ISSN: 2302 -2805

IMPLEMENTASI *PROXY SEVER* MENGGUNAKAN DHCP *SERVER* BERBASIS LINUX UBUNTU PADA JARINGAN INTERNET SEBAGAI *FILTER* DAN *SECURITY*

Seto Febriantoro [1)], Agus Ganda Permana [2)], Tengku A Riza [3)]
[1,2,3)] *Fak Elektro & Komunikasi IT Telkom Bandung*
Jl. Telekomunikasi No.1 Ters Buah Batu Bandung
email: setoittelkom@gmail.com [1)], agd@ittelkom.ac.id [2)], tka@ittelkom.ac.id [3)]

Abstrak

Konten didalam dunia internet pun begitu beragam, dari web informasi sampai social network tapi dalam kenyataannya situs-situs porno pun masih menjadi top list bagi pengguna internet. Selain konten di internet yang semakin beragam hacker pun sekarang berkembang pesat, hal ini dikarenakan semakin banyak informasi mengenai metode-metode dalam melakukan serangan yang dengan mudah di dapatkan di situs-situs informasi di dunia internet.

Untuk memblokir situs yang dianggap terlarang maka perlu diperlukan sebuah Proxy Server . Dengan adanya Proxy Server tersebut dapat memberikan filter pada system jaringan internet kita. Ini akan lebih memberikan kenyamanan sebagai user dalam menggunakan internet. Disamping fungsi server berjalan dengan baik perlu adanya sistem security yang handal, firewall adalah kuncinya yaitu dengan menambahkan beberapa rules di firewall bisa membuat sistem security di server semakin handal, banyak aplikasi firewall yang bisa di gunakan salah satunya UFW (Uncomplicated Firewall).

Pada penelitian ini performansi server memiliki prosentase keberhasilan sangat tinggi yaitu 93.30%, hal ini dikarenakan fungsi server sebagai proxy server berjalan dengan baik, dari 3 parameter yang diuji dengan mengambil beberapa sample yaitu keywords, url situs dan ext .3gp, hanya ext .3gp yang memiliki prosentase keberhasilan kurang dari 100% yaitu 80%, selain filtering yang berfungsi dengan baik sistem security dengan 2 metode pun berhasil meminimalisir serangan hacker (ICMP Flooding dan Root Compromise) dan dengan melakukan monitoring server berhasil melacak IP dari hacker tersebut.

Kata kunci :
Proxy Server, DHCP Server, Security, Firewall, UFW

1. Pendahuluan

Perkembangan teknologi saat ini telah memberikan berbagai kemudahan kepada manusia dalam banyak hal. Salah satunya untuk bisa mendapatkan informasi secara cepat dan akurat. Internet adalah salah satu cara untuk mendapatkan kemudahan informasi. Namun disinilah sebuah unsur *cyber crime* bisa dengan mudah pula terjadi. Seperti pada sebuah *cafe* atau tempat perbelanjaan yang memberikan fasilitas area *hotspot* dan akses *wifi* internet yang didukung oleh beberapa *access point*. Disini tentu saja merupakan sebuah kesempatan bagi seseorang yang ingin mengambil atau bahkan merusak sistem yang ada didalamnya. Maka dari itu dibutuhkan sebuah sistem kemanan yang handal untuk membentengi sistem tersebut.

Dewasa ini akses situs porno di Indonesia meningkat dengan tajam, bukan saja orang dewasa yang mengakses situs terlarang tersebut bahkan sekarang anak-anak kecil tingkatan SMP sudah pintar mengakses situs tersebut. Kenapa mereka mengakses karena mereka diberikan kebebasan, dengan cukup mengetikkan kata-kata yang berbau pornografi maka dalam hitungan detik semua situs tersebut terpampang di depan mata. Akankah orangtua membiarkan hal tesebut dilakukan oleh anak-anak mereka ?. Sistem keamanan yang dapat memberikan kenyamanan pada sistem tersebut antara lain dengan memberikan *filter* atau dalam dunia *security* dikenal dengan *firewall*. Dalam hal ini diperlukan suatu system yang dinamakan *Proxy Server* yang berfungsi sebagai *filter* terhadap situs-situs pada dunia internet.

