closed ports

PORT      STATE    SERVICE
21/tcp    open     ftp
22/tcp    filtered ssh
23/tcp    filtered telnet
80/tcp    filtered http
2000/tcp open      cisco-sccp

Nmap done: 1 IP address (1 host up) scanned
in 13.02 seconds
```

Gambar 15 : nmap sebelum knocking berhasil

Terlihat bahawa port 22, 23 dan 80 mempunyai state *filtered*.

b) Pengujian sesudah knocking berhasil

```
Nmap scan report for 192.168.1.1
Host is up (0.00095s latency).
Not shown: 95 closed ports

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http
2000/tcp open  cisco-sccp

Nmap done: 1 IP address (1 host up) scanned
in 12.09 seconds
```

Gambar 16 : nmap sebelum knocking berhasil

Terlihat bahawa port 22, 23 dan 80 mempunyai state *open* yang sebelumnya *filtered*.

3. Pengujian dengan ping

Sesaat sebelum knocking dilakukan maka perintah ping ke mikrotik dijalankan, kemudian tool port knocking yang telah dibuat juga dijalankan. Jika knocking telah memenuhi firewall rule mikrotik maka hasilnya nampak seperti gambar berikut :

Gambar 17 : Tes ping sesaat sebelum dan sesudah knocking berhasil dilakukan

Terlihat bahwa sesaat sebelum knocking berhasil dilakukan maka hasil ping menunjukkan *Request Time Out* dan setelah knocking selesai dan berhasil statusnya berubah menjadi *Reply*.

V. KESIMPULAN

Dari hasil beberapa pengujian dapat diambil kesimpulan sebagai berikut :

1. Pengujian export dan inport menunjukkan bahwa tool port knocking yang dibuat dapat digunakan untuk mengenerate script yang digunakan untuk membuat rule firewall filter di mikrotik.
2. Pengujian dengan nmap menunjukkan sebelum knocking berhasil maka port 22, 23, 80 berstatus *filtered* dan setelah berhasil dilakukan knocking maka port tersebut berstatus *open*.
3. Pengujian dengan ping menunjukkan bahwa sebelum knocking berhasil maka status ping adalah *RTO (Request Time Out)* dan setelah berhasil dilakukan knocking maka status ping berubah menjadi *Reply*.

REFERENSI

[1]. Krzywinski, M. 2003. Port Knocking: Network Authentication Across Closed Ports. SysAdmin Magazine 12: 12-17.

[2]. Haryanto, EDI, 2013, Meningkatkan Keamanan Port SSH dengan Metode Port Knocking Menggunakan Shorewall Pada Sistem Operasi Linux, Amikom, Yogyakarta

[3]. Saptono, Henry, 2011, Metode Port Knocking dengan Iptables untuk membuka port SSH, Infolinux, Jakarta

[4]. Lyon, Gordon Fyodor, 2009, Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning, Insecure, USA.

[5]. Krzywinski, M. 2013. Documentation Port Knocking, portknocking.org, Canada.

[6]. Agung, Rizky, 2013, Dasar Mikrotik: Dasar Mikrotik Firewall, http://mikrotikindo.blogspot.com/2013/03/belajar-mikrotik-dasar-firewall-mikrotik.html, diakses 25 November 2013

[7]. Prihanto, Agus, 2013, Scanning, http://cogierb201.wordpress.com/2013/03/19/scanning/, diakses 25 November 2013