

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Arsitektur Jaringan

Sebelum dilakukan pengujian, dirancang terlebih dahulu arsitektur jaringan sesuai dengan gambaran umum sistem. Implementasi ini menjelaskan pengalamatan serta mengkonfigurasi router agar dapat terkoneksi satu sama lain meskipun berbeda jaringan.

4.1.1 Konfigurasi *IP Address* IDS

Pada IDS terdapat dua buah NIC, yaitu eth0 dan eth1. Pada eth0 menggunakan mode jaringan *Network Address Translation* (NAT) dimana berfungsi agar dapat memiliki akses *Internet*. Kemudian pada eth1 menggunakan mode jaringan host only. Konfigurasi *IP Address* IDS sebagai berikut:

Tabel 4.1 Konfigurasi *IP Address* IDS

Network adapter	eth0	eth1
IP Address	192.168.10.129	192.168.1.128
Network	192.168.10.0	192.168.1.0
Netmask	255.255.255.0	255.255.255.0
Gateway	192.168.10.2	192.168.1.2
Broadcast	192.168.10.255	192.168.1.255

4.1.2 Konfigurasi *IP Address* Server

Komputer *server* hanya memiliki sebuah NIC, yaitu eth0. Pada eth0 menggunakan mode jaringan host only. Konfigurasi *IP Address* pada *server* adalah sebagai berikut:

Tabel 4.2 Konfigurasi *IP Address* Server

Network adapter	eth0
IP Address	192.168.1.130
Network	192.168.1.0
Netmask	255.255.255.0
Gateway	192.168.1.2
Broadcast	192.168.1.255

4.1.3 Konfigurasi *IP Address Attacker*

Komputer *attacker* memiliki sebuah NIC, yaitu *eth0* yang menggunakan mode jaringan *host only*. *IP Address attacker* berbeda jaringan dengan *server* sehingga membutuhkan router untuk dapat mengakses komputer *server*. Konfigurasi *IP Address* pada *attacker* adalah sebagai berikut:

Tabel 4.3 Konfigurasi *IP Address Attacker*

Network adapter	eth0
IP Address	192.168.2.128
Network	192.168.2.0
Netmask	255.255.255.0
Gateway	192.168.2.2
Broadcast	192.168.2.255

4.1.4 Konfigurasi Router

Tujuan menggunakan router yaitu untuk menghubungkan beberapa jaringan yang sama maupun berbeda. Implementasi ini akan menghubungkan jaringan 192.168.1.0/24 dengan 192.168.2.0/24 agar dapat berkomunikasi. Konfigurasi router adalah sebagai berikut:

Tabel 4.4 Konfigurasi *IP Address Router*

Network adapter	Ether1	Ether2
IP Address	192.168.1.2	192.168.2.2
Network	192.168.1.0	192.168.2.0
Netmask	255.255.255.0	255.255.255.0

4.2 Implementasi Perangkat Lunak

Sebelum melakukan tahap pengujian terdapat beberapa tahap instalasi dan konfigurasi yang terdiri dari:

4.2.1 Instalasi Aplikasi Pendukung

Tahap ini melakukan instalasi aplikasi yang dibutuhkan untuk mempermudah pengaplikasian sistem.

```
#apt-get install -y phpmyadmin php5-curl wkhtmltopdf xvfb lynx
unzip
```

4.2.2 Instalasi dan Konfigurasi Snort

Tahap ini diterapkan pada komputer yang bertugas sebagai IDS dengan sistem operasi Ubuntu Server 14.04, berikut tahapan yang dilakukan.

- a. *Install* aplikasi pendukung Snort yang terdapat pada repositori Ubuntu

```
#apt-get install -y build-essential libpcap-dev
libpcap3-dev libdumbnet-dev bison flex zlib1g-dev
liblzma-dev openssl libssl-dev
```

- b. *Download* dan *install library* DAQ dan Snort terbaru dari situs Snort

```
#wget https://snort.org/downloads/snort/daq-2.0.6.tar.gz
#tar -xvzf daq-2.0.6.tar.gz
#cd daq-2.0.6
#./configure
#make
#sudo make install
```

```
#wget https://snort.org/downloads/snort/snort-
2.9.8.3.tar.gz
#tar -xvzf snort-2.9.8.3.tar.gz
#cd snort-2.9.8.0
#./configure --enable-sourcefire
#make
#sudo make install
#ldconfig
#ln -s /usr/local/bin/snort /usr/sbin/snort
```

- c. *Konfigurasi* Snort

```
//Membuat user dan group snort
#groupadd snort
#useradd snort

//membuat direktori dan file untuk rule
#mkdir /etc/snort
#mkdir /etc/snort/rules
#touch /etc/snort/rules/nmap.rules
#touch /etc/snort/rules/ftp.rules
#touch /etc/snort/rules/ssh.rules
#touch /etc/snort/rules/ddos.rules
#touch /etc/snort/sid-msg.map
#mkdir /etc/snort/preproc_rules
#mkdir /usr/local/lib/snort_dynamicrules
#mkdir /etc/snort/so_rules
//membuat direktori untuk logging snort
#mkdir /var/log/snort
#mkdir /var/log/snort/archived_logs

//memberikan hak akses
#chmod -R 5775 /etc/snort
```

```
#chmod -R 5775 /var/log/snort
#chmod -R 5775 /var/log/snort/archived_logs

//mengganti kepemilikan folder
#chown -R snort:snort /etc/snort
#chown -R snort:snort /var/log/snort
#chown -R snort:snort /usr/local/lib/snort_dynamicrules

//Salin konfigurasi file
#snort-2.9.8.3/etc/
#sudo cp *.conf* /etc/snort
#sudo cp *.map /etc/snort
#sudo cp *.dtd /etc/snort
```

d. Konfigurasi file /etc/snort/snort.conf

```
ipvar HOME_NET 192.168.1.0/24
ipvar EXTERNAL_NET !$HOME_NET

var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/so_rules
var PREPROC_RULE_PATH /etc/snort/preproc_rules

#tambahkan rule yang dibutuhkan
include $RULE_PATH/nmap.rules
include $RULE_PATH/ftp.rules
include $RULE_PATH/ssh.rules
include $RULE_PATH/ddos.rules
```

e. Uji file konfigurasi snort

```
#snort -T -i eth0 -c /etc/snort/snort.conf
```

4.2.3 Instalasi dan Konfigurasi Barnyard2

Tahap ini melakukan instalasi aplikasi pendukung yang dibutuhkan oleh Snort yaitu Barnyard2

a. *Install* paket pendukung Barnyard2

```
#apt-get install -y mysql-server libmysqlclient-dev mysql-
client autoconf libtool
```

b. Tambahkan *output alert* pada file /etc/snort/snort.conf

```
output unified2: filename snort.u2, limit 128
```

c. *Download* dan *install* barnyard2

```
#wget https://github.com/firnsy/barnyard2/archive/
master.zip
#unzip master.zip
```

```
#cd barnyard2-master
#autoreconf -fvi -I ./m4
//Membuat link ke dumbnet.h
#ln -s /usr/include/dumbnet.h /usr/include/dnet.h
#ldconfig

//konfigurasi MySQL library sesuai arsitektur OS
#./configure --with-mysql --with-mysql
libraries=/usr/lib/i386-linux-gnu

#make
#make install
```

d. Konfigurasi barnyard2

```
//Salin dan buat beberapa file barnyard2 ke folder snort
#cp etc/barnyard2.conf /etc/snort
#mkdir /var/log/barnyard2
#chown snort.snort /var/log/barnyard2
#touch /var/log/snort/barnyard2.waldo
#chown snort.snort /var/log/snort/barnyard2.waldo
```