2. Tinjauan Pustaka

2.1 Security

Pada era global ini, keamanan sistem informasi berbasis Internet harus sangat diperhatikan, karena jaringan komputer Internet yang sifatnya publik dan global pada dasarnya tidak aman. Terutama pada sisi penyedia layanan itu sendiri dalam hal ini *server*,karena apabila tingkat keamanan *server* tidak tinggi akan berakibat tidak berfungsinya layanan dikarenakan serangan dari para *hacker*.

Apabila ingin mengamankan suatu jaringan maka harus ditentukan terlebih dahulu tingkat ancaman (*threat*) yang harus diatasi, dan resiko yang harus diambil maupun yang harus dihindari. Berikut ini akan dibahas mengenai ancaman dan metode pencegahannya .

2.1.1 Serangan

Pada dasarnya, ancaman datang dari seseorang yang mempunyai keinginan memperoleh akses ilegal ke dalam suatu jaringan komputer. Oleh karena itu, harus ditentukan siapa saja yang diperbolehkan mempunyai akses legal ke dalam sistem, dan ancaman-ancaman yang dapat mereka timbulkan. Ada beberapa tujuan

yang ingin dicapai oleh penyusup dan, sangat berguna apabila dapat membedakan tujuan-tujuan tersebut pada saat merencanakan sistem keamanan jaringan komputer. Beberapa tujuan para penyusup adalah:

1. Pada dasarnya hanya ingin tahu sistem dan data yang ada pada suatu jaringan komputer yang dijadikan sasaran.
2. Membuat sistem jaringan menjadi *down*.
3. Berusaha untuk menggunakan sumber daya di dalam sistem jaringan komputer.
4. Ingin tahu data apa saja yang ada di dalam jaringan komputer.

2.1.2 Jenis – jenis serangan

A. Probe

Sebuah *probe* dapat dikenali dari adanya usaha-usaha yang tidak lazim untuk memperoleh akses ke dalam suatu sistem atau untuk menemukan informasi tentang sistem tersebut. Salah satu contohnya adalah usaha untuk *login* ke dalam sebuah *account* yang tidak digunakan. *Probing* ini dapat dianalogikan sebagai usaha untuk memasuki sebuah ruangan yang dengan mencoba-coba apakah pintunya terkunci apa tidak.

B. Scan

Scan adalah kegiatan *probe* dalam jumlah yang besar dengan menggunakan *tool* secara otomatis. *Tool* tersebut secara otomatis dapat mengetahui *port-port* yang terbuka pada *host* lokal maupun *host remote*, IP *address* yang aktif, bahkan bisa untuk mengetahui sistem operasi yang digunakan pada host yang dituju.

C. Account Compromise

Account compromise adalah penggunaan *account* sebuah komputer secara ilegal oleh seseorang yang bukan pemilik *acoount* tersebut. *Account compromise* dapat mengakibatkan korban mengalami kehilangan atau kerusakan data. Sebuah insiden *account compromise* dapat berakibat lebih lanjut, yaitu terjadinya insiden *root compromise*, yang dapat menyebabkan kerusakan lebih besar.

D. Root Compromise

Root compromise mirip dengan *account compromise*, dengan perbedaan *account* yang digunakan secara ilegal adalah *account* yang mempunyai *privilege* sebagai *administrator* sistem. Istilah *root* diturunkan dari sebuah *account* pada sistem berbasis UNIX yang mempunyai *privelege* tidak terbatas.

