Puribersama Jurnal Publikasi Riset Bersama Dosen dan Mahasiswa
Volume 2 Nomor 6, Desember 2023

e-ISSN : 2828-4518
p-ISSN : 2828-3694

Membuat Akses Legal Terhadap Port Tertentu Yang Telah Ditutup oleh Firewall dengan Metode Port Knocking

Awan*
Fakultas Sains Teknologi, Sistem Informasi, Universitas IBBI
Email: one.awan@email.com
(*: coressponded author)

Abstrak: *Salah satu kesulitan utama dalam melindungi jaringan (terutama yang berjalan memberikan layanan) adalah bahwa, untuk sebagian besar terlihat dan dapat mengungkapkan informasi kepada siapa saja yang meminta. Jika seorang hacker (penyerang) menemukan server FTP perusahaan, ia dapat terhubung ke layanan tersebut dan dengan sangat detail akan diperlihatkan versi Software FTP yang sedang berjalan., Hacker kemudian dapat menggunakan informasi ini untuk memeriksa apakah ada atau tidak versi dari software ini yang rentan terhadap serangan tertentu yang akan memberinya hak root (admin) untuk akses ke server (atau mencoba dengan teknik brute force kepada username dan password). Banyak administrator jaringan tidak selalu memiliki waktu untuk patch dan up-to-date, dan bahkan kemudian, banyak layanan yang disebut dibiarkan saja. Jadi bagaimana administrator jaringan dapat melindungi server jaringan terhadap hal ini? Satu jawaban sederhana adalah dengan mematikan semua layanan yang tidak perlu, yang lain adalah dengan menggunakan perlindungan Firewall untuk mencegah siapa pun, kecuali kelompok IP tertentu, yang dapat terhubung ke layanan tersebut, yang jelas sangat ketat dalam hal ini. Lalu bagaimana cara untuk mencoba menjaga server jaringan tersembunyi dari calon penyerang, namun memungkinkan pengguna yang sah/legal untuk terhubung ke layanan yang berjalan pada server itu? Dalam arti luas, port knocking adalah metode dan cara transmisi Informasi dengan port yang telah tertutup agar dapat masuk, dengan tujuan otentikasi pengguna sebelum dapat mengakses layanan yang dilindungi tersebut.*

Kata Kunci: *Port Knocking, Firewall, IP, Layanan*

Abtract: *One major difficulty in protecting networked machines (especially those running services) is that they are, for the most part, visible and happy to disclose information to anyone who asks. If a hacker finds a corporate FTP server, he can connect to it and it will gladly tell him exactly what version of what FTP software it is running (if the FTP server hasn't been hardened, which is usually still the case). He can then use this information to check whether or not that version of the software is vulnerable to a particular attack which would give him root (admin) access to the server machine (or attempt to brute force the username and password). Many people, unfortunately, do not always have the time to keep all of their machines patched and up-to-date, and even then, many services have so-called 0day exploits for which patches do not even exist yet! So how can they protect themselves against this? One simple answer is to turn off all unnecessary services; another is to use a Firewall to prevent anyone, except a specific group of IPs, from connecting to that service, which is obviously very restrictive in terms of who is able to connect. How does one try to keep a machine hidden from would-be attackers, yet allow legitimate users to connect to services running on that machine? Enter Port Knocking. In broad terms, port knocking is a method for transmitting information across closed ports, with the aim of authenticating users before allowing them, and only them, to access a protected service.*

Keywords: *Port Knocking, Firewall, IP, Service*

e-ISSN : 2828-4518
p-ISSN : 2828-3694

1. PENDAHULUAN

Dalam jaringan komputer, port knocking adalah metode membuka *port* pada *firewall* yang dengan jelas sudah ditutup. Namun dalam waktu tertentu koneksi yang sah (legal) bila diterima, pada *firewall* dapat memperbolehkan koneksi atas *port* yang telah ditentukan.