4.2.4 Implementasi Database

a. Membuat Database snort

Pada *database*, akan dibuat duah buah *database* yaitu snort dan notifsntort. *Database* snort hanya mencatat aktifitas serangan, sedangkan database notifsntort menampilkan informasi dalam bentuk grafik dan Administrator dapat mengatur akun Telegram untuk kebutuhan notifikasi melalui aplikasi *instant messaging* Telegram.

```
#mysql -u root -p
#mysql> create database snort;
#mysql> use snort;
#mysql> source ~/snort_src/barnyard2-master/schemas/create
_mysql
#mysql> CREATE USER 'snort'@'localhost' IDENTIFIED
#BY'snort';
#mysql> grant create, insert, select, delete, update on
#snort.* to'snort'@'localhost';
#mysql> exit
```

b. Download library MySQL UDF agar dapat melakukan *trigger*

```
#wget
https://github.com/mysqludf/lib_mysqludf_sys/archive/master.
zip
#unzip master.zip
#cd lib_mysqludf_sys-master
```

```
//buka file Makefile dan ganti menjadi
LIBDIR=/usr/lib/mysql/plugin

//kemudian install
#./install.sh
```

c. *Download dan install acidbase*

1) Konfigurasi php.ini

```
//ganti error_reporting menjadi
error_reporting = E_ALL & ~E_NOTICE
```

2) *Download* aplikasi pendukung

```
#apt-get install -y php-pear libwww-perl php5-gd

//download packages yang dibutuhkan
#pear config-set preferred_state alpha
#pear channel-update pear.php.net
#pear install --alldeps Image_Color Image_Canvas
Image_Graph

#cd /build/php5-DgntdS/php5-5.9+dfsg/pear-build-download
#ls

//terdapat 6 package yang harus diinstall
#tar zxf Image_Color*.tgz
#cp package.xml ./Image_Color*/
#cd Image_Color*
#pear install package.xml
#cd ..
```

3) *Download* Base

```
#cd /usr/src
# wget
http://sourceforge.net/projects/secureideas/files/
BASE/base-#1.4.5/base-1.4.5.tar.gz
#tar -zxf base-1.4.5.tar.gz
#cp -r base-1.4.5 /var/www/html/base
#chown -R www-data:www-data /var/www/html/base
#service apache2 restart
```

4) Konfigurasi Antarmuka Base

Untuk melakukan konfigurasi Base, maka akses ke <http://localhost/base>

```
//isikan path
path : /usr/share/php/adodb

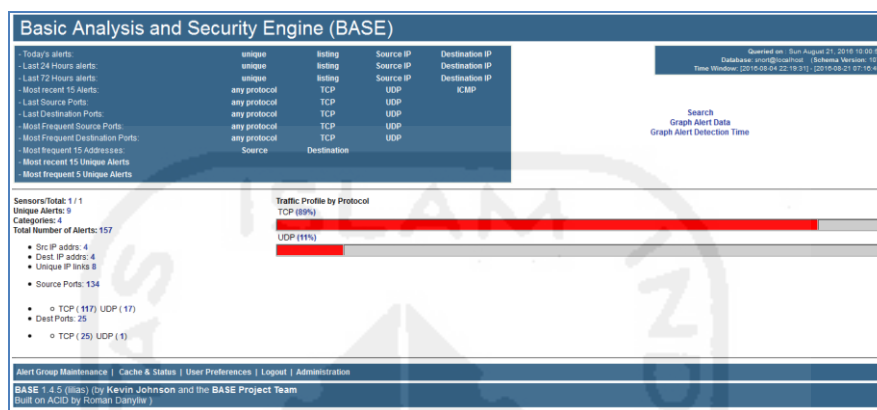
//isikan database
```

```

Database Name : snort
Database Host : localhost
Database User Name: snort
Database Password : snort

//kemudian klik Create BaseAG

```



Gambar 4.1 Implementasi Antarmuka BASE

4.2.5 Implementasi *Rule Snort*

Rule Snort merupakan aturan-aturan yang digunakan untuk mendeteksi adanya intrusi maupun aktifitas-aktifitas yang mencurigakan. Berikut beberapa rule snort dari situs resmi snort yang diimplementasikan pada penelitian ini:

a. *Rule Port Scanning* pada /etc/snort/rules/nmap.rules

```

alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"NMAP scan SYN"; flags:S,12; ack:0; threshold: type both, track by_dst, count 1, seconds 60; reference:arachnids,27; GID:1; sid:10000001; rev:001; classtype: attempted-recon;)
alert tcp $EXTERNAL_NET any -> $HOME_NET [21,22,80] (msg:"NMAP scan FIN"; flow:stateless; flags:F,12; ack:0; threshold: type both, track by_dst, count 3, seconds 10; reference:arachnids,27; GID:1; sid:10000002; rev:001; classtype:attempted-recon;)
alert tcp $EXTERNAL_NET any -> $HOME_NET [21,22,80] (msg:"SCAN nmap XMAS"; flow:stateless; flags:FPU,12; threshold: type both, track by_dst, count 3, seconds 10; reference:arachnids,30; classtype:attempted-recon; sid:1228; rev:7;)

```

b. *Rule FTP Access* pada /etc/snort/rules/ftp.rules

```

alert tcp $HOME_NET 21 -> $EXTERNAL_NET any (msg:"FTP access attempt"; content:"530 Login incorrect"; nocase; flow:from_server,established; classtype:attempted-admin; GID:1; SID:10000003; rev:001;)

```

```

alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP root
user access attempt"; content:"USER root"; nocase; GID:1;
SID:10000004; rev:001; classtype:attempted-admin;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP
anonymous access attempt"; content:"USER anonymous";
nocase; GID:1; SID:10000005; rev:001; classtype:attempted-
admin;)

```

c. *Rule SSH Access* pada `/etc/snort/rules/ssh.rules`

```

alert tcp $EXTERNAL_NET any -> $HOME_NET 22 (msg:"SSH login
attempt"; flow:to_server; flags:S; threshold:type
threshold, track by_src, count 4, seconds 60;
metadata:service ssh; classtype:misc-activity; GID:1;
sid:10000006; rev:001;)

```

d. *Rule Ddos* pada `/etc/snort/rules/ddos.rules`

```

alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (flags:S;
msg:"DdoS Detected"; flow:stateless; threshold: type both,
track by_dst, count 70, seconds 10; classtype:bad-unknown;
GID:1; sid:10000007; rev:001;)
alert udp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"DdoS
UDP"; flow:stateless; threshold:type both, track by_dst,
count 70, seconds 10; classtype:bad-unknown; GID:1;
sid:10000012; rev:001;)

