BAB V

HASIL DAN PEMBAHASAN

5.1 Hasil

5.1.1 Diagnosis

5.1.1.1 Analisis Kebutuhan

PT Matra Agung Persada yang merupakan perusahaan distributor yang memiliki banyak komputer dimana setiap komputer saling terhubung ke jaringan internet. Salah satu sarana penunjang kegiatan pekerjaan pada PT Matra Agung Persada adanya jaringan komputer yang menghubungkan dari komputer yang satu ke komputer lainya.

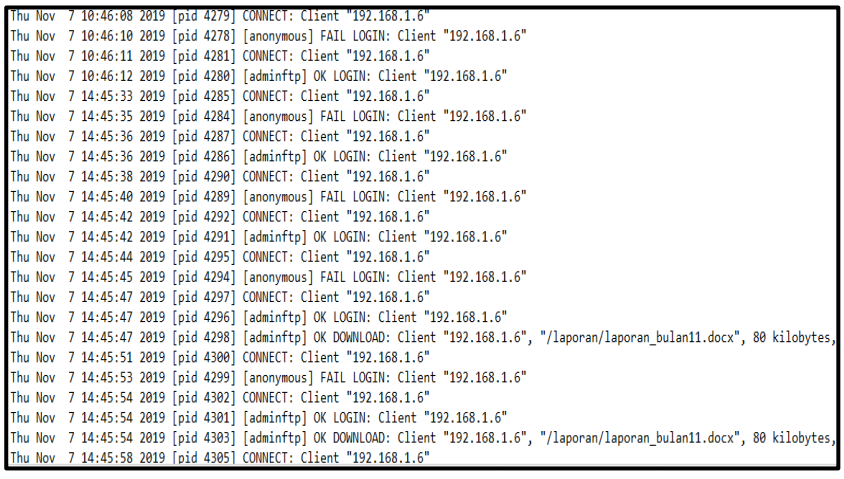
Pada setiap komputer pada PT Matra Agung Persada menyimpan data-data yang penting dimana data tersebut merupakan data-data untuk keperluan pelaksaan kegiatan pekerjaan di PT Matra Agung Persada, data tersebut sering mengalami kerusakan, hilang, bahkan terdapat serangan berupa bruteforce ke server oleh pihak yang tidak bertanggung jawab.

Untuk menyelesaikan permasalahan pada PT Matra Agung Persada maka diperlukan sistem keamanan jaringan *Intrusion Prevention System* (IPS) dan *honeypot* yang dimana berfungsi untuk mendeteksi dan mencegah serangan atau intrusi pada jaringan dan membuat suatu sistem pengalihan serangan

yang sengaja dibuat untuk menjadi umpan atau target para *attacker* dan penyusup yang melakukan aktifitas tidak diinginkan.

5.1.1.2 Analisis Permasalahan

Dari hasil pengamatan dan wawancara secara langsung dengan Ibu Christin selaku *general manager* bahwa belum terdapat sistem keamanan jaringan untuk melindungi jaringan dan data-data yang penting pada PT Matra Agung Persada yang menyebabkan data-data sering mengalami kerusakan, hilang dan bisa ditembus oleh penyusup atau *attacker*, jenis serangan yang diketahui dari hasil *log* yang tersimpan yaitu serangan *bruteforce* yang mencoba semua kemungkinan *user* dan *password* agar dapat *login* ke FTP *server* dan mengambil data penting pada PT Matra Agung Persada identitas yang tersimpan di *log* dengan IP a*ddress user* 192.168.1.6 seperti pada gambar 5.1.



Gambar 5.1 Penyusup Pada PT Matra Agung Persada

maka untuk mengatasi permaslahan tersebut peneliti akan membangun dan menerapkan sistem keamanan jaringan dengan berbasis *Intrusion Prevention Sistem* (IPS) dan *honeypot* agar masalah sebelumnya yang ada pada PT Matra Agung Persada dapat terselesaikan dan tidak terulang kembali.

Tabel 5.1 *Log* FTP

Date Time	User	Status	Client IP	Keterangan
Thus (Kamis) 7 November 10:46:10 2019	Anonymous	*Fail Login*	192.168.1.6	*Login*
Thus (Kamis) 7 November 10:46:12 2019	Admin ftp	Ok *Login*	192.168.1.6	*Login*
Thus (Kamis) 7 November 14:45:47 2019	Admin ftp	Ok Download	192.168.1.6	Mendownload *file* laporan_bulan11.docx

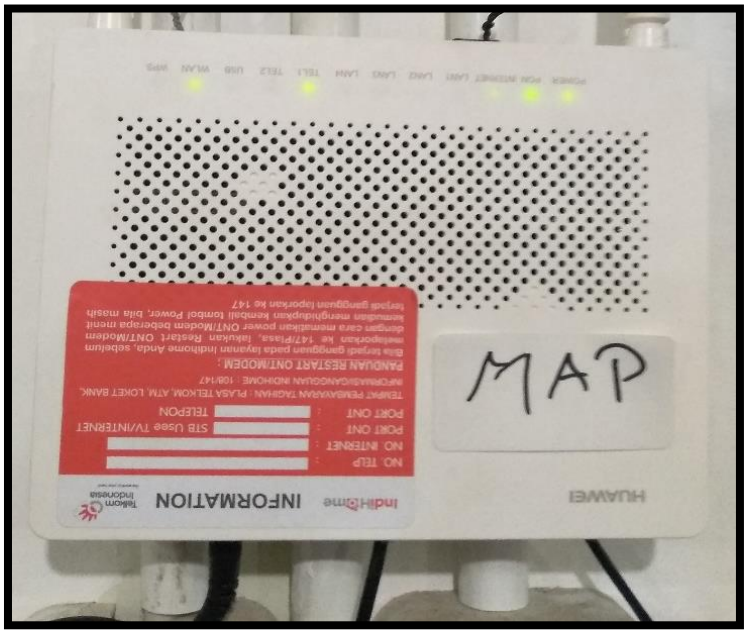
Pada tabel di atas terdapat penyusupan masuk dengan melakukan *login* pada FTP PT Matra Agung dengan alamat IP 192.168.1.16 tujuan ke *server* FTP beralamat 201.99.97.103 pada tabel terdapat gagal *login* pada jam 10:46 pagi dengan nomor proses 4278 selanjutya jam 14:45 pada hari itu juga mereka berhasil masuk dengan nomor proses 4296 dan mereka melakukan *download file* melalui FTP dengan nomor proses 4298 dengan status *ok download* melalui alamat yang sama dengan IP *address* 192.168.1.6 penyusup melakukan *download*

file pada *folder* /laporan dan pada *file* laporan_bulan11.docx disini kami mengambil kesimpulan terdapat serangan dan pencurian data melalui FTP *server*.

5.1.1.3 Teknologi Jaringan

a. Modem

Modem huawei seri HG8245A dengan spesifikasi 1 *port usb*, 2 *port* Telepon, 4 p*ort ethernet* / LAN, 1 *port* konektor SC , 1 WAN , *support* IPV4 dan IPV6 , 300 Mbps WLAN serta dikonfigurasi dengan OLT atau *web* atau NMS Dengan Perangkat Modem ini berfungsi untuk mengubah komunikasi 2 (dua) arah, dengan mengubah sinyal *digital* menjadi sinyal *analog*. Sebagai sumber internet yang akan terhubung pada *client* melalui jaringan LAN pada modem ini terhubung ke *mikrotik* yang akan mengatur jaringan yang berada di Ruang *General* atau Ruang *Server* yang mendistribusikan internet ke Ruangan Administrasi, Ruang *Finance* dan Ruang Direktur.