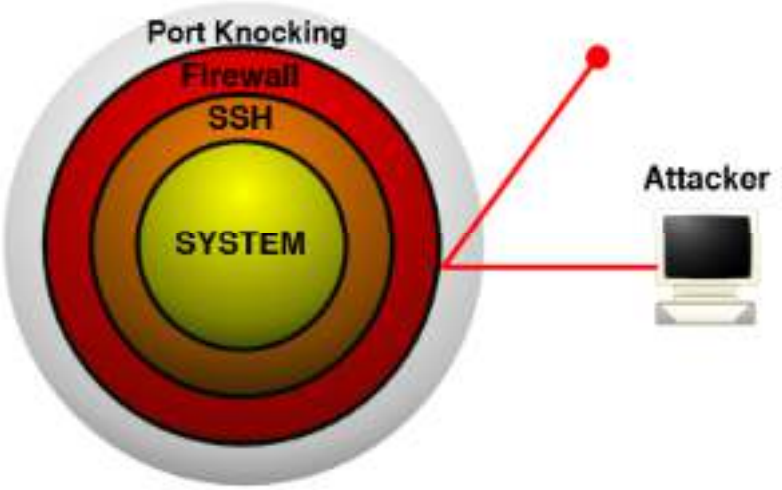
Tujuan berikutnya dari metode *port knocking* adalah untuk menyembunyikan dari calon penyerang saat *scanning* kepada jaringan yang berpotensi *exploitable* atas layanan yang diberikan. *Port* yang tidak boleh terbuka seperti FTP (21), SSH (22), Telnet (23) dan lainnya. Namun dari sisi administrator jaringan, tetap dapat melakukan konfigurasi dan monitoring, akan tetapi dengan langkah-langkah khusus (*knocking*) agar dapat diijinkan oleh *firewall* untuk akses ke *port* FTP, SSH dan lainnya.

2. METODE PENELITIAN

Penelitian dilakukan dengan mempelajari saat aliran paket data koneksi dari komputer *client*, dan router yang telah berlaku juga sebagai firewall, dimana port tertentu sudah ditutup. Kemudian dilanjutkan dengan mengamati proses *knocking* dan cara kerjanya.
Pengujian dilakukan pada komputer *client* dengan cara mengakses ke suatu perangkat router dalam hal ini adalah Router Mikrotik, prinsip dasar dari *port knocking* secara mudah dapat dijelaskan sebagai berikut :

- *Client* melakukan koneksi ke sistem remote yang menerapkan aturan *firewall*.
- *Client* sama sekali tidak dapat terkoneksi dengan port berapapun pada *remote* sistem, dengan kata lain semua *port* ditutup oleh *firewall.*
- *Client* mencoba melakukan koneksi dengan mengirimkan paket data ke sistem remote melalui beberapa port secara sekuensial dalam hal ini antara port FTP dan SSH yang tersedia, *client* tidak akan mendapatkan response apapun dari server yang melayani port tersebut.
- Melakukan *Port knocking* dengan mencatat percobaan koneksi, kemudian melakukan autentikasi terhadap percobaan tersebut dan bila autentikasi berhasil, dalam hal ini urutan port yang di coba untuk dikoneksikan sesuai dengan aturan tertentu pada *port knock*,ing maka akan melakukan perubahan terhadap file konfigurasi *firewall* agar mengijinkan *port* n yang dimaksudkan untuk dibuka kepada *client* dengan IP Addres ter-autentikasi dan diberikan waktu akses yang telah ditentukan.

Client melakukan koneksi ke *port* n menggunakan aplikasi seperti pada umumnya dan akan terputus dengan waktu yang telah ditentukan dan port akan tertutup kembali.



Gambar 1. Arsitektur Port Knocking pada jaringan dengan kondisi port tertutup Firewall

e-ISSN : 2828-4518
p-ISSN : 2828-3694

3. PEMBAHASAN dan HASIL

3.1. Implementasi pada Jaringan

Sistem terdiri dari komputer client dan router Mikrotik yang berfungsi sebagai firewall dengan menutup semua port. Komputer client akan mencoba untuk membuat koneksi dengan layanan FTP (port 21), maka firewall akan menolak atas akses layanan. Berikut ini yang ditunjukan oleh gambar 2.