```

4.2.6 Hasil Implementasi Laporan *Web Based* Sistem

Hasil implementasi ini merupakan hasil realisasi dari perancangan laporan berbasis *web* yang telah dibuat di dalam IDS yang sebenarnya. Hal ini bertujuan untuk memberikan informasi mengenai insiden yang terdeteksi dan informasi yang telah dikirimkan untuk administrator melalui aplikasi *instant messaging* Telegram. Laporan *web* tersebut didesain secara sederhana namun tetap menarik sehingga administrator dapat dengan mudah menerima informasi insiden yang terdeteksi. Adapun tampilan hasil implementasi sebagai berikut:

a. Halaman Beranda

Halaman beranda merupakan halaman yang pertama kali diakses oleh administrator. Pada halaman ini terdapat dua pilihan menu yaitu report dan base. Menu Report merupakan menu yang mengarah ke alamat `http://192.168.10.129/ids` yang menyajikan laporan khusus mengenai insiden yang dikirimkan ke administrator maupun laporan dalam bentuk grafik dan tabel. Sedangkan menu base akan mengarah ke antarmuka BASE

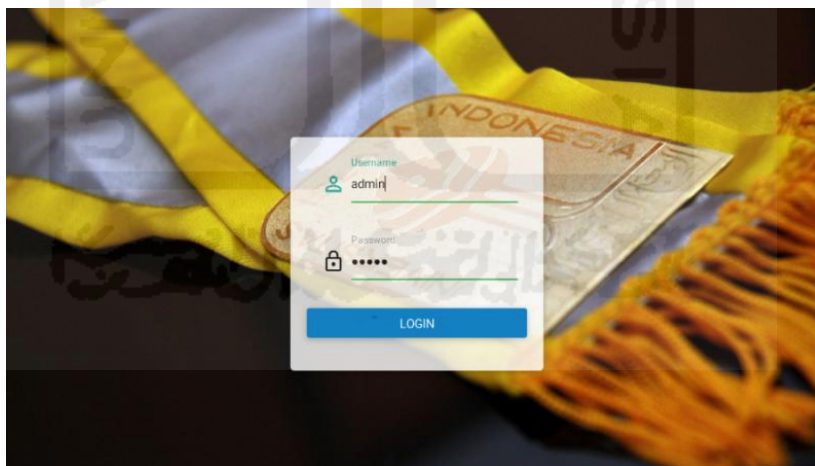
yang menyajikan informasi khusus insiden yang tercatat. Halaman dilihat pada Gambar 4.2



Gambar 4.2 Halaman Beranda

b. Halaman *Login*

Halaman *login* merupakan halaman dimana administrator harus memasukkan *username* dan *password* untuk dapat mengakses *dashboard* admin. Berikut halaman *login* dapat dilihat pada Gambar 4.3

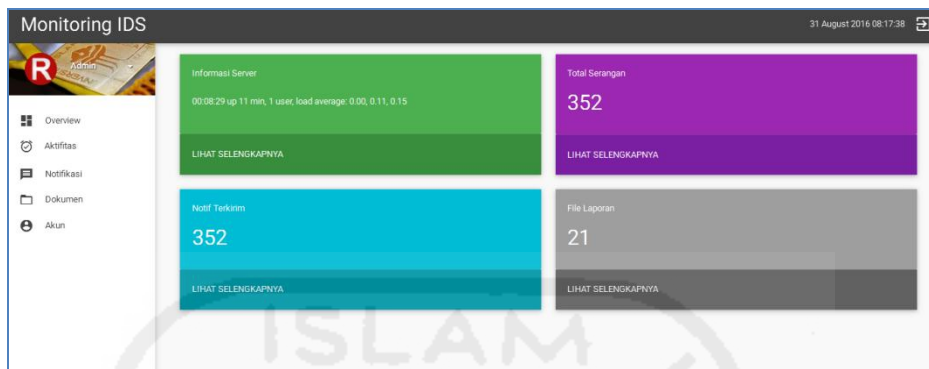


Gambar 4.3 Halaman *Login*

c. Halaman *Dashboard*

Pada halaman dashboard disajikan informasi mengenai status *server*, total insiden serangan yang terjadi, notifikasi yang terkirim, dan jumlah

laporan yang sudah diekspor ke dalam format PDF. Halaman dapat dilihat pada Gambar 4.4



Gambar 4.4 Halaman *Dashboard Admin*

d. Halaman Aktifitas

Halaman berikut merupakan halaman yang menampilkan aktifitas insiden serangan yang telah teridentifikasi. Pada halaman aktifitas ini, ditampilkan dalam bentuk grafik batang dan data dalam sebuah tabel yang menyajikan informasi mengenai serangan, klasifikasi serangan, *ip address* asal, *ip address* tujuan, *port* asal, *port* tujuan, dan waktu insiden terjadi.

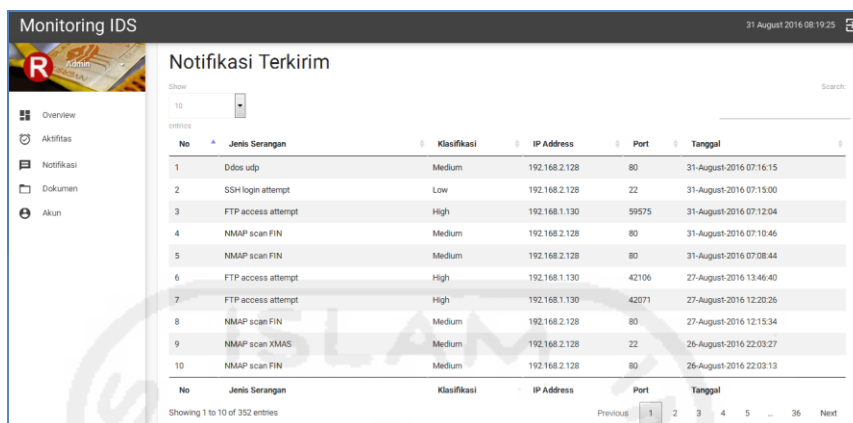
No	ID Tipe	Serangan	Klasifikasi	IP Asal	IP Tujuan	Port Asal	Port Tujuan	Waktu
1	521	Ddos udp	Medium	192.168.2.128	192.168.1.130	2436	80	31-August-2016 07:16:15
2	518	SSH login attempt	Low	192.168.2.128	192.168.1.130	45734	22	31-August-2016 07:15:00
3	515	FTP access attempt	High	192.168.2.128	192.168.1.130	21	59575	31-August-2016 07:12:04
4	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	40200	80	31-August-2016 07:10:46
5	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	64800	80	31-August-2016 07:08:44

Gambar 4.5 Halaman Aktifitas

e. Halaman Notifikasi

Notifikasi yang telah terkirim ke administrator dapat dilihat pada menu notifikasi. Pada menu ini disajikan informasi singkat mengenai jenis

serangan, klasifikasi serangan, *ip address*, *port* tujuan dan tanggal terjadinya insiden. Menu notifikasi dapat dilihat pada Gambar 4.6.