E. Packet Sniffer

Packet Sniffer adalah suatu *device*, baik perangkat lunak maupun perangkat keras yang digunakan untuk memperoleh informasi yang melewati jaringan komputer. Kegunaan dari paket *sniffer* adalah membuat NIC (*Network Interface Card*), contohnya *Ethernet*, dalam mode *promiscuous* sehingga dapat menangkap semua trafik dalam jaringan. *Mode promiscuous* adalah *mode* di mana semua *workstation* pada jaringan komputer "mendengar" semua trafik, tidak hanya trafik yang dialamatkan ke *workstation* itu sendiri. Jadi *workstation* pada mode *promiscuous* dapat "mendengarkan" trafik dalam jaringan yang dialamatkan kepada *workstation* lain.

F. *Denial Of Service* (Dos)

DoS merupakan serangan yang cukup menakutkan di dunia internet karena akibat dari serangan ini server akan mati dan tidak dapat beroperasi lagi sehingga otomatis tidak dapat meberikan pelayanan lagi. Dalam sebuah serangan *Denial of Service*, si penyerang akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut:

1. Membanjiri lalu lintas jaringan dengan banyak data sehingga lalu lintas jaringan yang datang dari pengguna yang terdaftar menjadi tidak dapat masuk ke dalam sistem jaringan.
2. Membanjiri jaringan dengan banyak *request* terhadap sebuah layanan jaringan yang dibedakan oleh sebuah *host* sehingga *request* yang datang dari pengguna terdaftar tidak dapat dilayani oleh layanan tersebut.
3. Mengganggu komunikasi antara sebuah *host* dan kliennya yang terdaftar dengan menggunakan banyak cara, termasuk dengan mengubah informasi konfigurasi sistem atau bahkan perusakan fisik terhadap komponen dan *server*.

Security Policy

Security Policy menyediakan kerangka-kerangkan untuk membuat keputusan yang spesifik, misalnya mekanisme apa yang akan digunakan untuk melindungi jaringan. *Security Policy* juga merupakan dasar untuk mengembangkan petunjuk pemrograman yang aman untuk diikuti user maupun bagi administrator sistem.

Faktor-faktor yang berpengaruh terhadap keberhasilan *Security Policy* antara lain adalah:

- Komitmen dari pengelola jaringan.
- Dukungan teknologi untuk menerapkan *security policy* tersebut.
- Kesadaran semua user terhadap keamanan jaringan.

Teknik-teknik yang dapat digunakan untuk mendukung keamanan jaringan antara lain:

a. Authentikasi terhadap sistem.
b. Enkripsi terhadap sistem untuk penyimpanan dan pengiriman data penting.
c. *Tool-tool* jaringan, misalnya *firewall* dan *proxy*.

2.2 Linux Ubuntu

Ubuntu adalah suatu *Operating System* (OS) dari *distro Linux*. Sifatnya *Open Source* yang artinya semua orang bisa mengembangkan tanpa terpusat oleh suatu pengembang[2]. Keuntungan lainnya dari ubuntu ini adalah gratis dan langsung bisa digunakan tanpa harus mengaktivasi terlebih dahulu.

Dalam Ubuntu ada istilah yang dikenal dengan terminal. Terminal sering disebut *command prompt* atau *shell*. Di masa lalu, hal ini adalah cara pengguna untuk berinteraksi dengan komputer, dan para penggunanya berpendapat bahwa penggunaan perintah melalui *shell*

ISSN: 2302 -2805

akan lebih cepat dibanding melalui aplikasi berbasis grafik dan hal ini masih berlaku sampai sekarang[2].

Salah satu keunggulan lagi dari Ubuntu ini adalah adanya *repository*. *Repository* adalah tempat penyimpanan kumpulan *software*/aplikasi yang bisa di*download* untuk digunakan[2], karena pada dasarnya setiap menginstall *software* di Linux dibutuhkan *repository*, maka Linux akan mencari *software* tersebut di *repository*. Jika telah ditemukan maka proses instalasi bisa dilanjutkan.

2.3 *Proxy Server*

Proxy server adalah suatu *server* yang bertindak sebagai perantara dalam pengaksesan permintaan dari suatu *client* ke suatu *server*[2].
Tujuan adanya *Proxy Server* sebagai berikut[2].