Gambar 2. Fungsi Firewall secara sederhana menolak akses layanan

3.2. Firewall

Firewall adalah istilah yang biasa digunakan untuk menunjuk pada suatu komponen atau sekumpulan komponen jaringan, yang berfungsi membatasi akses antara dua jaringan, lebih khusus lagi, antara jaringan internal dengan jaringan global internet.
Tugas firewall antara lain :

1. Harus dapat mengimplementasikan kebijakan security dijaringan (site security policy). Jika aksi tertentu tidak diperbolehkan oleh kebijakan ini, maka firewall harus meyakinkan bahwa semua usaha yang mewakili operasi tersebut harus gagal atau digagalkan. Dengan demikian, semua akses ilegal antarjaringan (tidak diotorisasikan) akan ditolak.
2. Melakukan filtering: mewajibkan semua traffik yang ada untuk dilewatkan melalui firewall bagi semua proses pemberian dan pemanfaatan layanan informasi. Dalam konteks ini, aliran paket data dari/menuju firewall, diseleksi berdasarkan IP address, nomor port, atau arahnya, dan disesuaikan dengan kebijakan security.
3. Firewall juga harus dapat merekam/mencatat even even mencurigakan serta memberitahu administrator terhadap segala usaha usaha menembus kebijakan security.

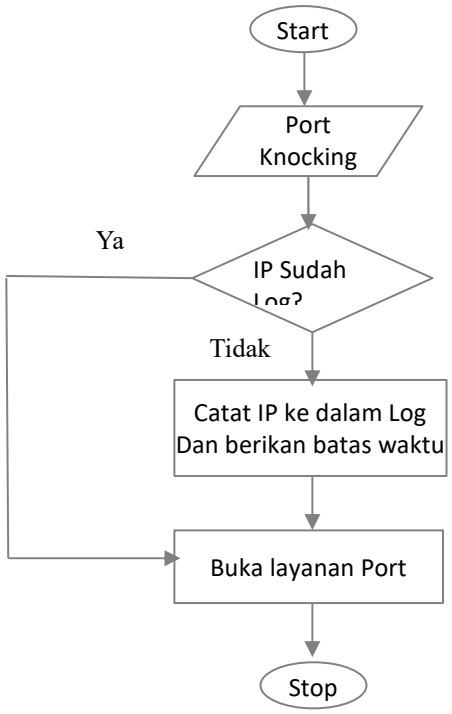
Gambar 3. Firewall

e-ISSN : 2828-4518
p-ISSN : 2828-3694

3.3 Algoritma Port Knocking

Agar dapat diimplementasikan, maka penulis memetakan sebagai berikut, sistem ini terdiri dari *client* dan perangkat *Miktorik* dengan fungsi *firewall*, dimana *client* mengirimkan paket *ICMP* sebagai bentuk *Port Knocking* kepada *firewall*. Saat paket diterima oleh *firewall*, akan melakukan pencatatan alamat IP tersebut, kemudian *firewall* memberikan batas waktu akses. *Client* dapat segera melakukan koneksi yang dimaksudkan seperti *FTP* melewati *firewall*. *Firewall* akan memeriksa *log* pencatatan IP sebelumnya, apabila *valid* dapat diteruskan *port* tersebut ke dalam jaringan, namun bila sebaliknya *firewall* akan menolak (*drop*) paket permintaan tersebut.

Berikut ini digambarkan algoritma dari sistem port knocking yang dimaksudkan :