Gambar 5.2 Modem Pada PT Matra Agung Persada

b. *Router*

merupakan sebuah perangkat keras pada jaringan komputer yang berfungsi untuk menghubungkan beberapa jaringan, baik jaringan sama ataupun berbeda. Pada PT. Matra Agung Persada memakai Mikrotik RB750Gr3 (Hex) dengan spesifikasi RAM 256MB, *slot* USB, *slot* MicroSD serta kecepatan *gigabit ethernet*, berperan sebagai pengendali atau pengatur lalu lintas antar jaringan *router* ini menjadi media penghubung dua buah jaringan atau lebih yang berguna dalam meneruskan data dari satu jaringan ke jaringan lainnya. termasuk memisahkan *traffic* internasional dan *local* untuk kebutuhan pada Ruang *Server router* yang terhubung melalui modem di koneksikan pada *port* 1. 2 dan 3 terhubung ke

switch yang akan mendistribusikan ke setiap perangkat komputer. *port* 4 dan 5 menghubungkan ke *access point* yang akan menghubungkan *client* untuk jaringan *nirkabel*.

Gambar 5.3 *Router* Pada PT Matra Agung Persada

c. *Server*

Server merupakan *server* penyimpanan data-data karyawan pada PT Matra Agung Persada. Menggunakan Thinkserver Seri TS150 dengan spesifikasi *memory* DDR4 2133MH hingga 64GB, penyimpanan HDD hingga 24TB , 4TB penyimpanan SSD RAID dan *Network Interface Controller* (NIC) *gigabitethernet*. mendukung untuk sistem operasi *client* dan *server* sebagai pusat data sehingga *client / user* dapat mengelolah data sesuai kebutuhan. *Server* ini terhubung melalui *mikrotik* yang difungsikan untuk FTP *server* untuk menyimpan data-data perusahaan.

Gambar 5.4 *Server* Pada PT Matra Agung Persada

d. *Switch*

Merupakan suatu perangkat jaringan yang digunakan untuk menghubungkan beberapa komputer dalam sebuah jaringan, berbeda dengan *hub*, *switch* lebih terarah. Perangkat yang digunakan *Switch* D-LINK Seri DES-1061D memiliki spesifikasi jumlah 16 *port* dengan jenis utama 100BASE-TX *fast ethernet* dengan kecepatan *switching* 3,2 Gbps, Qos Per *port* 4. *Switch* ini terhubung melalui *Router* RB750gr3 pada Ruang *General* akan menghubungkan ke Ruangan Administrasi dan Ruangan Direktur yang akan menghubungkan ke Ruangan F*inance*

Gambar 5.5 *Switch* Pada PT Matra Agung Persada

e. *Acces Point*

Perangkat jaringan yang digunakan untuk menghubungkan perangkat *nirkabel* untuk terhubung ke dalam jaringan dengan menggunakan Wi-Fi yang dapat meneruskan data antar perangkat *nirkabel* dengan jaringan berkabel pada satu jaringan. *Access point* yang digunakan pada PT Matra Agung Persada, *Access point* UBIQUITI UNiFi *Access point* UBNT UAP-AC-Lite yang memiliki spesifikasi data *rate* Up to 867Mbps dengan 1x 10/100/1000 *ethernet port* dengan *frequency* 2.4 GHz dan 5GHz memiliki radio *Signal Dual Band* (SDB) yang memungkinkan perangkat yang *support* 5GHz dapat terhubung dengan internet yang stabil hingga 867Mbps sedangkan 2.4GHz mendapatkan 300Mbps, Perangkat ini terhubung pada sisi Ruangan General dan Ruang Direktur.

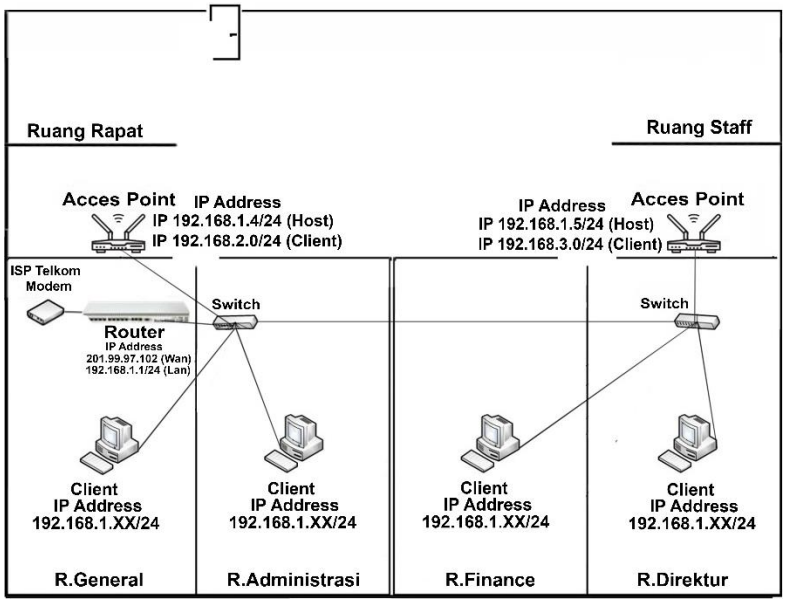
Gambar 5.6 *Acces Point* Pada PT Matra Agung Persada

5.1.1.4 Analisis Topologi Yang Sedang Berjalan

a. IP *addres* yang digunakan pada jaringan *Local Area Network* (LAN) di PT Matra Agung Persada yaitu menggunakan IP *address* kelas C dengan menggunakan *network* 192.168.1.0/24 dengan memakai *default subnet mask* 255.255.255.0 yang digunakan untuk menghubungkan dari jaringan *Local Area Network* (LAN) ke modem. *Internet Service Provider* (ISP) yang digunakan di PT Matra Agung Persada menggunakan Telkom *Speedy Indihome*.

b. Pada PT Matra Agung Persada terdapat beberapa perangkat jaringan sepeti 2 *switch hub* 24 *port*, 2 *acces point*, 1 *device mikrotik* dan menghubungkan 20 *unit* komputer dan 30 *unit* laptop di jaringan dan memanfaatkan jaringan *Internet*

Service Provider (ISP) yang digunakan di PT Matra Agung Persada yaitu Telkom *Speedy Indihome*, topologi dapat dilihat pada gambar 5.7.



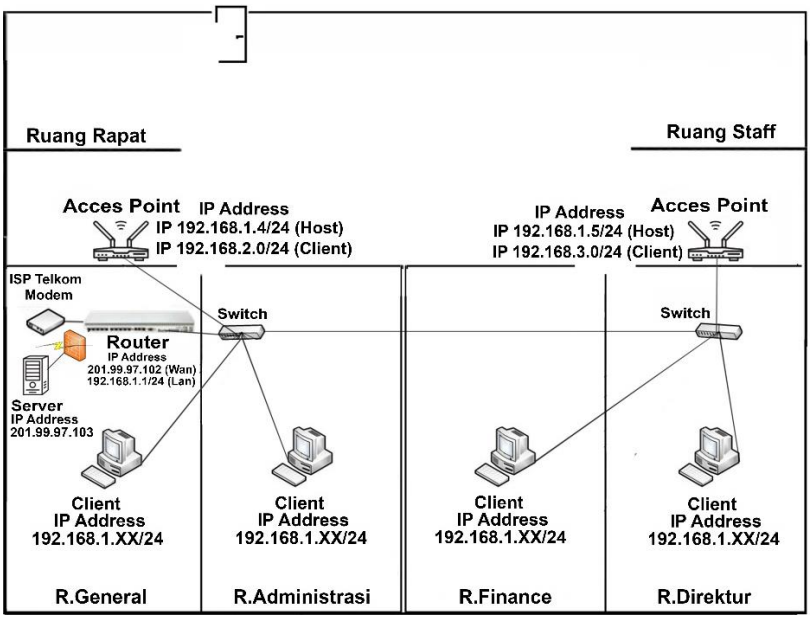
Gambar 5.7 Topologi PT Matra Agung Persada

5.2 *Action Planning*

Pada tahapan ini peneliti memahami masalah dan akan melakukan penerapan konfigurasi *Intrusion Prevention System* (IPS) dengan *snort* dan honeypot menggunakan sistem operasi *linux ubuntu* 18.04 untuk menyelesaikan masalah keamanan jaringan pada PT Matra Agung Persada Pada tahap ini penulis melakukan persiapan kebutuhan perangkat keras (*hardware*) dan perangkat lunak (*software*) yang diperlukan dan akan melakukan uji coba serangan dengan sistem yang telah diimplementasikan pada PT Matra Agung Persada..