1. Menambah kecepatan *web-surfing* (melalui *cache*).
2. Menghemat *bandwidth.*
3. Untuk *Filtering* (Memblokir situs-situs yang di larang ditempat tempat seperti sekolah, kantor dan sebagainya).

Dalam implementasi *system proxy* digunakan suatu aplikasi yaitu *squid proxy*. Secara umum dalam *squid* terdapat 3 parameter penting.

A. Access Control List

Digunakan untuk memberikan akses ataupun sebaliknya kepada IP tertentu untuk mengakses layanan pada *server*.

B. Filtering

Merupakan bagian terpenting dari *squid*, dalam mengatur *rule* pengaksesan internet apakah *web* tersebut di izinkan untuk diakses atau tidak.

C. Bandwidth Limiter

Memberikan batasan *download* atau kecepatan *download* dalam suatu jaringan.

Cara kerja *Proxy Server* (*squid*) sebagai berikut.

1. Klien *merequest web page via browser* yang sudah terkoneksi dengan *proxy.*
2. *Proxy server* menerima *request* dari klien.
3. *Proxy server* melakukan autentikasi *request web page* dari klien.
4. Jika *web page* tersebut mendapat *access* langsung *proxy* mengirim *request* ke *web server* yang dituju klien.
5. *Web server* memberikan balasan layanan ke *proxy* dan diteruskan *proxy* ke klien yang melakukan *request.*

2.4 *Firewall*[4]

Firewall adalah sebuah sistem atau perangkat yang mengizinkan lalu lintas jaringan yang dianggap aman untuk melaluinya dan mencegah lalu lintas jaringan yang tidak aman. Salah satu aplikasi dari *firewall* pada ubuntu adalah UFW (*Uncomplicated Firewall*).

UFW merupakan tool front-end yang secara *default* ter*install* pada ubuntu. Ubuntu yang mengintegrasikan UFW ke dalam paket instalasinya dimulai dari Ubuntu 8.04 sampai saat ini, jadi UFW ini tidak akan ditemui pada ubuntu versi 7.10 ke bawah.

UFW lebih mudah digunakan dari pada iptables karena menggunakan *command-line* yang sangat sederhana dan mudah di ingat sehingga dalam mengkonfigurasi *firewall* menjadi lebih mudah dan lebih aman bagi *user* awam[5].

2.5 *IPtraf*

Iptraff adalah *tool* jaringan berbasis konsol yang ada di linux. *Tool* ini berfungsi untuk mengumpulkan informasi seperti koneksi TCP berupa paket, jumlah *byte* yang diterima,statistik *interface* dan indikator aktivitas jaringan dan sebagainya.

5. Metode Penelitian

Dalam pembuatan penelitian ini dibutuhkan langkah-langkah sebagai berikut :

1. Perancangan Sistem
   Perancangan sistem tersebut berdasarkan pada teori tentang *proxy serve*r dan *security server*.
2. Perancangan dan Pengenalan Perangkat Lunak (*Software*)
   Perancangan dan pengenalan *software* meliputi *software IPtraf, Putty* dan juga s*quid proxy.*
3. Pembuatan dan Pengujian Perangkat Lunak (*Software*)

Setelah merancang perangkat lunak, kemudian proses pembuatan pertama meliputi proses instalisasi *linux, DHCP dan squid proxy*. Proses pembuatan kedua meliputi konfigurasi *DHCP, UFW SSH dan squid.*

6. Pengujian Sistem
   Setelah semua program dibuat, proses pengujiannya yaitu dengan menjalankan *web browser* untuk menguji *filtering* dan melakukan serangan ke server dengan *command promt dan putty*.
7. Analisa Sistem
   Kemudian kita dapat menganalisa sistem tersebut dengan mencari kekurangan- kekurangan yang ada supaya kita bisa memperbaiki kekurangan-kekurangan tersebut.