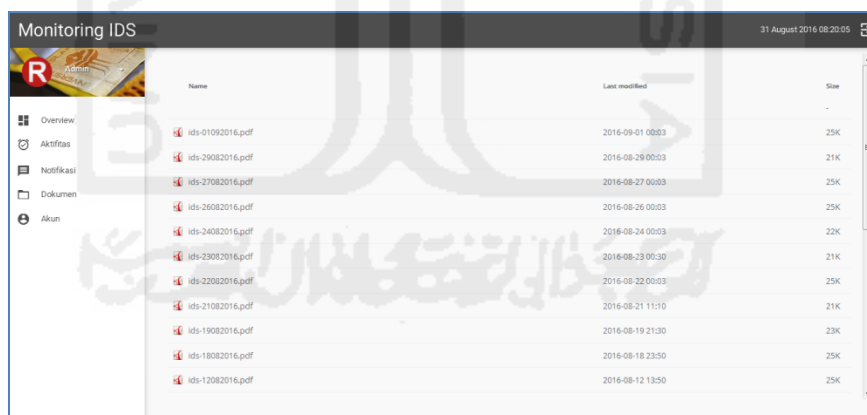


No	Jenis Serangan	Klasifikasi	IP Address	Port	Tanggal
1	Ddos udp	Medium	192.168.2.128	80	31-August-2016 07:16:15
2	SSH login attempt	Low	192.168.2.128	22	31-August-2016 07:15:00
3	FTP access attempt	High	192.168.1.130	59575	31-August-2016 07:12:04
4	NMAP scan FIN	Medium	192.168.2.128	80	31-August-2016 07:10:46
5	NMAP scan FIN	Medium	192.168.2.128	80	31-August-2016 07:08:44
6	FTP access attempt	High	192.168.1.130	42106	27-August-2016 13:46:40
7	FTP access attempt	High	192.168.1.130	42071	27-August-2016 12:20:26
8	NMAP scan FIN	Medium	192.168.2.128	80	27-August-2016 12:15:34
9	NMAP scan XMAS	Medium	192.168.2.128	22	26-August-2016 22:03:27
10	NMAP scan FIN	Medium	192.168.2.128	80	26-August-2016 22:03:13

Gambar 4.6 Halaman Notifikasi

f. Halaman Dokumen

Untuk mengetahui laporan yang sudah dihasilkan sistem dalam bentuk *file* digital PDF dapat dilihat pada menu dokumen. Pada menu ini hanya menampilkan *file* yang memiliki format PDF. Menu ini dapat dilihat pada Gambar 4.7

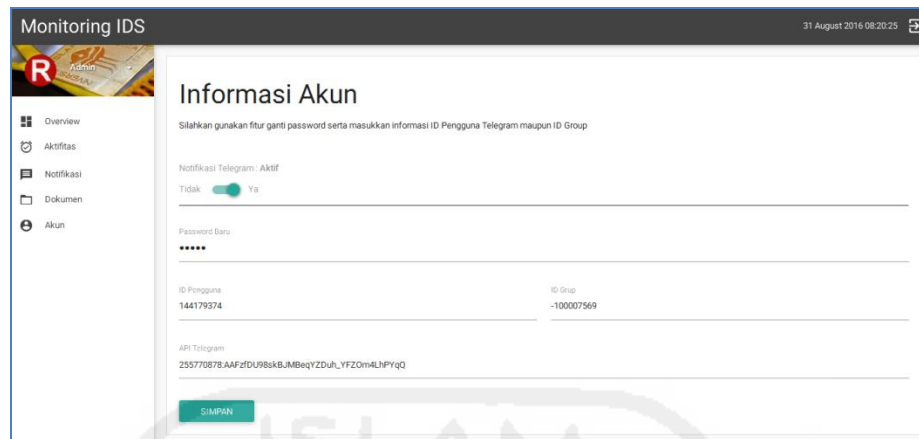


Name	Last modified	Size
ids-01092016.pdf	2016-09-01 00:03	25K
ids-29082016.pdf	2016-08-29 00:03	21K
ids-27082016.pdf	2016-08-27 00:03	25K
ids-26082016.pdf	2016-08-26 00:03	25K
ids-24082016.pdf	2016-08-24 00:03	22K
ids-23082016.pdf	2016-08-23 00:30	21K
ids-22082016.pdf	2016-08-22 00:03	25K
ids-21082016.pdf	2016-08-21 11:10	21K
ids-19082016.pdf	2016-08-19 21:30	23K
ids-18082016.pdf	2016-08-18 23:50	25K
ids-12082016.pdf	2016-08-12 13:50	25K

Gambar 4.7 Menu Dokumen

g. Halaman Akun

Untuk mengetahui maupun mengatur *username*, *password*, *id chat user* Telegram, *id grup chat* Telegram, dan *token* dapat memanfaatkan menu akun. Terdapat fitur untuk mengaktifkan maupun menonaktifkan pengiriman notifikasi. Menu akun dapat dilihat pada Gambar 4.8