5.2.1 Topologi Yang Diusulkan

Dapat dilihat pada gambar dibawah merupakan topologi yang diusulkan, dimana nantinya *Intrusion Prevention System* IPS dan *honeypot* ini akan di letakan pada *server* di PT Matra Agung Persada untuk membuat sistem keamanan jaringan. Pada gambar topologi terdapat *server* yang memiliki IP *addres public* 201.99.97.103 pada *server* tersebut akan melakukan penerapan *snort* dan *honeypot* sebagai proteksi pada *server*, pada Ruang *General* terdapat *router* RB750gr3(hEX) sebagai manajemen jaringan LAN dengan alamat IP *address* 192.168.1.1/24 yang akan distribusikan jaringan menggunakan *switch* ke beberapa Ruangan dan memiliki 2 *access point* UAP-AC-Lite sebagai koneksi *non nirkabel* / Wi-Fi yang terpasang pada Ruangan *General* AP 1 memiliki IP *addres* LAN 192.168.1.4 sedangkan pada *client* memiliki IP *addres client* 192.168.2.1/24 dan AP 2 memiliki IP *addres* LAN 192.168.1.5 dan untuk *client* memiliki IP *addres* 192.168.3.0/24. Pada topologi terdapat *Intrusion Prevention System* (IPS) yang akan menjaga *server* agar tetap aman dari serangan.



Gambar 5.8 Topologi Yang Diusulkan

5.2.2 Implementasi *Server*

Pada penelitian ini Penulis akan melakkukan instalasi *server* menggunakan *Ubuntu Server* 18.04 Distributor *Linux* (*distro*) turunan dari *debian,* Setelah proses instalasi Pada *server Ubuntu*, maka selanjutnya akan melakukan instalasi *snort* konfigurasi *interface ens33* pada jaringan *ubuntu*, *snort* akan melakukan *capture* paket pada jaringan jika ada paket yang di anggap tidak wajar maka paket tersebut akan di *drop* koneksinya pada jaringan yang akan melakukan penyerangan IP *server* 201.99.97.103 dan pada *honeypot* akan berjalan pada *service* FTP melalui *port* 21 sedangkan *port* FTP aslinya pada service *port* 4402.

5.2.3 *Honeypot*

Sistem untuk mencegah terjadinya pencurian data dan informasi, dalam hal memasulkan *service* pada *server* atau mencegah usaha-usaha

pengguna yang tidak mempunyai hak akses ke dalam sebuah sistem informasi. *Honeypot* ini menjadi pengalih dari perhatian penyerang untuk membuatnya seolah-olah berhasil masuk maupun mengambil informasi dari sebuah jaringan padahal data itu tidak penting dan sudah terisolir dari berbagi layanan seperti FTP, *Web, Server* dan sebagainya.

a. Konfigurasi

Untuk penginstallan *honeypot* akan dilakukan dengan cara mendownload melalui *github,* lalu di *install,* kemudian untuk *service honeypot* ini akan berjalan pada *port* 21 yaitu FTP. *Honeypot* akan mengahlikan perhatian *attacker* yang akan mencoba masuk pada *port* 21 akan berjalan layaknya *port* 21 FTP tetapi bukan FTP aslinya, sedangkan FTP aslinya berjalan pada *service port* 4402.

b. Pegujian

Pengujian *service honeypot* yang telah dijalankan akan dilakukan *port scanning* menggunakan *nmap versi open source backtrack* yang akan diarahkan ke IP *address server* yang telah diimplementasikan *honeypot*.

5.2.4 *Intrusion Prevention System* (IPS)

Snort menggunakan metodologi berurutan untuk menangkap, mengidentifikasi, dan mengelompokkan lalu lintas jaringan sebuah *Local Area Network* (LAN). *Intrusion Prevention System* (IPS) mengkombinasikan teknik *firewall* dan metode *Intrusion Detection*

System (IDS) Sistem kerja *Intrusion Prevention System* (IPS) yaitu pendeteksi berbasis *signature*, pendeteksian berbasis anomali dan *monitoring file* pada sistem *host.*

a. Konfigurasi

Konfigurasi yang akan diterapkan untuk *Intrusion Prevention System* (IPS) pada PT Matra Agung Persada ini dengan penginstallan *snort* dengan beberapa *rule*, yaitu:

1. *Rules Bruteforce* FTP

Untuk pencegahan serangan yang terdeteksi melakukan usaha secara terus menerus ke FTP yaitu 201.99.97.103 melalui *port* 4402.

```
#Drop Bruteforce FTP
    drop tcp any any > 201.99.97.103 4402 ( \
        msg:"Drop Bruteforce FTP";
        flow:established,to_server; \
        content:"FTP"; nocase; offset:0; depth:4; \
        detection_filter:track by_src, count 30, seconds 60; \
        sid:1000001; rev:1;)
```

Gambar 5.9 *Rules Drop Bruteforce* FTP

2. *Rules Packet Sniffing*

Semua sumber yang terkoneksi akan di putuskan secara otomatis jika terdeteksi melakukan *capture packet* dari IP *address server* 201.99.97.103

```
#Drop Sniffing
    drop tcp any any > 201.99.97.103 80 (msg:"Drop Packet Sniffing"; flow:established,to_server; \
        content:"Sniffing"; nocase; offset:0; depth:4; \ detection_filter:track by_src, count 30, seconds 60; \
        sid:1000004; rev:4;)
```

Gambar 5.10 *Rules Drop Sniffing*

3. *Rules Smurf Attack*

Pengiriman *packet* berjumlah banyak secara terus menerus ke IP *address server* 201.99.97.103 akan di *drop* secara otomatis terhadap semua sumber IP *address* yang terdeteksi melakukan pengiriman *packet* ICMP.

```
#Drop Packet Smurf Attack
    drop tcp any any > 201.99.97.103 23 (msg:"Drop Packet Smurf Attack"; flow:established,to_server; \
        content:"Smurf"; nocase; offset:0; depth:4; \ detection_filter:track by_src, count 30, seconds 60; \
        sid:1000003; rev:3;)
```

Gambar 5.11 *Rules Drop Smurf Attack*

4. *Rules Denial Of Service*

Untuk mengatasi serangan DoS, *rules* yang akan digunakan yaitu akan memutuskan koneksi terhadap semua sumber IP *address* yang terdeteksi melakukan pengiriman *packet* yang berjumlah banyak secara terus menerus ke IP *server* yaitu 201.99.97.103.

```
#Drop DOS Attack
    drop tcp any any > 201.99.97.103 25 ( \
        msg:"Drop DOS Attack";
        flow:established,to_server; \
        content:"DOS"; nocase; offset:0; depth:4; \
        detection_filter:track by_src, count 30, seconds 60; \
        sid:1000002; rev:2;)
```

Gambar 5.12 *Rules Drop DoS*

b. Pengujian

Pengujian yang akan dilakukan terhadap *Intrusion Prevention System* (IPS) dengan *snort* dan *rules* yang telah dikonfigurasi ada beberapa pengujian diantaranya:

1. ***Bruteforce***

   Teknik *cracking password* dengan cara menembak kemungkinan *password*, membantu *cracker* membobol sebuah *server* dalam melakukan *bruteforce cracker* akan mencari informasi sekumpulan kata acak dari beberapa informasi spesifik menjadikan *wordlist* data sesuai kondisi dan menggabungkan jumalah kata yang tepat. *Tools* yang digunakan yaitu *medusa* dapat di *install* dengan perintah *wget http://www.foofus.net/jmk/tools/medusa-2.1.1.tar.gz -O - | sudo tar -xvz* setelah melakukan *download* masuk folder *cd medusa* lalu ketik perintah */ configure* dan perintah *sudo make && sudo make install* setelah melakukan instalasi *medusa* untuk menjalankan *medusa* sesuai gambar dibawah, keterangan medusa –h (IP *address*_target) –U (*file_username_list*) –P (*file_Password list*) –m (*Module_Protocol*) biarkan "bekerja" program dan jika semuanya berhasil, *shell* akan menampilkan parameter yang diperlukan untuk akses -> Nama pengguna atau *username* dan kata sandi atau *password*.