4. Hasil dan Pembahasan

Pengujian ini bertujuan untuk memastikan bahwa sistem yang dibangun dapat memenuhi tujuannya. Pengujian dilakukan antara *Client-Server*. Sesuai dengan skenario, pengujian difokuskan pada jaringan internal.

Skenario yang pertama *client* melakukan *request* ke *server* untuk masuk ke layanan internet, dalam hal ini *client* mendapatkan balasan layanan sesuai *rule* yang telah di konfigurasi pada sisi *proxy server* yaitu adanya *filtering* situs-situs porno dan pembatasan *download* untuk ext .3gp.

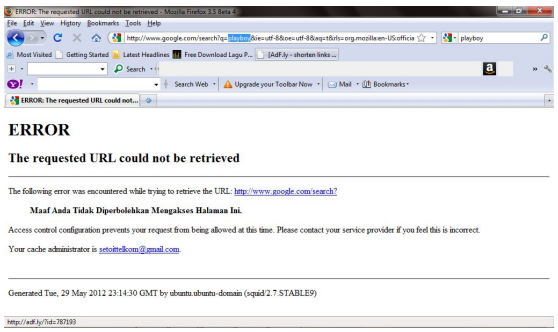
Skenario yang kedua *client* bertindak seakan-akan sebagai *hacker* yang mencoba melakukan serangan ke *server*, dalam hal ini ada dua metode yang digunakan yaitu *client* melakukan ICMP *flooding* dan *Root compromise* dengan memanfaatkan *openssh server*.

4.1 Skenario Pengujian Sistem

4.1.1 *Filtering* situs porno dan pembatasan *download* ext .3gp

Filtering situs porno

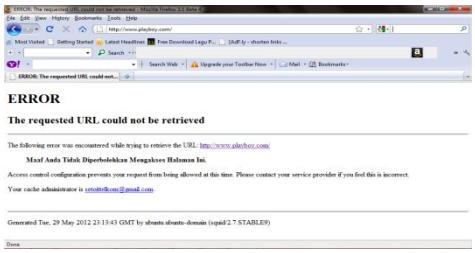
Client melakukan *request* layanan untuk mengakses alamat *web* yang dituju (dalam kasus ini contohnya *playboy*) melalui *search engine* (dalam kasus ini contohnya *google*).



Gambar 1 Mencoba Mengakses situs porno

Berdasarkan Gambar 4.1 *client* tidak bisa mengakses *playboy* dikarenakan *rule* yang ada di *server* melakukan pemblokiran terhadap *keyword* porno sebagai contohnya *keyword playboy*.

Client melakukan *request* layanan untuk mengakses alamat *web* yang dituju (dalam kasus ini contohnya *playboy*) melalui url www.playboy.com.



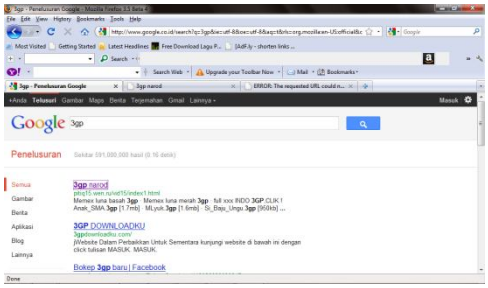
Gambar 2 Mencoba mengakses situs porno dengan url

Berdasarkan Gambar 4.2 *client* tidak bisa mengakses *playboy* dikarenakan *rule* yang ada di *server* melakukan pemblokiran terhadap url porno sebagai contohnya www.playboy.com.

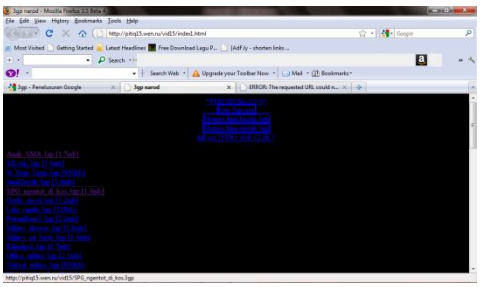
Pembatasan *download* ext .3gp

Di dunia internet terkadang ada beberapa portal yang memiliki nama bersifat normal atau tidak mengandung kata porno tapi didalamnya terdapat penyimpanan video porno (kebanyakan ada di xxx.blogspot.com dan ber ext .3gp).