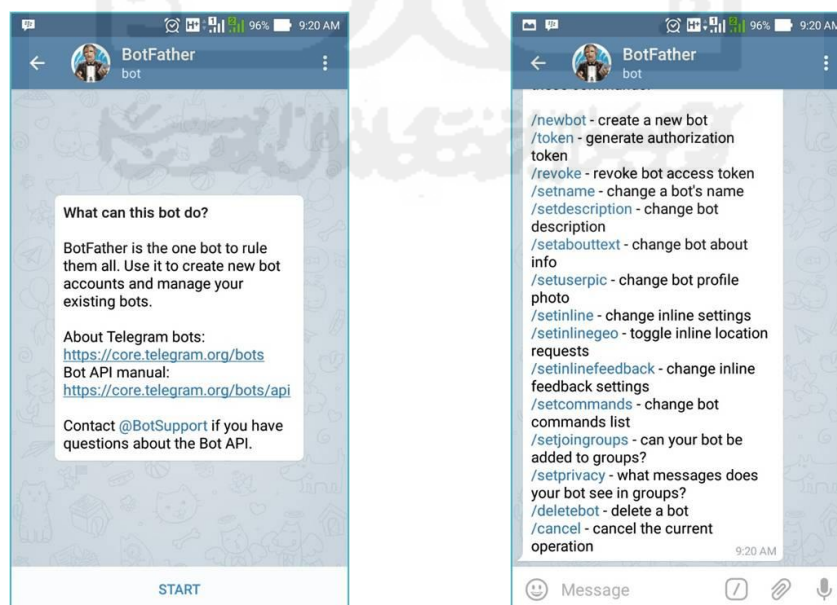


Gambar 4.8 Menu Akun

4.2.7 Implementasi Telegram Bot

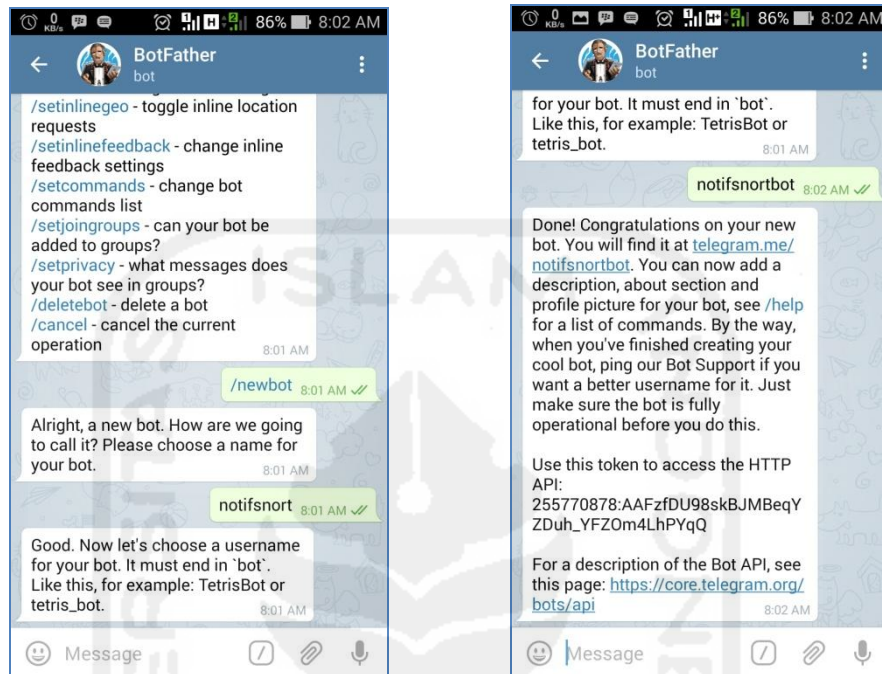
Telegram *bot* dapat dimanfaatkan sebagai mesin robot otomatis yang mampu menjembatani antara sistem dengan *user*. Dalam implementasinya, *user* harus memiliki akun Telegram kemudian melakukan *request* kepada @BotFather untuk mendapatkan *username bot*, *token*, *id chat user*, maupun *id chat group*. Berikut langkah-langkahnya:

- a. Melakukan pencarian id @BotFather kemudian, klik *START*. Selanjutnya untuk membuat *bot* pilih */newbot*. Alur pembuatan *bot* dapat dilihat pada Gambar 4.9



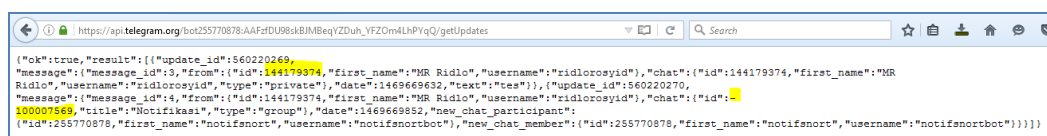
Gambar 4.9 Request Telegram Bot

b. Kemudian, *user* akan diminta untuk menentukan nama *bot* yang diinginkan. Setelah itu, *user* memasukkan id *bot*, misalnya *notifsnortbot*. Setiap *bot* akan memiliki *token* seperti pada Gambar 4.10 di bawah ini:



Gambar 4.10 Membuat Telegram Bot

c. Selain *token*, diperlukan *id chat user* maupun *id chat group* yang didapatkan dengan melakukan kirim *chat* kepada *bot* tersebut. Untuk melakukan *chat*, cari *bot* dipencarian kemudian memulai *chat*. Selain itu, *bot* dapat ditambahkan ke dalam grup. Sama halnya dengan *id chat user*, *id chat group* didapatkan setelah *user* mengirimkan obrolan di dalam grup. Kemudian *user* mengakses alamat https://api.telegram.org/bot255770878:AAFzfDU98skBJMBeqYZDuh_YFZOm4LhPYqQ/getUpdates. Hasil dari pengaksesan alamat tersebut didapatkan *id chat* yang dapat dilihat pada Gambar 4.11



Gambar 4.11 Mendapatkan ID chat user dan group

Informasi yang didapatkan dari *bot* tersebut mengenai nama *bot*, *username*, *token*, *id chat user*, dan *id chat group* kemudian disimpan dalam *database* notfinsnort pada tabel akun yang telah dibuat. Informasi tersebut dapat dilihat pada tabel 4.5

Tabel 4.5 Informasi Telegram *Bot*

Parameter	Value
Nama	notifsnort
Username	notifsnortbot
Token	bot255770878: AAFzfDU98skBJMBeqYZDuh_YFZOm4LhPYqQ
ID chat user	144179374
ID chat grup	-100007569

4.2.8 Implementasi *Trigger*

Trigger merupakan pemicu dimana akan mengeksekusi sebuah perintah. Pada implementasi sistem ini menggunakan *trigger* di dalam *database* snort maupun notfinsnort.

a. *Trigger* insertAktifitas

Pada tabel *acid_event* yang berada di *database* snort diberikan *trigger After Insert*. Jadi, ketika data baru telah masuk, maka *trigger* akan mengeksekusi untuk menambahkan data pada *database* notfinsnort tepatnya tabel aktifitas.

```
INSERT INTO notfinsnort.aktifitas
(id_tipe,signature,klasifikasi,ip_src,ip_dst,src_port,dst_p
ort,timestamp)
VALUES (NEW.signature, NEW.sig_name, NEW.sig_priority,
NEW.ip_src,NEW.ip_dst,NEW.layer4_sport,NEW.layer4_dport,NEW
.timestamp)
```

b. *Trigger* kirimNotif

Pada tabel *acid_event* yang berada di *database* snort diberikan *trigger Before Insert*. Jadi, ketika data akan masuk, maka *trigger* akan mengeksekusi *file* PHP untuk mengirimkan notifikasi.

```
BEGIN
DECLARE cmd CHAR(255);
DECLARE result CHAR(255);
```

```

SET cmd = CONCAT('/usr/bin/php ',
'/var/www/html/config/notif.php');
SET result = sys_exec(cmd);
END

```

c. *Trigger* insertNotif

Data yang dikirimkan ke pada administrator akan disimpan pada tabel `notif_terkirim`. *Trigger* berada pada tabel aktifitas dimana akan dijalankan pada kondisi *after insert*. Hal ini berguna untuk melihat informasi yang telah dikirimkan kepada administrator.

```

INSERT INTO notifsnotif.notif_terkirim
(tipe_serangan, klasifikasi, ip_address, port, tanggal)
VALUES (NEW.signature, NEW.klasifikasi, NEW.ip_src,
NEW.dst_port, NEW.timestamp)

```

4.2.9 Implementasi Crontab

Crontab merupakan tool yang berfungsi untuk menjalankan suatu pekerjaan sesuai penjadwalan dalam sistem. Pada penelitian ini, memanfaatkan crontab untuk membantu proses pengiriman notifikasi serta menghasilkan *file* laporan ke dalam format PDF. Perintah crontab tersebut adalah sebagai berikut:

```

* * * * * curl http://localhost/base/grafik.php
* * * * * ( sleep 15; curl http://localhost/base/grafik.php )
* * * * * ( sleep 30; curl http://localhost/base/grafik.php )
* * * * * ( sleep 30; curl http://localhost/base/grafik.php )
* * * * * ( sleep 45; curl http://localhost/base/grafik.php )

03 00 * * * /usr/bin/xvfb-run /usr/local/bin/wkhtmltopdf http://192.168.10.129/
laporan/laporan.php /var/www/html/ids/dokumen/ids-$(date +%d%m%Y).pdf

05 00 * * * /usr/bin/php /var/www/html/laporan/kirimLaporan.php

```

Pada baris 1-5, dilakukan eksekusi membuat koneksi untuk melakukan *refresh* pada alamat `http://192.168.10.129/base/grafik.php` setiap 15 detik sekali. Hal ini untuk membantu proses pengiriman notifikasi sesuai dengan *trigger* yang telah dibuat.