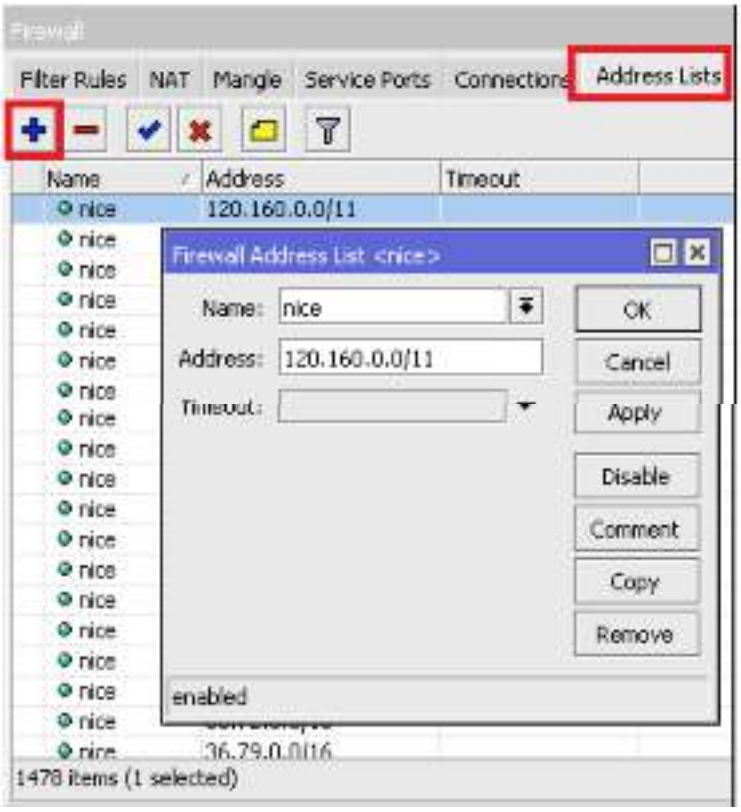
Gambar 4. Algoritma Port Knocking

3.4. Konfigurasi Router Mikrotik

Untuk dapat menerapkan metode *port knocking* penelitian dilakukan pada perangkat router *Mikrotik*, dapat dilakukan sebagai berikut :

1. Pada *Mikrotik* dengan memanfaatkan *Address-List*, dimana fitur ini dapat digunakan untuk melakukan pengelompokan / grouping *IP address* yang selanjutnya dapat digunakan pada fitur lain seperti *firewall filter*, *NAT* atau *mangle*. Pembuatan *address-list* dapat dilakukan dengan memilih menu IP → Firewall → Address List, dimana IP yang telah ditentukan yang dapat melakukan koneksi port. Atau dapat juga dibuatkan address list yang nantinya secara dinamis akan disimpan / log ke dalam firewall dimana biasanya diterapkan jika IP yang akan dimasukan ke dalam group *address list* belum diketahui sebelumnya atau memang sifatnya berubah-ubah. Dalam contoh *setting port knocking* ini, akan digunakan *dynamic address list* tersebut.

e-ISSN : 2828-4518
p-ISSN : 2828-3694



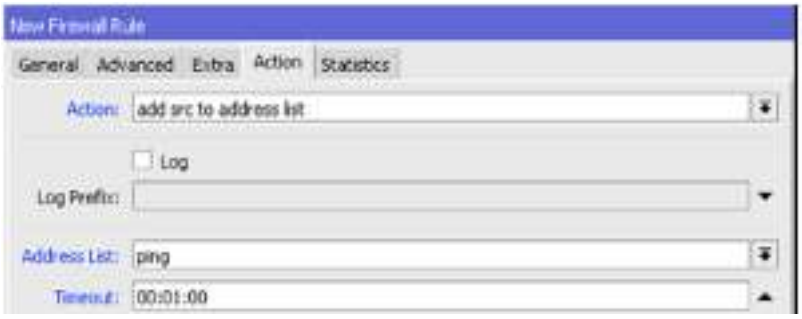
Gambar 5. Setting Address list pada Mikrotik

2. Pada konfigurasi ini, penulis akan melakukan *port knocking* bagaimana *client* dapat melakukan *knock ICMP* terlebih dulu. Cara kerjanya yaitu dengan memasukan IP *adrress client* yang mengirimkan paket ICMP ke router *firewall* ke dalam sebuah *address list* secara otomatis. Setelahnya, hanya IP yang sudah terdaftar pada *address list* yang dapat akses ke layanan FTP.
3. Untuk melakukan *grouping* IP secara otomatis dapat menggunakan fitur firewall filter. Dimana dilakukan konfigurasi *matcher firewall*. Dapat dispesifikasikan hanya trafik ICMP ke router yang akan ditangkap.