2. ***Packet Sniffing***

   Pemantauan lalu lintas jaringan pada setiap paket yang dikirim lalu menangkap semua lalu lintas masuk dan keluar, termasuk *username* dan *password* atau data

sensitive lain nya. Untuk membaca dan menganalisa setiap *protocol* yang melintas pada jaringan di perlukan kemapuan khusus dan bantuan aplikasi seperti *wireshark* teknik ini di sebut *spoofing*, *attacker* akan bertindak sebagai *Man-In-the-Middle* (MITM), *tools* yang digunakan *wireshark tools* untuk memonitoring jaringan dan menangkap lalu lintas dalam sebuah jaringan baik lalu lintas yang masuk maupun lalu lintas keluar, *tools* dapat *download* melalui *wireshark* pada *https://www.wireshark.org/download.html*

3. *Smurf Attack*

Serangan amplifikasi yang meningkatkan potensi kerusakannya dengan mengeksploitasi karakteristik jaringan terhadap *pengirimkan* 'aliran' data ke *Internet Control Message Protocol* (ICMP), dengan *broadcast server* menggunakan IP *address*, sehingga respon dari *broadcast + networknya*. Semakin banyak komputer yang terdapat di dalam jaringan yang sama dengan target, maka semakin banyak pula ICMP *echo reply* yang dikirimkan kepada target, sehingga akan membanjiri pada layanan.

4. *Denial Of Service*

Mengirim banyak *request* (permintaan) data ke *server* dengan layanan jaringan yang disedakan oleh

sebuah *host* sehingga *request* yang datang dari pengguna tidak dapat dilayani oleh *server* mengakibatkan *server down* tidak bisa di akses. *Tools* yang digunakan *hping3* yaitu berbasis *linux* untuk cara instalasi dengan melakukan perintah *apt-get install hping3* setelah melakukan *install hping3* untuk cara menggunakan lihat pada gambar dibawah, keterangan *hyping3* –I (*interval*) u100(total *rquest* dikirim) –S (SYN *flag*) –p (*destport*) 80 201.99.97.103 (IP_*address*)

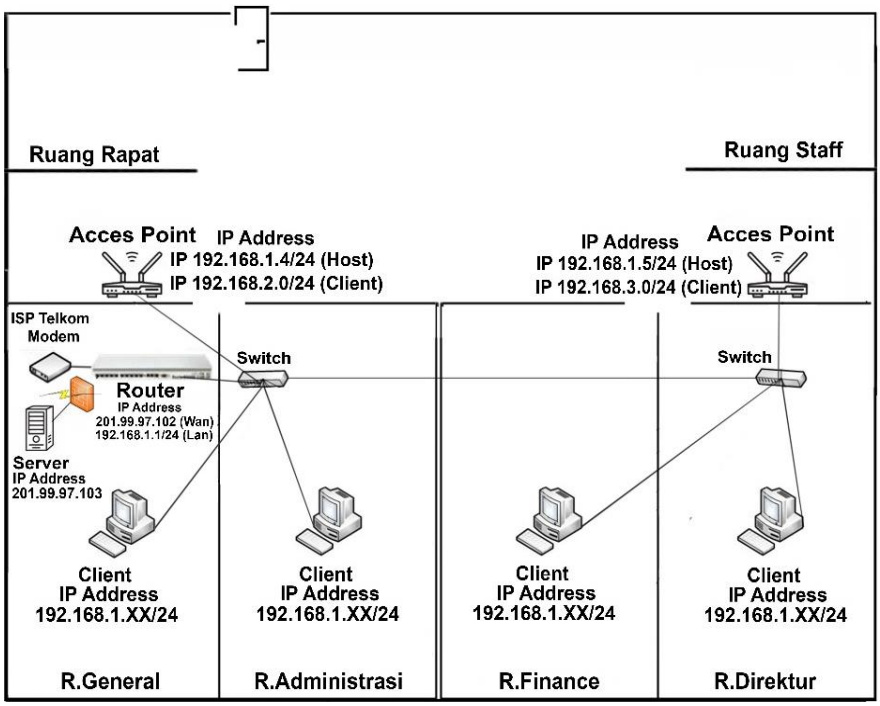
5.3 *Intervention*

Sistem keamanan jaringan yang dirancang pada PT Matra Agung Persada menggunakan keamanan *Intrusion Prevention System* (IPS) dan *honeypot* yang berfungsi untuk mendeteksi dan mencegah serangan atau intrusi pada jaringan serta pengalihan penyerangan sebagai keamanan data dan keamanan jaringan pada PT Matra Agung Persada.

5.3.1 *Implementasi*

1. **Topologi Yang Diterapkan**

   Topologi yang diterapkan dengan tambahan *Intrusion Prevention System* (IPS) dan *honeypot* pada *server* dengan konfigurasi IP *address public* 201.99.97.103, pada Ruangan General



Gambar 5.13 Topologi Yang Diterapkan

2. Implementasi *Server*

Melakukan instalasi sistem operasi *server Ubuntu* 18.04 distributor *linux* (*distro*) turunan dari debian pada perangkat Lenovo Thinkserver. Konfigurasi yang dilakukan dengan IP *Address* 201.99.97.103 sebagai *server* untuk FTP, dengan *Intrusion Prevention System* (IPS) dan *honeypot* akan berjalan pada *service* FTP melalui *port* 21 sedangkan *port* FTP aslinya pada service *port* 4402.

3. *Honeypot*

Penulis melakukan instalasi *honeypot* dengan cara *download* dari *github*. *Honeypot* ini berfungsi untuk mengalihkan perhatian penyerang tentang *server* yang menjadi target.

a. Konfigurasi

honeypot ini akan dijalankan pada *port* 21 sehingga penyerang akan mengira bahwa *server* tersebut adalah FTP padahal sesungguhnya *service* pada *port* 21 telah diganti menjadi *port* 4402 oleh *service honeypot*

Gambar 5.14 Hasil *Port Scanning* Sebelum Ada *Honeypot*

Selanjutnya untuk melakukan konfigurasi dan instalasi *honeypot* penulis *download* dari *github* dengan cara *git clone https://github.com/foospidy/HoneyPy*, setelah itu masuk ke "*directory honeypot*" cd *honeyport* untuk menjalankannya yaitu dengan cara *sudo python setup.py* "*port* yang akan digunakan".