Selanjutnya pada baris kedua, sistem akan melakukan *generate file* menggunakan aplikasi `xvfb-run` dan `wkhtmltopdf` pukul 00:03. *File* tersebut

merupakan *file* PHP pada alamat `http://192.168.10.129/laporan/laporan.php` dan akan disimpan pada direktori `/var/www/html/ids/dokumen/` dengan format penamaan “ids-” dengan penambahan tanggal sesuai hari pembuatan file PDF tersebut.

Dari hasil *generate file* PDF tersebut, sesuai dengan baris ketiga akan dilakukan eksekusi *file* `kirimLaporan.php` untuk mengirimkan *attachment file* kepada administrator. Pengiriman laporan tersebut dilakukan setiap pukul 00.05 melalui aplikasi *instant messaging* Telegram.

4.2.10 Hasil Pengujian Serangan

Pengujian serangan dilakukan pada jaringan lokal sesuai dengan rancangan yang telah dibuat. Pengujian ini dilakukan terhadap beberapa layanan pada *server* yang dijadikan sebagai target.

a. Hasil Pengujian *Port Scanning*

Pengujian *port scanning* bertujuan untuk mendapatkan informasi mengenai *port* yang terbuka pada *server*. Pengujian menggunakan aplikasi nmap dapat dilihat pada Gambar 4.12.

```

root@kali:~# nmap -sF 192.168.1.130
Starting Nmap 6.46 ( http://nmap.org ) at 2016-08-31 07:10 WIB
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.1.130
Host is up (0.0013s latency).
Not shown: 997 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered ftp
22/tcp    open|filtered ssh
80/tcp    open|filtered http
Nmap done: 1 IP address (1 host up) scanned in 1.85 seconds

```

Gambar 4.12 Pengujian *Port Scanning*

Pada pengujian *port scanning* menggunakan *FIN scan* untuk mengetahui *port* yang terbuka. Pengujian ini dilakukan oleh *attacker* dengan *IP Address* 192.168.2.128 pada jam 07:10 WIB tanggal 31-08-2016. Hasil pengujian berhasil mendapatkan beberapa layanan yang memiliki *port* terbuka yaitu FTP, SSH, dan HTTP.

Di sisi lain, IDS telah aktif dan mendeteksi adanya intrusi dengan waktu 07:10:46 tanggal 08/31 dan SID 10000002 yang dilakukan oleh *IP Address* 192.168.2.128 ke *IP Address server* 192.168.1.130 port 80. Deteksi intrusi dapat dilihat pada Gambar 4.13

```
08/31-07:10:46.483538  [**] [1:10000002:1] Snort Alert [1:10000002:1] [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.2.128:40200 -> 192.168.1.130:80
```

Gambar 4.13 Deteksi *Port Scanning*

Setelah terdeteksi sebuah intrusi, administrator mendapatkan notifikasi melalui aplikasi *instant messaging* Telegram. Notifikasi tersebut dapat dilihat pada Gambar 4.14

```
Intrusi Terhadap Sistem!
IP Address : 192.168.2.128
Serangan : NMAP scan FIN
Port : 80
Klasifikasi : Medium
Waktu : 2016-08-31 07:10:46 7:10 AM
```

Gambar 4.14 Notifikasi *Port Scanning*

b. Hasil Pengujian FTP Akses

Pengujian FTP akses dilakukan dengan memasukkan sembarang *username* yaitu "tess" dengan target *IP Address* 192.168.1.130. Pengujian ini dapat dilihat pada Gambar 4.15.

```
root@kali:~# ftp 192.168.1.130
Connected to 192.168.1.130.
220 (vsFTPd 3.0.2)
Name (192.168.1.130:root): tess
331 Please specify the password.
Password:

530 Login incorrect.
Login failed.
ftp>
ftp> bye
221 Goodbye.
```

Gambar 4.15 Pengujian FTP Akses

Pada sistem IDS akan mendeteksi intrusi tersebut dikarenakan teridentifikasi ada yang mencoba mengakses FTP komputer target dengan IP Address 192.168.1.130/24 sehingga memberikan respon serangan sesuai dengan *rule*. Deteksi tersebut dapat dilihat pada Gambar 4.16

```
08/31-07:12:04.496313  [**] [1:10000003:1] Snort Alert [1:10000003:1] [**] [Classification: Attempted Administrator Privilege Gain] [Priority: 1] {TCP} 192.168.1.130:21 -> 192.168.2.128:59575
```

Gambar 4.16 Deteksi FTP Akses

Intrusi yang terdeteksi pada tanggal 08/31 pukul 07:12:04 dan memiliki SID 10000003. Sumber intrusi berasal dari *IP Address* 192.168.2.128 dengan target yaitu *IP Address* 192.168.1.130 *port* 21. Setelah terdeteksi, terjadi sebuah *trigger* yang mengirimkan notifikasi kepada administrator yang dapat dilihat pada Gambar 4.17

```
Intrusi Terhadap Sistem!
IP Address : 192.168.2.128
Serangan : FTP access attempt
Port : 21
Klasifikasi : High
Waktu : 2016-08-31 07:12:04 7:12 AM
```

Gambar 4.17 Notifikasi FTP Akses

c. Hasil Pengujian SSH Akses

Tahap pengujian SSH Akses dilakukan dengan melakukan *brute force*. *Attacker* mencoba dengan username “root” dan beberapa *password* yang tersimpan pada *file* pass.txt. Target serangan adalah *IP Address* 192.168.1.130 Pengujian SSH Akses dapat dilihat pada Gambar 4.18

```
root@kali:~# hydra -l root -P pass.txt 192.168.1.130 ssh
Hydra v7.6 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only

Hydra (http://www.thc.org/thc-hydra) starting at 2016-08-31 07:15:00
[DATA] 6 tasks, 1 server, 6 login tries (l:l/p:6), ~1 try per task
[DATA] attacking service ssh on port 22
1 of 1 target completed, 0 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2016-08-31 07:15:00
```

Gambar 4.18 Pengujian SSH *Brute Force*

Pada IDS mendeteksi adanya sebuah intrusi terhadap akses *port* 22 pada *IP Address* 192.168.1.130 yang merupakan target serangan. Intrusi terhadap SSH tersebut memiliki SID 10000006 dan berasal dari *IP Address* 192.168.2.128. Deteksi intrusi dapat dilihat pada Gambar 4.19

```
08/31-07:15:00.558107  [**] [1:10000006:1] Snort Alert [1:10000006:1] [**] [Classification: Misc activity] [Priority: 3] {TCP} 192.168.2.128:45734 -> 192.168.1.130:22
```

