

## Implementasi Iptables Firewall dan Intrusion Detection System Untuk Mencegah Serangan DDoS Pada Linux Server

1) Theodorus Kristian Widiyanto

Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Jl. Diponegoro 52-60 Salatiga, Jawa Tengah, Indonesia  
E-Mail: theodoruskristianwidiyanto@gmail.com

2) Wiwin Sulistyono

Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Jl. Diponegoro 52-60 Salatiga, Jawa Tengah, Indonesia  
E-Mail: wiwinsulistyo@uksw.edu

### ABSTRACT

Security on computer networks is currently a matter that must be considered especially for internet users because many risks must be borne if this is negligent of attention. Data theft, system destruction, and so on are threats to users, especially on the server-side. DDoS is a method of attack that is quite popular and is often used to bring down servers. This method runs by consuming resources on the server computer so that it can no longer serve requests from the user side. With this problem, security is needed to prevent the DDoS attack, one of which is using iptables that has been provided by Linux. Implementing iptables can prevent or stop external DDoS attacks aimed at the server.

**Keyword : Linux, Iptables, DDoS attack, Network, Security**

### PENDAHULUAN

Perkembangan teknologi pada saat ini melaju dengan sangat pesat, kita banyak dimudahkan dalam melakukan berbagai hal. Teknologi sudah merambah diberbagai aspek kehidupan mulai dari ekonomi, sosial, serta pendidikan sehingga segala sesuatunya serba dimudahkan oleh teknologi itu sendiri. Teknologi pasti tak akan lepas dari yang namanya jaringan internet, dikarenakan untuk dapat mengakses data maupun berkomunikasi jaringan internet itu sendiri sangat diperlukan. Dengan adanya komunikasi data melalui jaringan internet maka akan timbul sebuah masalah yaitu keamanan jaringan itu sendiri salah satunya yaitu keamanan. Keamanan jaringan merupakan aspek pertahanan suatu jaringan komputer [1]. Sangatlah penting peran dari keamanan jaringan ini sendiri karena resiko tercurinya data maupun akses yang tidak dikehendaki dalam sebuah jaringan sangat mungkin terjadi.

Maka dari itu diperlukan sebuah pengamanan salah satunya menggunakan *firewall*. *Firewall* merupakan salah satu pelindung yang dibutuhkan untuk mendapatkan akses yang aman ketika berhubungan dengan jaringan komputer, baik dari luar (internet) maupun dari dalam (intranet) dengan cara membuat aturan tertentu [2]. Secara konsep, *firewall* didefinisikan sebagai komponen yang berada diantara sebuah jaringan dengan internet dan tugasnya memblok ataupun memberi akses diantara keduanya [3]. Pengaplikasian *firewall* ini sendiri sangatlah beragam, salah satunya adalah *packet filtering*. *Packet filtering* adalah cara untuk mengawasi

jalannya lalu lintas data yang terjadi pada suatu jaringan. Fitur yang dapat digunakan dari *firewall* untuk *packet filtering* adalah *iptables* pada *linux*. *Iptables* ini sendiri berfungsi dalam memblokir akses menuju server sehingga *request* yang terindikasi sebagai serangan DDoS dapat diredam pergerakannya, dengan ini maka server tidak akan mengalami kehabisan sumber daya lagi dan dapat menjalankan fungsinya dengan maksimal [4]. Server merupakan penyedia layanan pusat yang bertugas mengolah data pada satu jaringan, data-data yang didapat merupakan *request* dari *client* yang selanjutnya akan diolah server untuk selanjutnya memberikan *feedback* kepada *client* atas *request* yang sudah dikirimkan [5]. Untuk menjaga kenyamanan server, biasanya *network administrator* memonitor server tersebut serta melihat kinerja yang ditampilkan, sekaligus memperhatikan apabila ada aktivitas yang mencurigakan yang terjadi di server untuk selanjutnya dilakukan tindakan yang lebih lanjut yang biasanya dapat dilakukan dengan menerapkan *Intrusion Detection System (IDS)*. IDS itu sendiri adalah sebuah sistem yang digunakan untuk mendeteksi adanya aktivitas pada sebuah jaringan [6]. Dari sini akan terlihat semua usaha-usaha penyusup pada sebuah sistem dengan melakukan pengamatan secara *real-time*.

### BAHAN PENELITIAN

Untuk melakukan pengujian maka dibutuhkan beberapa komponen yaitu *server target*, *tools* yang digunakan untuk penyerangan, dan juga *tools* untuk memantau lalu lintas data. Komputer penyerang menggunakan Sistem Operasi

Windows 10 dengan menggunakan *tools* LOIC, sedangkan *server* menggunakan Sistem Operasi Linux Ubuntu Server 16.04. LOIC banyak digunakan dalam melakukan serangan DDoS karena sangat mudah digunakan dengan *interface* yang cukup sederhana.