Gambar 6. Setting menangkap paket ICMP

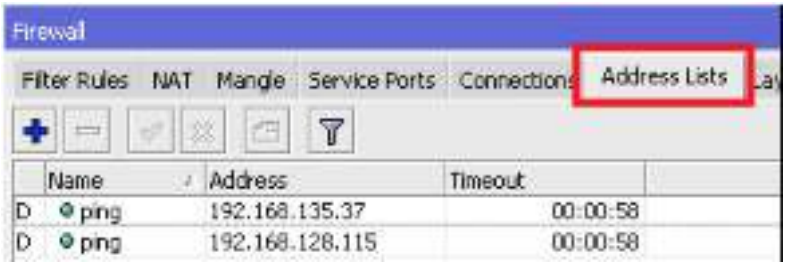
4. Setelah itu, gunakan action=add-src-to-address-list untuk memasukkan IP *address user* yang melakukan *ping* ke router ke dalam sebuah group. Nama group dapat didefiniskan pada parameter *address list*, agar IP tersebut tidak selamanya ada dalam daftar group, maka definisikan parameter *timeout*.

e-ISSN : 2828-4518
p-ISSN : 2828-3694



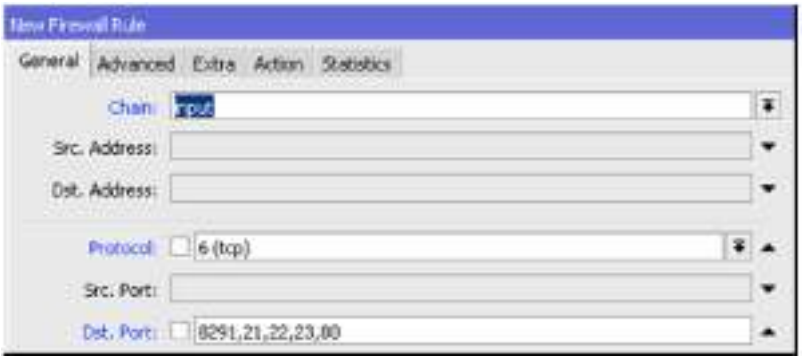
Gambar 7. Setting menangkap IP address dan diberikan batas waktu

5. Sampai langkah ini, jika ada *client* yang melakukan ping ke router maka IP *client* tersebut akan dimasukan ke dalam *address list* dengan nama = ping.



Gambar 8. Hasil tangkapan dari paket ICMP

6. Perbedaan *address list* yang ditambahkan secara otomatis terletak pada flag "D" didepannya. Karena sebelumnya *timeout* ditentukan maka IP *address* tersebut akan dihapus otomatis dari daftar pada saat *timeout* telah habis.
7. Langkah selanjutnya yang harus dilakukan adalah membuat *rule firewall filter* untuk melakukan *blocking* akses kepada *port* yang ditujukan selain dari IP *address* yang sudah masuk dalam daftar *address list*.



Gambar 9. Block port selain dari IP yang masuk daftar list

8. Selanjutnya dapat dispesifikasikan *scr-address* dari paket data yang akan ditangkap, untuk kasus ini dapat menggunakan nama *address list* yang sebelumnya ditambahkan secara otomatis. Karena yang akan ditangkap adalah trafik data selain IP yang sudah terdaftar maka dapat menggunakan logika *NOT* (!).

e-ISSN : 2828-4518
p-ISSN : 2828-3694

Gambar 10. Block pelayanan yang tidak terdaftar

9. Langkah terakhir adalah penentuan aksi. Untuk tujuan blocking digunakan action=drop.

Gambar 11. Aksi Drop

10. Apabila dilihat secara keseluruhan *rule firewall filter* yang dibuat menjadi seperti berikut :

Gambar 12. *Rule firewall* secara keseluruhan

3.4 Uji Coba

Dalam uji coba ini, penulis akan menjalankan semua fungsi yang ada pada router yang telah terkonfigurasi seperti diatas dan berjalan sebagai *firewall*. Untuk menjalankan ujicoba penulis melakukan hal-hal berikut :