Gambar 4.19 Deteksi SSH *Brute Force*

Setelah terdeteksi adanya intrusi SSH *brute force* tersebut, maka administrator akan mendapatkan notifikasi singkat yang dapat dilihat pada Gambar 4.20



```
Intrusi Terhadap Sistem!
IP Address : 192.168.2.128
Serangan : SSH login attempt
Port : 22
Klasifikasi : Low
Waktu : 2016-08-31 07:15:00 7:15 AM
```

Gambar 4.20 Notifikasi SSH *Brute Force*

d. Hasil Pengujian Ddos *Attack*

Pengujian Ddos menggunakan aplikasi *hping3* pada komputer *attacker*. Target pengujian yaitu *IP Address* 192.168.1.130 dengan *port* 80 dan protokol UDP. Pengujian dapat dilihat pada Gambar 4.21

```
root@kali:~# hping3 -S -p 80 192.168.1.130 --udp --flood
HPING 192.168.1.130 (eth0 192.168.1.130): udp mode set, 28 headers + 0 data byte
s
hping in flood mode, no replies will be shown
^C
--- 192.168.1.130 hping statistic ---
24383 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Gambar 4.21 Pengujian Ddos *Attack*

Intrusi yang terdeteksi oleh IDS pada tanggal 08/31 pukul 07:16:15 dengan SID 10000006 berasal dari *IP Address* 192.168.2.128 dengan target 192.168.1.130 yang memiliki *port* 80. Hasil deteksi dari IDS dapat dilihat pada Gambar 4.22