Pada pengujian kali ini *client* memanfaatkan *search engine* (dalam kasus ini contohnya *google*) untuk mencari portal penyedia video porno (dalam kasus kebanyakan ext .3gp).



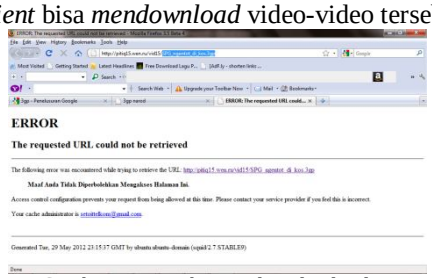
Gambar 3 Mengakses memanfaatkan ext .3gp



Gambar 4 Url http://pitiq15.wen.ru/vid15/index1.htm

Berdasarkan Gambar 4.4 url yang tertera adalah http://pitiq15.wen.ru/vid15/index1.htm

Dalam url tersebut tidak terdapat *keyword* yang bersifat porno, jika tidak di beri *rule* pembatasan *download* .3gp maka *client* bisa *mendownload* video-video tersebut.



Gambar 5 Mencoba mendownload video

Berdasarkan Gambar 4.5 *client* tidak bisa melakukan *download* ke link yang dituju karena di sisi *server* sudah ada *rule* pembatasan *download* ext .3gp.

4.1.2 Sistem *security server* dengan metode ICMP *flooding* dan *Root compromise*

ICMP *flooding*

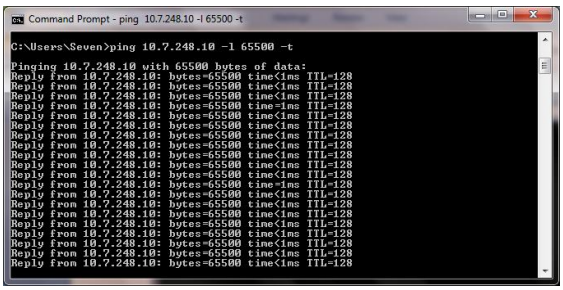
Penyerang melakukan *ping* dengan *byte* besar agar memenuhi *bandwidth* dan membanjiri jaringan dengan *request* pengiriman paket besar sekaligus, dalam kasus ini jika *ping* dilakukan secara terus menerus dan dalam *byte* yang besar akan di *reject* tapi jika dengan *ping byte* normal akan di *accept*.

Penyerang: *Client* (IP Address: 10.7.248.10)
Tujuan : *Server* (IP Address: 10.7.248.120)

ISSN: 2302 -2805

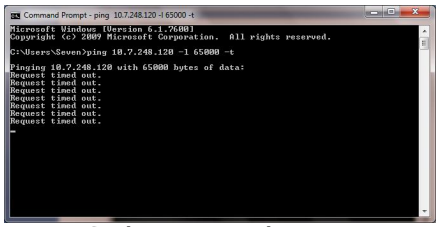
Berdasarkan Gambar 4.6 *client* melakukan *ping* dengan *byte* normal (32 bytes) maka akan mendapatkan balasan dari *server.*

Berikut *ping* keadaan intrusi dengan 65500 *bytes* secara terus menerus dari komputer *client*:



Gambar 6 Melakukan *ping*

Berdasarkan kasus gambar 4.7 *server* menggunakan *firewall* UFW (*Uncomplicated Firewall*) untuk melakukan *reject* terhadap serangan *Ping* yang berlebihan (*ICMP Flooding*) :



Gambar 7 Menggunakan UFW

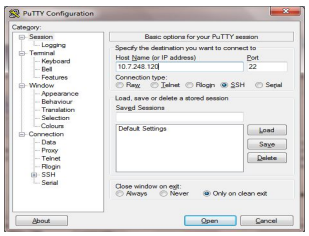
Berdasarkan Gambar 4.8 sistem *server* dengan *firewall* ufw berhasil melakukan *reject ping* dengan skala besar dan terus menerus.