Gambar 5.15 Menjalankan *Honeypot*

Berdasarkan gambar diatas *honeypot* akan berjalan pada *port* 21. Dan untuk membuktikannya akan dilakukan pengujian

melalui *port scanning*. Dengan pengantian *port* FTP asli ke 4402 dan *honeypot* berjalan pada *port* 21 dapat dilihat pada gambar untuk mencegah terjadinya serangan *bruteforce* pada *server* FTP.

```
Nmap scan report for localhost (201.99.97.103)
Host is up (0.0000090s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
3306/tcp open  mysql
4402/tcp open  filtered
```

Gambar 5.16 Pergantian *Port* FTP Ke 4402

b. Pengujian

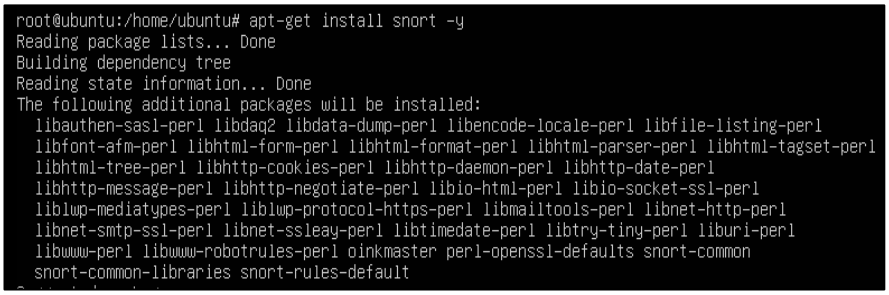
Hoenypot yang telah dikonfigurasi dilakukan pengujian melalui *port scanning*. Dengan pengantian *port* FTP asli ke 4402 dan *honeypot* berjalan pada *port* 21 untuk mencegah terjadinya serangan *bruteforce* pada *server* FTP.

```
Starting Nmap 7.60 ( https://nmap.org )
Nmap scan report for localhost (201.99.97.103)
Host is up (0.0000090s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
3306/tcp open  mysql
4402/tcp open  filtered
```

Gambar 5.17 Pengujian *Port Scanning*

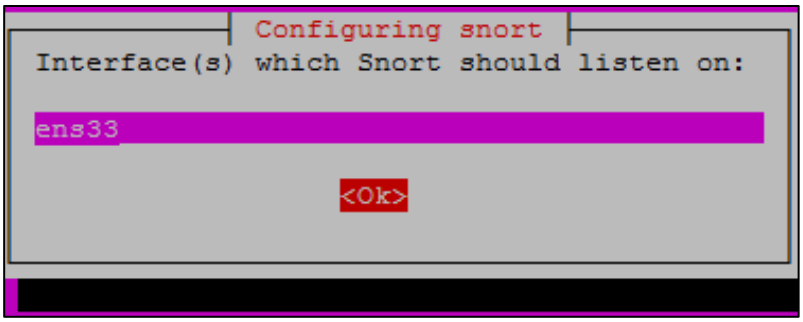
4. Instalasi dan Konfigurasi IPS *Snort*

Untuk instalasi IPS *snort* melalui *repository* dari *ubuntu server* tersebut dengan menggunakan perintah *apt-get install snort*



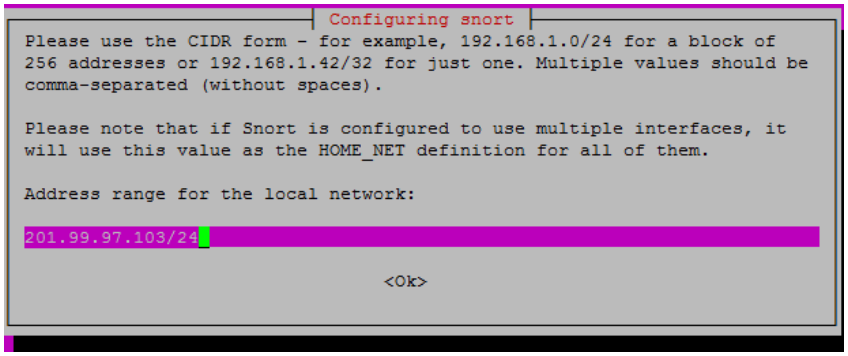
Gambar 5.18 Konfigurasi Dan instalasi *snort*

Pada saat instalasi berjalan maka langsung dilakukan konfigurasi seperti menentukan *interface* dan IP *address* dari *server* yang akan di *install Intrusion Prevention System* IPS.



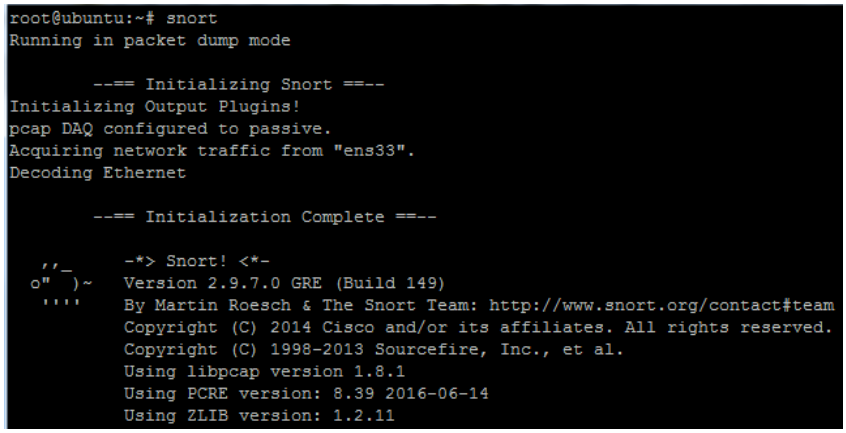
Gambar 5.19 Konfigurasi *Interface Snort*

Selanjutnya akan diarahkan ke *form* untuk konfigurasi IP *address* yang akan di *install* IPS, pada *form* ini diisikan IP *address server* yaitu 201.99.97.103/24.



Gambar 5.20 Konfigurasi IP pada *Snort*

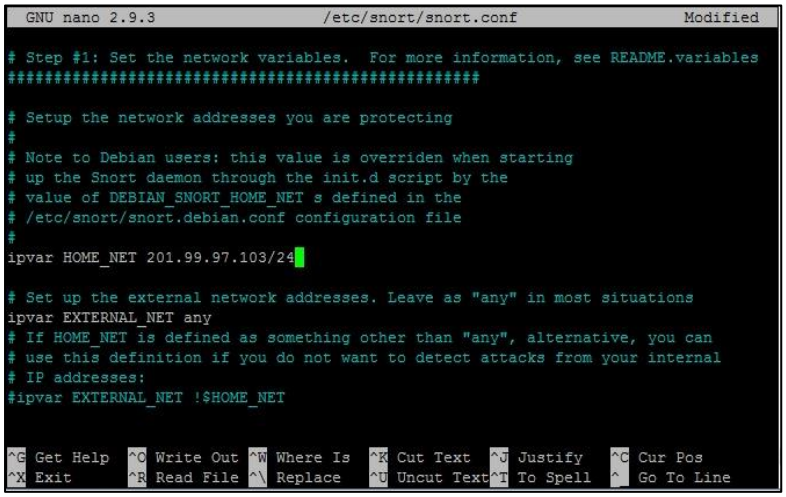
Setelah proses *install* selesai, untuk mencoba apakah *snort* sudah terpasang atau belum dapat menjalankan perintah "*snort*".



Gambar 5.21 Menjalankan *Snort*

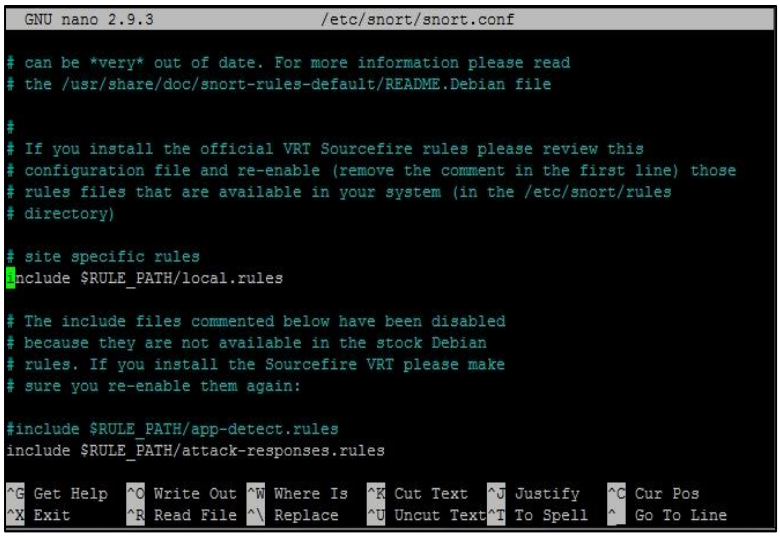
a. Konfigurasi

Selanjutnya yaitu akan dilakukan konfigurasi pada *file /etc/snort/snort.conf*, pada *file* ini akan ditambahkan IP *Address* dan lokasi *file rules local*.