Gambar 1. LOIC (Sumber : netsparker.com)

Penyerangan menggunakan *tools* ini menargetkan *server* dengan menyantumkan IP *server target* tersebut serta menentukan *port* yang digunakan dalam penyerangan. Secara definisi, *port* adalah sebuah mekanisme yang mengizinkan adanya koneksi antara komputer dengan perangkat jaringan lainnya [7]. LOIC ini memiliki fitur yang cukup banyak dalam skema penyerangan, mulai dari *timeout* serangan yang akan diluncurkan, *port* yang akan dituju, jumlah *threads* yang akan dikirimkan dalam satu waktu dan lain sebagainya. Panel-panel yang tersedia juga cukup memudahkan dalam melakukan penyerangan mulai dari target yang akan dituju hingga status serangan yang sedang dilancarkan. Dan yang terakhir adalah *Wireshark*, *tools* ini digunakan untuk memantau jalannya paket data saat dilakukan proses serangan DDoS terhadap *server*.

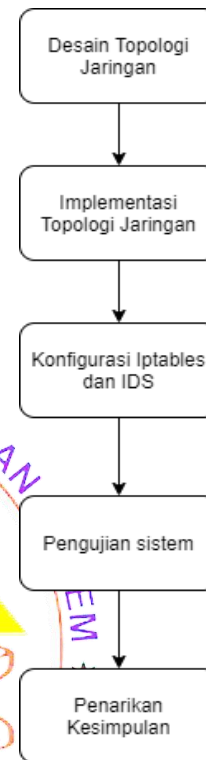


Gambar 2. Wireshark (Sumber : freecodecamp.org)

*Wireshark* memiliki fitur yang cukup lengkap dalam keperluan merekam lalu lintas data. Pada saat perekaman akan disajikan segala data yang berkaitan dengan proses ini, mulai dari IP *address* baik dari asal maupun tujuan, *port* yang terlibat, jenis komunikasi datanya, dan lain sebagainya.

## METODE PELAKSANAAN

Dalam pengimplementasian *iptables* untuk mencegah serangan DDoS maka dibutuhkan tahapan-tahapan agar penelitian dapat berjalan dengan baik. Beberapa tahapan yang akan dilakukan adalah seperti gambar 3 sebagai berikut.



Gambar 3. Tahap Perancangan

Tahapan Penelitian sebagai berikut:

- Desain Topologi Jaringan  
Pada tahap pertama dilakukan desain topologi terlebih dahulu karena tahap ini sangat penting dan mendasar serta mempengaruhi pengujian yang akan dilakukan.
- Implementasi Topologi Jaringan  
Setelah melakukan desain selanjutnya dengan mengimplementasikan desain topologi yang sudah dibuat,
- Konfigurasi *Iptables* dan IDS  
kemudian dilakukan konfigurasi *iptables* serta IDS yang akan dipakai dan juga *rules* yang diperlukan untuk melakukan pengujian. *Iptables* merupakan fitur dari *firewall* yang cukup *powerfull* dan memiliki cukup banyak fitur, secara garis besar *iptables* memiliki fitur antara lain *mentracking* keluar masuknya paket dan membatasi koneksi yang sudah diatur untuk tujuan keamanan [8]. *Rules* itu sendiri adalah sebuah teknis atau skema yang dilakukan oleh *network administrator* untuk dapat mengontrol kerja dari *firewall* itu sendiri [9]. Selain mengontrol kerja *firewall*, dengan *rules* ini maka *network administrator*

dapat memonitor kinerja perangkat-perangkat jaringan yang digunakan.

d. Pengujian Sistem

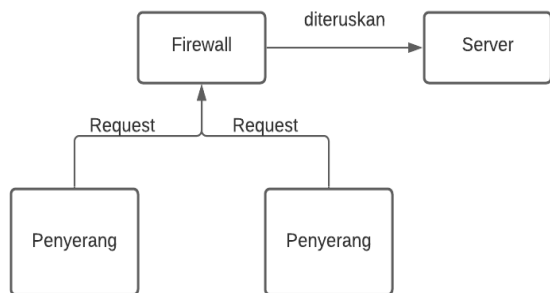
Kemudian setelah semua telah terkonfigurasi selanjutnya dilakukan pengujian terhadap sistem yang sudah dikonfigurasi, dilakukan simulasi penyerangan terhadap jaringan yang sudah disusun.

e. Penarikan Kesimpulan

Dan langkah terakhir setelah semua data yang diperlukan sudah tercapai maka dilakukan penarikan kesimpulan atas semua yang telah disusun, apakah masih membutuhkan perbaikan dan pengembangan untuk pengembangan kedepan.