Root compromise

Penyerang: *Client* (IP Address: 10.7.248.10) *Login putty* : imas

Tujuan : *Server* (IP Address: 10.7.248.120)

Client berusaha masuk ke sistem *root server* melalui *openssh server*, *openssh* sendiri memiliki *default port* 22 yang berfungsi sebagai jalur *meremote server* dari jauh, dalam kasus ini penyerang menggunakan *software Putty* untuk masuk ke sistem *root* .



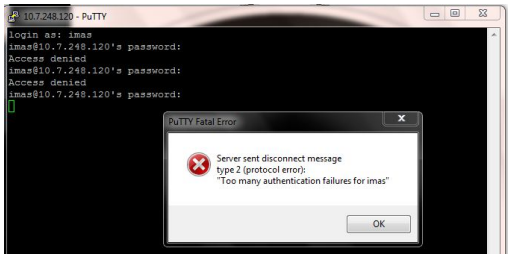
Gambar 8 Mencoba Openssh

Gambar 9 Mencoba memasukkan *password*

Gambar 4.10 merupakan proses ketika *client* berhasil masuk ke *interface root server.*

Dalam proses selanjutnya penyerang harus mengetahui *password server* agar bisa masuk ke sistem *root server* itu sendiri, jika penyerang ini tidak tahu *password server* maka mereka menggunakan metode *brute force login* yaitu melakukan kombinasi kata untuk menyamakan *password.*

Di proses inilah *server* harus pintar, harus memiliki *rule* untuk memberikan limit waktu untuk *Putty* dan limit jumlah melakukan authentikasi *password*, dalam kasus ini limit untuk waktu *standby putty* 60s dan max authentikasi 3 kali.



Gambar 10 Melakukan *Putty* lebih dari 3 kali

Berdasarkan gambar 4.11 terlihat jika penyerang tidak bisa masuk ke sistem *root* dikarenakan melakukan *brute force login* sebanyak 2 kali dan langsung ada *message* pemberitahuan dari *server.*

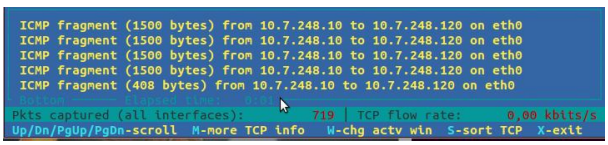
4.2.3 Monitoring serangan client

Dalam penelitian ini sistem monitoring lebih di tekankan pada serangan yang masuk ke sistem *server*, dalam kasus contohnya ICMP *flooding* dan *Root compromise.*

Monitoring ICMP flooding

Dalam melakukan *monitoring ICMP flooding* pada kasus ini *server* menggunakan aplikasi IPtraf.

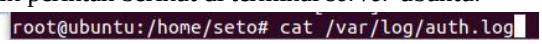
Berikut *ping* keadaan intrusi dengan (65500) *bytes* dari komputer *client*



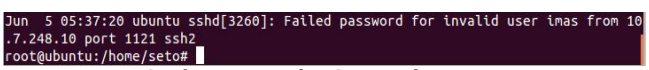
Gambar 11 Monitoring ICMP *flooding*

Monitoring *Root compromise*

Ketik perintah berikut di terminal *server* ubuntu:



Tampilan catatan yang tersimpan di sistem *root* sebagai berikut :



Gambar 12 Tampilan Commnad Line

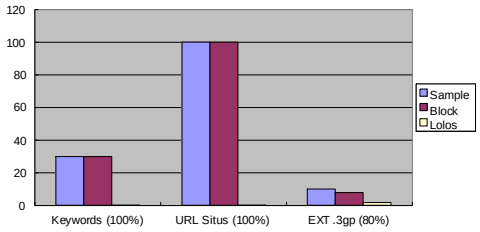
Berdasarkan Gambar 4.14 terlihat *user* dengan nama imas (IP 10.7.248.10).