Gambar 5.22 Konfigurasi IP Pada *File Snort.Conf*

Selanjutnya menambahkan lokasi *file local rules*, *file* ini berfungsi untuk menambahkan *rules* yang kita buat sendiri.



Gambar 5.23 Konfigurasi *Local Rules Snort.Conf*

Selanjutnya *edit local.rules* dan masukkan *rules* IPS untuk mencegah serangan *Bruteforce* FTP, *Sniffing*, *Smurf attack* dan *Denial of Service*.

1. ***Rules Drop Bruteforce FTP***

Rules yang digunakan untuk mengatasi *bruteforce* FTP yaitu akan memutuskan koneksi kepada IP *address* yang terdeteksi mengakses *server* FTP secara terus menerus melalui *port* 4402.

```
#Drop Bruteforce FTP
    drop tcp any any > 201.99.97.103 4402 ( \
        msg:"Drop Bruteforce FTP";
        flow:established,to_server; \
        content:"FTP"; nocase; offset:0; depth:4; \
        detection_filter:track by_src, count 30, seconds 60; \
        sid:1000001; rev:1;)
```

Gambar 5.24 *Rules Drop Bruteforce* FTP

Dari gambar diatas rules *drop brutefoce* FTP adalah *action* yang akan dilakukan pada FTP, TCP adalah *protocol* yang digunakan, *any any* adalah IP *address* dan *port* ftp - > akan mendirect ke IP 201.99.97.103 dan *port* tujuan, *msg* adalah notifikasi yang akan muncul identitas dari *rules,* dan *rev* adalah nomor *revisi* dari *log rules.*

2. ***Rule Drop Packet Sniffing***

Rules yang digunakan untuk mengatasi *packet sniffing* yaitu akan memutuskan koneksi IP *address* yang terdeksi melakukan *packet sniffing* dari IP *address server.*

```
#Drop Sniffing
    drop tcp any any > 201.99.97.103 80 (msg:"Drop Packet Sniffing"; flow:established,to_server; \
        content:"Sniffing"; nocase; offset:0; depth:4; \ detection_filter:track by_src, count 30, seconds 60; \
        sid:1000004; rev:4;)
```

Gambar 5.25 *Rules Drop Sniffing*

Dari gambar diatas rules *drop packet sniffing* adalah *action* yang akan dilakukan, TCP adalah *protocol* yang digunakan, *any any* adalah IP *address* dan *port 80*, - > akan mendirect ke IP *address* 201.99.97.103 dan *port* tujuan, *msg* adalah notifikasi yang akan muncul, *sid* adalah identitas dari *rules*, dan *rev* adalah nomor *revisi* dari *log rules*.

3. *Rules Drop Smurf Attack*

Rules yang digunakan untuk mengatasi serangan *smurf attack* yaitu akan memutuskan koneksi IP *address* yang terdeksi melakukan pengiriman *packet* ICMP yang berjumlah banyak secara terus menerus ke *server*.

```
#Drop Packet Smurf Attack
    drop tcp any any > 201.99.97.103 23 (msg:"Drop Packet Smurf Attack"; flow:established,to_server; \
        content:"Smurf"; nocase; offset:0; depth:4; \ detection_filter:track by_src, count 30, seconds 60; \
        sid:1000003; rev:3;)
```

Gambar 5.26 *Rules Smurf Attack*

Dari gambar diatas *rules drop Smurf Attack* adalah *action* yang akan dilakukan, TCP adalah *protocol* yang digunakan, *any any* adalah IP *address* dan *port*, > akan mendirect ke IP 201.99.97.103 dan *port* tujuan, *msg* adalah notifikasi yang akan muncul, *sid* adalah identitas dari *rules*, dan *rev* adalah nomor *revisi* dari *log rules*.

4. *Rules Drop Denial OF Service (DoS)*

Rules yang digunakan untuk mengatasi serangan *Denial of Service* (DoS) yaitu akan memutuskan koneksi ip yang terdeteksi melakukan pengiriman *packet* yang berjumlah banyak secara terus menerus ke *server*.

```
#Drop DOS Attack
    drop tcp any any > 201.99.97.103 25 ( \
        msg:"Drop DOS Attack";
        flow:established,to_server; \
        content:"DOS"; nocase; offset:0; depth:4; \
        detection_filter:track by_src, count 30, seconds 60; \
        sid:1000002; rev:2;)
```

Gambar 5.27 *Rules Drop DoS*

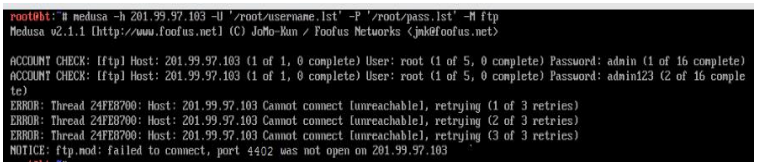
Dari gambar diatas rules *drop brutefoce Dos* adalah *action* yang akan dilakukan, TCP adalah *protocol* yang digunakan, *any any* adalah IP *address* dan *port 3306*, - > akan mendirect ke IP 201.99.97.103 dan *port* tujuan, *msg* adalah notifikasi yang akan muncul, *sid* adalah identitas dari *rules*, dan *rev* adalah nomor *revisi* dari *log rules*.

b. Pengujian

1. Uji Coba *Rules Drop Bruteforce FTP*

Untuk pengujian *bruteforce* FTP akan dilakukan untuk menemukan *user* dan *password* agar dapat *login* di FTP *server* tersebut. Untuk serangan ini akan digunakan *medusa versi* open *source backtrack* yang akan diarahkan ke IP *address server* dengan menggunakan *wordlist* untuk

username yaitu "*username.lst*" dan untuk *password* yaitu "*pass.lst*".



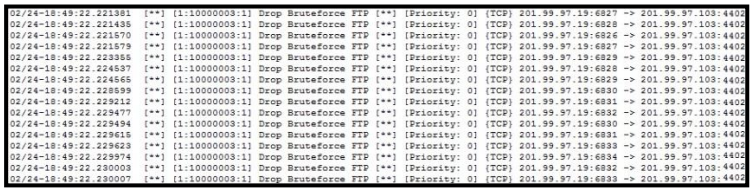
Gambar 5.28 *Bruteforce* FTP Pada *Server*

Dapat di lihat dari gambar diatas *bruteforce* FTP tidak dapat berjalan karena koneksi ke *server* terputus secara otomatis yang terdeteksi oleh *Intrusion Prevention System* (IPS) bahwa ada yang melakukan *login* terus menerus ke *port* FTP yaitu 4402 dengan rentan waktu yang berdekatan. Sehingga *Intrusion Prevention System* (IPS) mendeteksi serangan tersebut adalah *bruteforce login* FTP.

Tabel 5.2 Hasil pengujian *Bruteforce* FTP

IP Attacker	Tindakan	Hasil
201.99.97.19	*Drop Bruteforce* FTP	*Rules* Berhasil Menghentikan B*ruteforce*

Untuk pengujian telah dilakukan uji coba serangan *bruteforce* ke FTP *server* yang telah diterapkan IPS dengan rules untuk mengatasi serangan *bruteforce* FTP. Dan setetelah diterapkan *rules* tersebut berhasil mengatasi serangan *bruteforce* FTP pada *server*. Dan hasil serangan tertangkap pada *log snort* yang dijalankan.