```
08/31-07:16:15.488320 [**] [1:10000012:1] Snort Alert [1:10000012:1] [**] [Classification: Potentially Bad Traffic] [Priority: 2] {UDP} 192.168.2.128:2436 -> 192.168.1.130:80
```

Gambar 4.22 Deteksi Ddos Attack

Notifikasi akan dikirimkan ke administrator setelah intrusi tersebut tercatat oleh IDS. Notifikasi dikirimkan dengan pesan “Ddos udp” sesuai dengan *signature* yang terdeteksi oleh IDS. Pemberitahuan tersebut dapat dilihat pada Gambar 4.23

Intrusi Terhadap Sistem!
 IP Address : [192.168.2.128](#)
 Serangan : Ddos udp
 Port : 22
 Klasifikasi : Medium
 Waktu : 2016-08-31 07:16:15 7:16 AM

Gambar 4.23 Notifikasi Ddos Attack

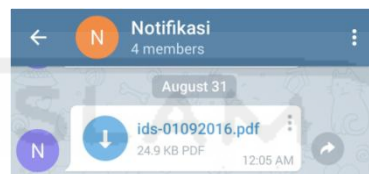
Pada antarmuka *web based* disajikan mengenai intrusi yang terdeteksi. Intrusi tersebut ditampilkan dalam bentuk grafik dan data pada tabel. Masing-masing intrusi ditampilkan sesuai dengan hasil deteksi. Hasil deteksi tersebut dapat dilihat pada Gambar 4.24

No	ID Tipe	Serangan	Klasifikasi	IP Asal	IP Tujuan	Port Asal	Port Tujuan	Waktu
1	521	Ddos udp	Medium	192.168.2.128	192.168.1.130	2436	80	31-August-2016 07:16:15
2	518	SSH login attempt	Low	192.168.2.128	192.168.1.130	45734	22	31-August-2016 07:15:00
3	515	FTP access attempt	High	192.168.2.128	192.168.1.130	21	59575	31-August-2016 07:12:04
4	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	40200	80	31-August-2016 07:10:46
5	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	64800	80	31-August-2016 07:08:44
6	515	FTP access attempt	High	192.168.2.128	192.168.1.130	21	42106	27-August-2016 13:46:40
7	515	FTP access attempt	High	192.168.2.128	192.168.1.130	21	42071	27-August-2016 12:20:26
8	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	47115	80	27-August-2016 12:15:34
9	520	NMAP scan XMAS	Medium	192.168.2.128	192.168.1.130	40114	22	26-August-2016 22:03:27
10	514	NMAP scan FIN	Medium	192.168.2.128	192.168.1.130	54308	80	26-August-2016 22:03:13

Gambar 4.24 Informasi Serangan Terdeteksi

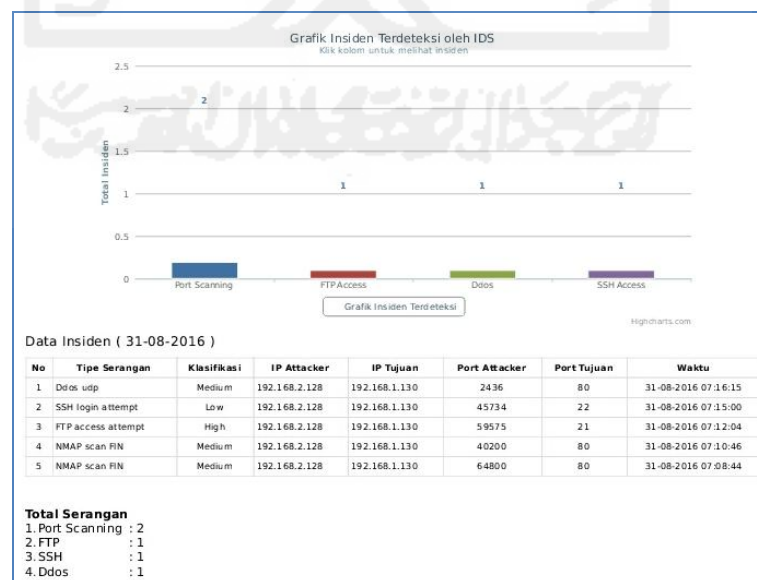
Setiap terdeteksi adanya intrusi, maka administrator akan mendapatkan notifikasi secara *real time* melalui aplikasi *instant messaging*

Telegram seperti pada gambar di atas. Tidak hanya itu, administrator juga mendapatkan laporan mengenai intrusi yang terdeteksi dalam bentuk dokumen digital yang memiliki format PDF. Laporan tersebut menampilkan informasi insiden yang terjadi dalam satu hari. Notifikasi laporan digital tersebut dapat dilihat pada Gambar 4.25



Gambar 4.25 Notifikasi Laporan Intrusi

Dokumen laporan tersebut dapat diunduh oleh administrator untuk mempermudah dalam mengidentifikasi serangan yang sering terjadi pada *server*. Laporan tersebut menampilkan grafik secara kuantitas dan informasi dalam bentuk tabel mengenai tipe serangan, klasifikasi serangan, *ip address attacker*, *ip address tujuan*, *port asal*, *port tujuan*, dan waktu insiden terjadi. Laporan digital tersebut dihasilkan dalam *interval* satu hari sebelum dikirimkan ke administrator. Dokumen tersebut dapat dilihat pada Gambar 4.26



Gambar 4.26 Laporan Insiden Terdeteksi

4.3 Hasil Akurasi Deteksi Intrusi

Dari aktifitas pengujian terhadap server, didapatkan waktu terhadap masing-masing aktifitas mulai dari penyerangan, deteksi, hingga terkirimnya notifikasi. Hasil catatan waktu tersebut digunakan untuk mengukur tingkat akurasi kecepatan deteksi hingga terkirimnya *alert*. Data tersebut dapat dilihat pada Tabel 4.6

Tabel 4.6 Tingkat Akurasi Waktu

No	Tipe Serangan	Tingkat Akurasi Waktu (timestamp)		
		Awal Serangan	Terdeteksi	Terkirim
1	<i>Port Scanning</i>	07:10:46	07:10:46	07:10:56
2	<i>FTP Bad login</i>	07:12:01	07:12:04	07:12:26
3	<i>SSH Brute force</i>	07:15:00	07:13:00	07:13:25
4	<i>Ddos Attack</i>	07:16:15	07:14:15	07:14:27

Tingkat akurasi waktu dihitung dari selisih waktu terdeteksi dan awal serangan. Dari selisih tersebut didapatkan rata-rata kecepatan deteksi IDS. Selain itu, kecepatan notifikasi terkirim juga didapatkan dari selisih waktu terkirim dengan waktu deteksi. Tabel selisih waktu dapat dilihat pada Tabel 4.7

Tabel 4.7 Selisih Waktu Serangan

No	Tipe Serangan	Waktu (detik)	
		Selisih serangan dan deteksi	Selisih terkirim dan terdeteksi
1	<i>Port Scanning</i>	0	10
2	<i>FTP Bad login</i>	3	12
3	<i>SSH Brute force</i>	0	25
4	<i>Ddos Attack</i>	0	12
Total		3	59
Rata-rata		0.75	29.4

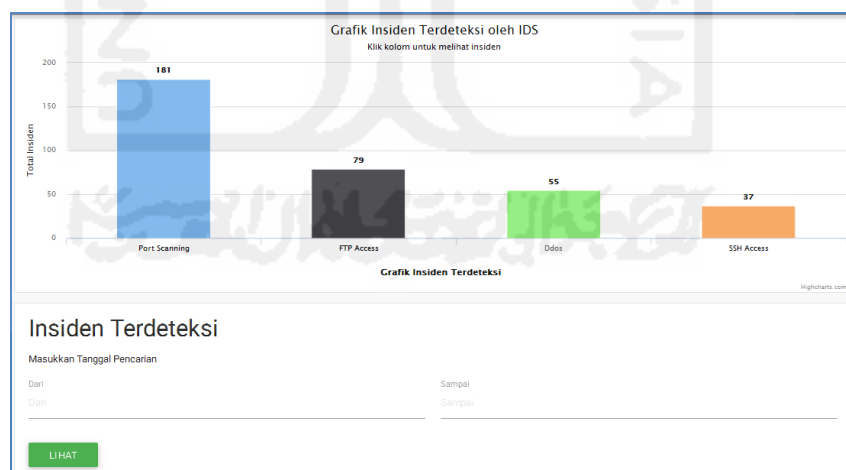
Data hasil perhitungan pada tabel di atas, pada pengujian serangan *port scanning* dan FTP akses memerlukan waktu deteksi selama 3 detik. Namun, pada serangan SSH dan Ddos waktu deteksi yaitu tidak lebih dari 1 detik setelah dilakukan penyerangan. Dari data tersebut, rata-rata waktu yang digunakan untuk proses deteksi yaitu selama 0.75 detik.

Di sisi lain, selisih waktu antara terdeteksi dan notifikasi terkirim menjadi tolak ukur tingkat akurasi *alert*. Selisih waktu tersebut dapat dilihat pada tabel 4.5 dimana selisih paling banyak yaitu SSH akses dengan waktu 25 detik. Rata-rata notifikasi terkirim dalam jangka waktu 29.5 detik setelah IDS mendeteksi adanya intrusi. Dari hasil pengujian didapatkan informasi intrusi yang terdeteksi oleh IDS yang dapat dilihat pada tabel 4.8

Tabel 4.8 Informasi Serangan Terdeteksi

No	Waktu	Asal Serangan	Tipe Serangan	
			Port	Layanan
1	31-08-2016 07:10:46	192.168.2.128	80	Web Server
2	31-08-2016 07:12:04	192.168.2.128	21	FTP
3	31-08-2016 07:13:00	192.168.2.128	22	SSH
4	31-08-2016 07:14:15	192.168.2.128	80	Web Server

Dari serangkaian pengujian didapatkan informasi mengenai insiden yang terjadi. IDS mendeteksi terdapat *attacker* dengan *IP Address* 192.168.2.128 melakukan serangan terhadap server yang dijaga oleh IDS. *Attacker* melakukan uji coba serangan pada tiga layanan yaitu *web server*, FTP, dan SSH. Hasil deteksi semua serangan secara kuantitas dapat dilihat Gambar 4.27



Gambar 4.27 Grafik Kuantitas Serangan

Informasi serangan disajikan dalam bentuk tabel sesuai dengan penilaian hasil pengujian sistem dimana sistem berjalan dengan baik atau tidak. Secara rinci, hasil pengujian dapat dilihat pada tabel 4.9.

Tabel 4.9 Hasil Pengujian Sistem

No	Skenario Pengujian Sistem	Uji Coba	Hasil yang diharapkan	Hasil pegujian sistem	Kesimpulan
1	<i>Port Scanning</i>	NMAP FIN scan	Terdeteksi	Terdeteksi	Berhasil
2	FTP	FTP <i>Bad login</i>	Terdeteksi	Terdeteksi	Berhasil
3	SSH	SSH <i>Brute force</i>	Terdeteksi	Terdeteksi	Berhasil
4	Ddos	Ddos UDP 80	Terdeteksi	Terdeteksi	Berhasil

Dari hasil pengujian sistem pada tabel di atas, seluruh pengujian mendapatkan hasil yang sesuai yang diharapkan. Sistem dapat mendeteksi pengujian yang dilakukan oleh *attacker*. Pendeteksian serangan sesuai dengan aturan yang dibuat mulai dari *port scanning*, FTP akses, SSH akses, dan Ddos. Sistem yang dibangun telah diuji dan dapat mendeteksi adanya intrusi.

Semakin berkembangnya teknologi, maka metode penyerangan akan semakin beragam. Penambahan *rule* terhadap intrusi akan memberikan dampak yang positif terhadap keamanan server maupun *host* yang dilindungi. Dengan begitu, pekerjaan Administrator akan mudah dalam menangani server.

Selain dapat mendeteksi intrusi, sistem diharapkan mampu melakukan tindakan pencegahan atau dinamakan *Intrusion Prevention System (IPS)*. Intrusi yang dianggap bisa membahayakan sistem dapat dengan mudah dicegah sehingga tidak mengganggu kinerja dan integritas data di dalam *server*.

Pemberitahuan mengenai intrusi juga menjadi hal yang penting bagi administrator untuk mengetahui keadaan *server* saat itu juga. Adanya *delay* pada server mengakibatkan administrator mendapatkan notifikasi lebih dari 1 pesan, meskipun demikian data yang terkirim tetap sama dan tidak berubah. Selain itu, perlu adanya interaksi antara IDS dengan administrator dengan memanfaatkan Telegram *bot* dimana tidak hanya berjalan satu arah yaitu sistem hanya memberikan notifikasi. Pemanfaatan Telegram *bot* akan lebih efektif apabila memanfaatkan *webhook* dimana akan ada interaksi Administrator tanpa perlu melihat langsung monitor sistem.