mengalami *Failed Password* dalam pencobaan pengaksesan *ssh server.*

4.2.4 Performansi *Server*

ISSN: 2302 -2805

Pengujian performansi *server* dilakukan dengan 3 tahapan dan dilakukan dari sisi *client*, yaitu menguji keywords, url situs dan .ext 3gp. Masing –masing tahapan diuji dengan jumlah sample yang berbeda, untuk lebih jelasnya 38bisa di lihat pada Grafik 4.1 berikut :



Gambar 13 Tahapan Uji Sistem

5. Kesimpulan dan Saran

5.1 Kesimpulan

Dari hasil implementasi penelitian ini serta pengambilan data pengukuran dan pengujian mengenai implementasi *network load balancing*, maka dapat diambil kesimpulan sebagai berikut :

1. Fungsi *proxy* sebagai *filtering keywords* dan situs berjalan dengan baik, dengan melihat table 4.1 pengujian performansi dari beberapa *sample* memiliki tingkat keberhasilan masing – masing 100%. Dari 30 kali pengambilan *sample parameter keywords* tidak ada satu pun *keywords* yang lolos, sama halnya dengan *parameter* url situs dari 100 pengambilan *sample* tidak ada situs yang berhasil lolos dari *filtering.*
2. Fungsi *proxy server* dalam membatasi *download* pada ext .3gp berjalan cukup baik, berdasarkan table 4.1 dari 5 kali pengambilan sample hanya 1 kali situs yang lolos, dengan melihat prosentase keberhasilan yang mencapai 80% menunjukan *sever* berhasil meminimalisir akses untuk melakukan *download* ext .3gp.
3. Sistem *security* yang di bangun dengan menggunakan 2 metode berhasil meminimalisir upaya dari *hacker* untuk melakukan serangan ke *sever.*
4. *Monitoring* yang dilakukan *server* dengan menggunakan IPtraf dan authentikasi *login* berfungsi dengan baik sehingga berhasil melacak alamat IP *hacker* .
5. Berdasarkan grafik 4.1 dengan melihat prosentase keberhasilan dari masing-masing *parameter* yaitu *keywords* 100%, url situs 100% dan ext .3gp 80% hal ini menunjukan *server* memiliki performansi yang handal sehingga akses ke informasi yang bersifat porno bisa di minimalisir.

5.2 Saran

Beberapa saran yang dapat diberikan guna pengembangan lebih lanjut antara lain:

1. *Keywords* dan url situs porno yang tersimpan di *squid* harus di *update* secara berkala dikarenakan belum semua situs porno *keyword* nya tersimpan di *squid.*
2. Metode dalam sistem *security* lebih diperbanyak karena cara *hacker* menyerang pun semakin banyak di dunia internet.

Daftar Pustaka

[1] Asdani Kindarto. (2010). 123 Tip Trik Jitu Mengoptimalisasi Linux Ubuntu.Yogyakarta: Andi.
[2] CNC Lab IT Telkom.(2012). Parade Pelatihan CNC 2012 Materi Server. Bandung : CNC Lab IT Telkom
[3] Wahana Komputer. (2011). Administrasi Jaringan dengan Linux Ubuntu 11. Yogyakarta: Andi
[4] Firewall. Retrieved Mei 4,2012 from https://help.ubuntu.com/10.04/serverguide/firewall.html
[5] UFW. Retrieved Mei 4,2012 from https://help.ubuntu.com/community/UFW

Biodata Penulis

Tengku A Riza, memperoleh gelar Sarjana Teknik (S.T), Program Studi Teknik Elektro USU, lulus tahun 2002. Tahun 2008 memperoleh gelar Magister Teknik Elektro (M.T) dari Program Teknik Elektro Telekomunikasi IT Telkom. Saat ini sebagai Staf Pengajar program D3 Teknik Telekomunikasi (D3-TT) IT Telkom, Bandung.