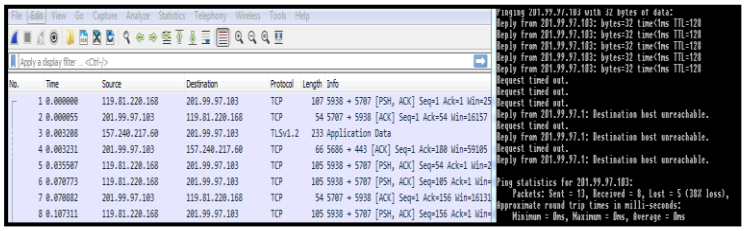


Gambar 5.29 *Log Pengujian Bruteforce* FTP

Log snort terdapat *log* serangan FTP yang berhasil di *drop* dengan keterangan tanggal dan waktu penyerangan {**}{nomor poses id identitas dan terdapat *rules* FTP pada nomor rev 1:sid100000003:rev} dengan keterangan *drop bruteforce* FTP dengan *priority* 0 , penyerang dengan IP *address* 201.99.97.19:37148 -> ke alamat IP *address server*:*port* yang di serang 201.99.97.103:4402

2. Uji Coba *Rules Packet sniffing*

Untuk pengujian *packet sniffing* akan dilakukan untuk menemukan *user* dan *password* agar dapat *login* pada *server*. Untuk serangan ini akan digunakan aplikasi *wireshark* yang akan merekam lalu lintas atau *traffic* pada *server*.



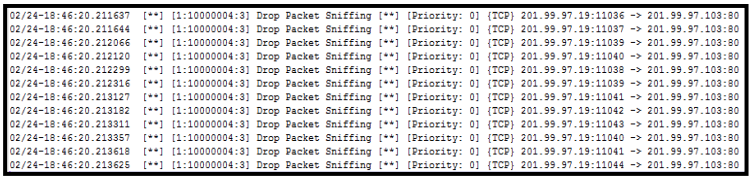
Gambar 5.30 *Packet Sniffing*

Berdasarkan pada gambar diatas setelah konfigurasi dan *sniffing* dijalankan beberapa saat kemudian koneksi ke *server* terputus dikarekan terdeteksi oleh *snort Intrusion Prevention System* (IPS) bahwa ada akses penangkapan paket dari penyerang yang melakukan *sniffing* terhadap arus paket menuju dan keluar dari *server*.

Tabel 5.3 Hasil Pengujian *Sniffing*

IP Attacker	Tindakan	Hasil
201.99.97.19	*Drop Packet Sniffing*	Rules Berhasil Menghentikan *Sniffing*

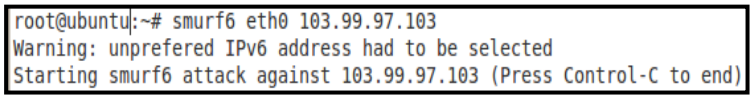
Untuk pengujian telah dilakukan ujicoba serangan *packet sniffing* untuk menangkap informasi pada *server* yang telah diterapkan IPS dengan *rules* untuk mengatasi serangan *sniffing*. Dan setetelah diterapkan *rules* tersebut berhasil memutus koneksi dari serangan *packet sniffing* pada *server*. Dan hasil serangan tertangkap pada *log snort* yang dijalankan.



Gambar 5.31 *Log packet Sniffing*

3. Uji Coba *Rules Packet Smurf Attack*

Selanjutnya untuk uji coba *Smurf Attack* akan dilakukan Untuk serangan ini akan digunakan *smurf6 versi open source backtrack* yang akan diarahkan ke IP *address server*



Gambar 5.32 *Smurf Attack*

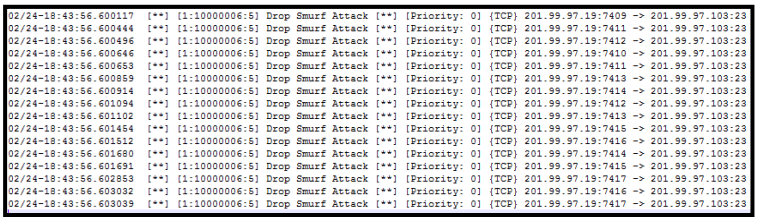
Dapat di lihat dari gambar diatas *Smurf Attack* tidak dapat berjalan karena koneksi ke *server* diputus karena terdeteksi oleh *Intrusion Prevention System* (IPS) ada uji coba pengirim *packet* terus menerus ke *server* dengan satu sumber yang sama dan rentan waktu yang berdekatan. Sehingga IPS mendeteksi serangan tersebut adalah *Smurt Attack.*

Tabel 5.4 Hasil Pengujian *SmurfAttack*

IP Attacker	Tindakan	Hasil
201.99.97.19	*Drop Smurf Attack*	Rules Berhasil Menolak *Packet Smurf Attack*

Untuk pengujian telah dilakukan uji coba serangan *packet sniffing* untuk menangkap informasi pada *server* yang telah diterapkan IPS dengan *rules* untuk mengatasi serangan *Sniffing*. Dan setetelah diterapkan *rules* tersebut berhasil memutus koneksi dari serangan *packet sniffing*

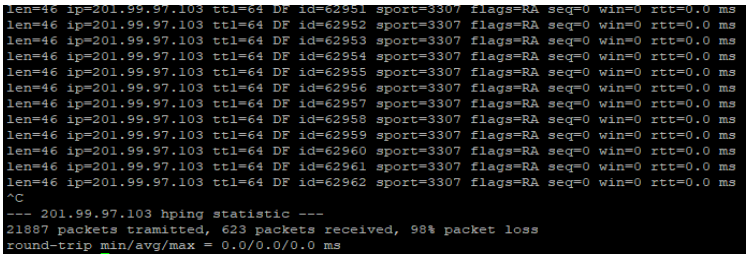
pada *server*. Dan hasil serangan tertangkap pada *log snort* yang dijalankan.



Gambar 5.33 *Log pengujian SmurfAttack*

4. Uji Coba *Denial Of Service* (DOS)

Selanjutnya akan dilakukan pengujian *Intrusion Prevention System* (IPS) menggunakan serangan *Denial of Service* (DoS). Pada penelitian ini akan digunakan *hping3* dari *tools backtrack* yang akan diarahkan ke *port* 3306 yaitu pada IP *address server* yang telah di *setting* IPS.



Gambar 5.34 *Denial Of Service* (DOS) *Pada Server*

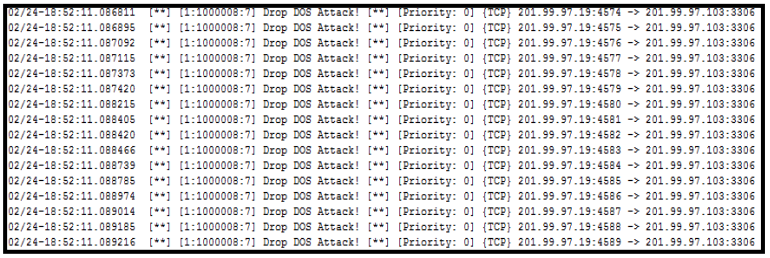
Dapat di lihat dari gambar diatas *DoS* tetap berjalan namun paket ditolak oleh *server*. Hasil serangan *DoS* terdapat *packet* yang dikirim sebanyak 21887 namun *packet* tersebut banyak yang masuk sebagai packet *loss*, yaitu sebanyak 98%. Hal tersebut terjadi karena IPS *mendeteksi* terdapat *packet* yang sama masuk secara

bersamaan sehingga IPS memblock *packet* tersebut untuk tidak diteruskan.

Tabel 5.5 Hasil Ppengujian *Denial Of Service* (DoS)

IP Attacker	Tindakan	Hasil
201.99.97.19	*Drop Packet* DoS	Rules Berhasil Menolak *Packet* Dari DoS

Untuk pengujian telah dilakukan ujicoba serangan *Denial of Service* dengan cara membanjiri *server* dengan *packet* yang sangat banyak terhadap *server* yang telah diterapkan IPS dengan *rules* untuk mengatasi serangan DoS. Dan setetelah diterapkan rules tersebut berhasil memutus koneksi dari serangan DoS pada *server*. Dan hasil serangan tertangkap pada *log snort* yang dijalankan.