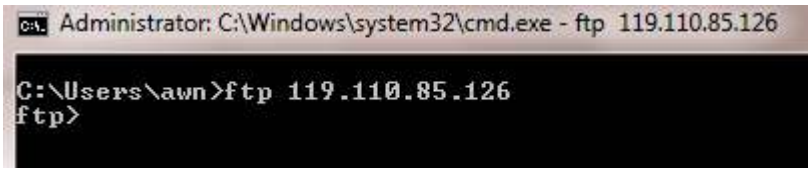
1. Menyediakan perangkat *Mikrotik* dengan konfigurasi menutup semua *port*.
2. Menunggu usuha koneksi yang dialamatkan kepadanya dengan memonitor *interface* yang ada.
3. Memeriksa apakah ada *knocking* dengan paket *ICMP*.
4. Bila sesuai dengan aturan akan dicatat ke log group IP yang telah disediakan.
5. Membuka *port* yang telah ditentukan.

Data sebelum pengujian dapat ditampilkan pada tabel 1 :

No	Nama	IP
1	Komputer *Client*	202.62.16.122
2	Router berlaku sebagai *firewall*	119.110.85.126

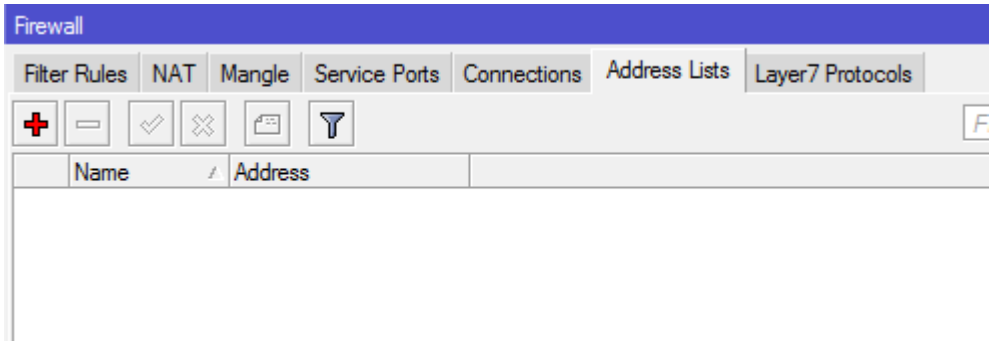
Pengujian dilakukan dengan mengamati komputer client dan router *Mikrotik* yang berfungsi *firewall*, seperti terlihat pada gambar 13 dibawah ini. Dimana saat belum melakukan *port knocking* kepada *firewall*. Komputer client berusaha untuk mengkoneksikan diri dengan FTP dan paket langsung di tolak oleh *firewall*.

e-ISSN : 2828-4518
p-ISSN : 2828-3694



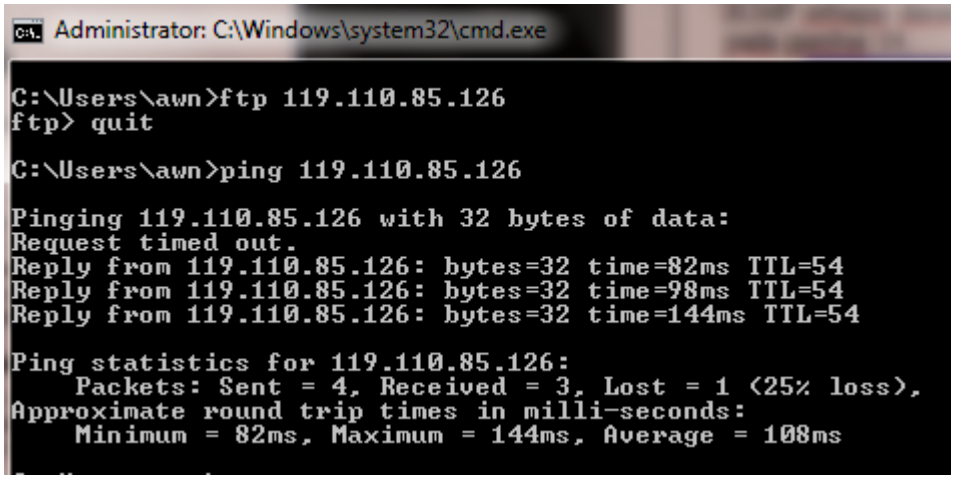
Gambar 13. Komputer *client* tidak dapat melakukan koneksi FTP

Kemudian dapat dilanjutkan dengan tampilan pada *firewall* tidak adanya tangkapan paket data ICMP sebagai dasar untuk memberikan izin membuka *port* layanan FTP. Berikut ini dapat ditunjukan pada gambar 14.



Gambar 14. Log pada firewall masih kosong

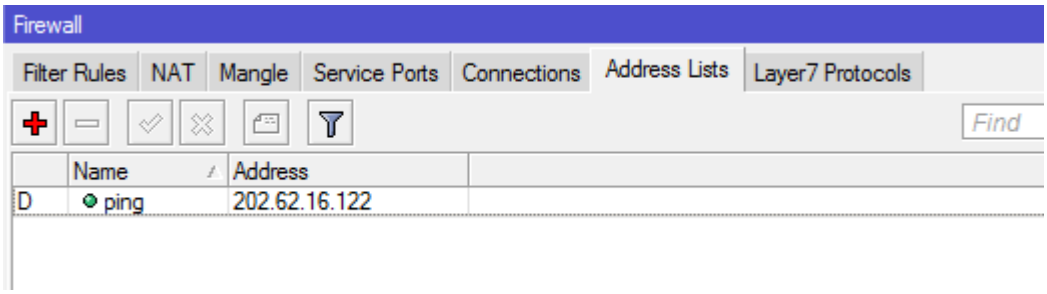
Dilanjutkan dengan komputer *client* melakukan *port knocking*, seperti pada tampilan gambar 15.



Gambar 15. Komputer *client* berhasil ping (ICMP) kepada router

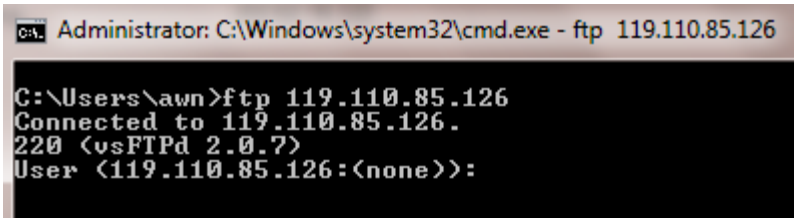
Maka pada router *Mikrotik* akan terlog dan memasukan ke dalam group IP yang diberi nama ping dan hanya bertahan selama 1 menit (dapat diatur lebih lanjut apabila diperlukan). Berikut gambar 16.

e-ISSN : 2828-4518
p-ISSN : 2828-3694



Gambar 16 Router Mikrotik dapat menangkap paket ICMP dan disimpan pada tabel group IP ping

Kemudian komputer client sudah dapat melanjutkan koneksi layanan yang diperlukan, dalam hal ini adalah layanan FTP, seperti yang ditunjukan pada gambar 17, komputer client sudah diizinkan koneksi.



Gambar 17 Komputer client sudah dapat terkoneksi pada layanan FTP

5. KESIMPULAN

Dari hasil pengamatan dan pengujian oleh penulis, dapat diambil kesimpulan berikut :

1. Metode *port knocking* dapat menjadi sebuah *security layer* tambahan pada suatu *firewall*.
2. Memungkinkan Administrator Jaringan melakukan koneksi kepada server meskipun *firewall* memblock semua *port* yang ada.
3. Pemeriksaan IP yang telah ditentukan sebelumnya dan dibandingkan kembali dengan IP hasil dari proses *port knocking*, apabila sama dapat diberikan akses lebih lanjut.
4. Dapat ditambahkan pemeriksaan lebih lanjut agar keamanan *firewall* lebih baik seperti enkripsi paket dan pemeriksaan *username* dan *password*.

6. REFERENSI

Donald A. Tevault 2020, Mastering Linux Security and Hardening : Protect your Linux Systems from intruders, malware attacks and other cyber threats, edition 2

Wale Soyinka 2020, Linux Administration: A Beginner's Guide, eighth Edition: Edition 8

Nathan Gusti Ryan 2018, Best Practice Linux Server Administrator Seri Linux CentOS 6 & 7