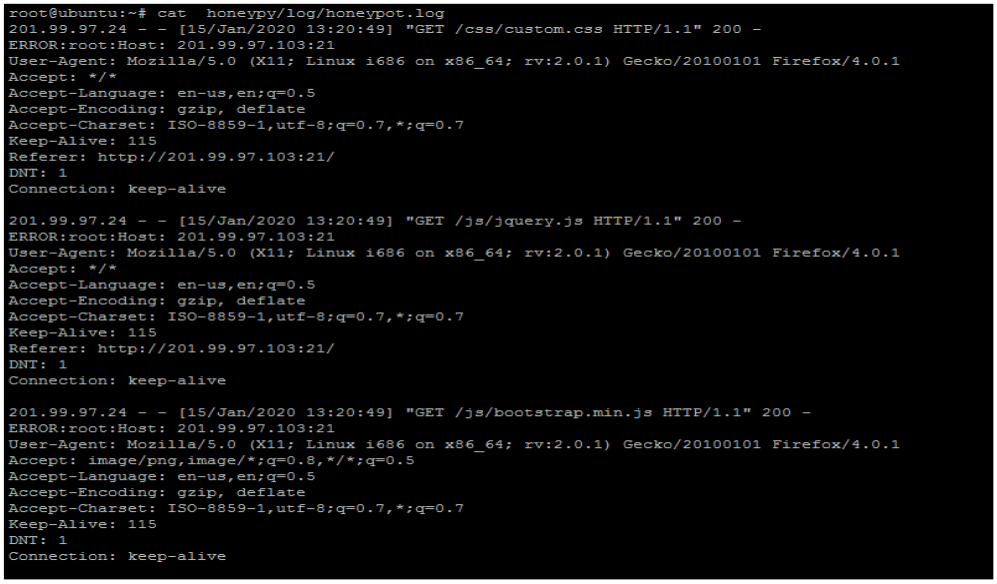


Gambar 5.35 *Log* Pengujian *Denial Of Service* (DOS)

5.4 *Reflection*

Ditahap terkhir *Action Research ini*, akan dilakukan *review* dan evaluasi terhadap implementasi *honeypot* dan IPS yang dilakukan terhadap *server*, *honeypot* akan menambah layanan melalui *port*, sehingga *port* yang terbuka bertambah, namun *port* tersebut hanya untuk *honeypot* itu sendiri

bukan untuk *server*. Untuk *port* yang ditambah oleh *honeypot* ini adalah *port* 21, dimana *port* terebut adalah *port* untuk yang digunakan untuk layanan FTP palsu dari *honeypot* itu sendiri, sehingga setelah *honeypot* ini diimplementasikan pada *server*, *hacker* akan mengira *server* tersebet digunakan untuk FTP sampai *hacker* tersebut mencoba menyerang melalu *port* 21 tersebut menggunakan *port scanning* hasilnya akan tampil *port* 21 adalah *service* FTP, namun sebenarnya *service* pada *port* 21 telah diganti dengan *service honeypot*.



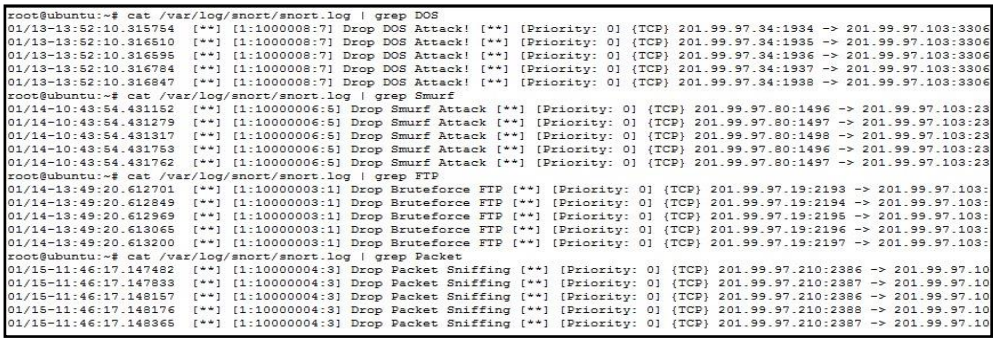
Gambar 5.36 *Log Honeypot*

Tabel 5.6 Hasil Implementasi Honeypot

No	Nama Serangan	Jumlah Seranagan	Sumber Serangan	Target Serangan	Hasil Serangan
1	*Port Scanning*	5	201.99.97.24	201.99.97.103	*Connection Refused*

Pada *table* 5.2 dijelaskan pada *log honeypot* mencatat terdapat jumlah serangan sebanyak 5 kali serangan *port scanning* yang berasal dari alamat 201.99.97.24 dengan hasil serangan *connection refused* dikarenakan koneksi di tolak oleh *server*.

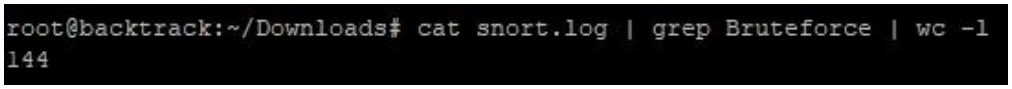
Untuk *Intrusion Prevention System* (IPS) nya sendiri, setelah di implementasikan dan ditambah *local rules* untuk mencegah serangan *Bruteforce* FTP, *Sniffing, Smurf attack* dan *Denial of Service* (DoS). Setelah dilakukan pengujian serangan pada *rules* yang telah di implementasikan IPS berhasil mendeteksi dan melakukan tindakan dengan memutus koneksi dan melakukan *drop packet* dari penyerang melalui IPS dengan *block packet* yang mencurigakan pada setiap *packet* yang masuk ke *server* berdasarkan catatan *log snort* terdapat serangan sebanyak 144 kali percobaan *bruteforce* FTP, 45 *Sniffing*, 1734 *packet* serangan *Smurf Attack* dan serangan *Denial of Service* (DoS) berupa *packet* serangan yaitu sebanyak 10240 untuk melumpuhkan *server* tetapi berhasil di *drop* oleh IPS sehingga dicatat sebagai *packet loss* sebanyak 98% dari jumlah *packet* serangan *Denial of Service* (DoS) yang dijalankan.



Gambar 5.37 *Log* IPS

Tabel 5.7 Hasil Implementasi *Intrusion Prevention System* (IPS)

No	Nama Serangan	Jumlah Serangan	Sumber Serangan	Target Serangan	Hasil Serangan
2	*Bruteforce* FTP	144	192.168.1.6	201.99.97.103	*Drop Bruteforce* FTP
3	*Packet Sniffing*	45	201.99.97.210	201.99.97.103	*Drop Packet Sniffing*
4	*Smurf Attack*	1734	201.99.97.80	201.99.97.103	*Drop Smurf Attack*
5	*Denial of Service*	10240	201.99.97.34	201.99.97.103	*Drop Packet DoS*



Gambar 5.38 *Log* Jumlah Serangan *Bruteforce*

Dapat dilihat hasil implementasi IPS serangan *bruteforce FTP* dengan jumlah serangan 144 dan IP *address* 201.99.97.19 berhasil di *drop* dengan keterangan *drop bruteforce* FTP pada hasil serangan, serangan *Sniffing* beralamat 201.99.97.201 jumlah serangan 45 *packet* berhasil di *drop* hasil keterangan *drop packet Sniffing* pada hasil serangan, *Smurf Attack* dengan jumlah 1734 *packet* berhasil di *drop* pada tabel dengan IP *address* 201.99.97.80 dan pada serangan *DoS* memiliki 10240 total *packet* serangan dari alamat 201.99.97.34 berhasil di *drop* oleh IPS dengan keterangan *drop packet* DoS.