DDoS merupakan model serangan yang ditujukan kepada server secara langsung dengan tujuan menjatuhkan server dengan cara memberikan request yang sangat besar sehingga server tidak lagi mampu untuk menanggapi request lain karena kehabisan sumber daya dan lain sebagainya [10]. Serangan DDoS ini datang secara acak dan tiba-tiba sehingga diperlukan keamanan yang baik untuk menangkang jenis serangan ini. Serangan DDoS biasanya juga dilakukan secara bersama-sama agar tujuan penyerang dapat tercapai, karena dengan makin banyaknya penyerang maka server yang menjadi target akan lebih mudah jatuh.

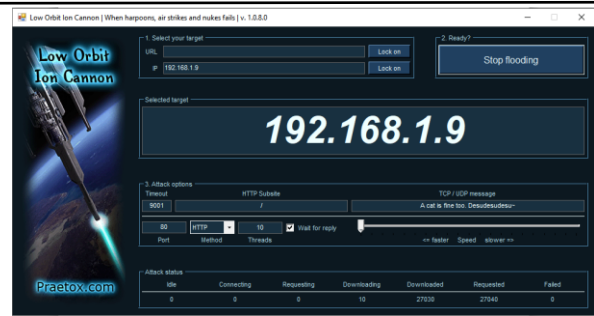
Penelitian dimulai dengan melihat proses serangan DDoS secara umum dan yang sering terjadi. Biasanya serangan dilakukan oleh beberapa penyerang secara bersama-sama, hal ini dinilai lebih efektif dalam menjatuhkan server. Serangan menuju server, pertama akan menghadapi firewall terlebih dahulu sehingga pada firewall ini dapat ditetapkan rules tertentu agar mencegah serangan yang terus menerus. Setelah sampai pada firewall, paket yang datang secara besar-besaran akan diblok arusnya, hal ini dilakukan agar server tidak down saat dilakukan penyerangan. Gambarnya dapat dilihat di gambar 4 sebagai berikut.



Gambar 4. Gambaran Proses Serangan

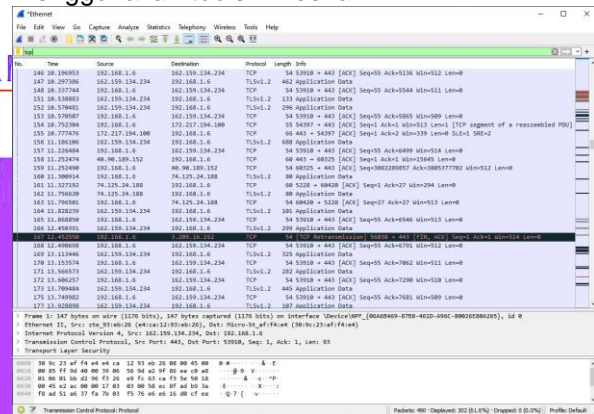
HASIL DAN PEMBAHASAN

Pertama serangan dilakukan terhadap server dengan IP address 192.168.1.9 melalui port 80 dan metode serangannya HTTP, dapat dilihat pada gambar berikut.



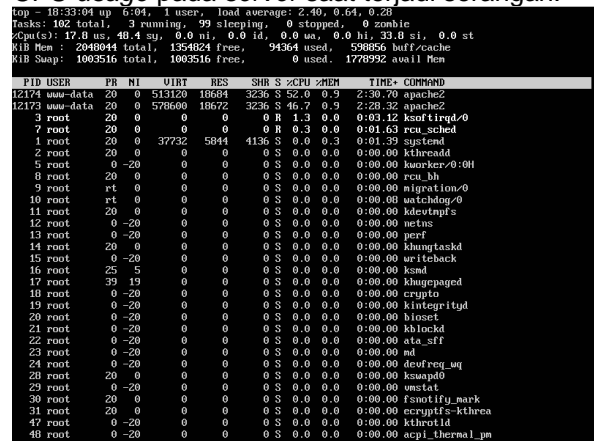
Gambar 5. Serangan Menggunakan LOIC

Pada gambar 5 terlihat bahwa serangan menuju server dengan IP 192.168.1.9 melalui port 80 sukses dilakukan. Kemudian coba memantau pergerakan data dengan menggunakan tools Wireshark.



Gambar 6. Monitoring Dengan Wireshark

Pada gambar 6 dapat dilihat bahwa pergerakan serangan terhadap server benar terjadi dan aktivitas request berlangsung secara bertubi-tubi menuju server. Kemudian cek kondisi CPU usage pada server saat terjadi serangan.



Gambar 7. Kondisi Server Awal

Dapat dilihat pada gambar 7 yang mana CPU usage langsung naik ke 52% padahal serangan baru dilakukan oleh satu penyerang saja dan kondisi tersebut cukup mengganggu aktivitas server.

Setelah proses serangan yang telah dilakukan, selanjutnya dilakukan proses antisipasi terhadap serangan tersebut dengan

memasukkan beberapa *rules* pada *iptables* supaya serangan DDoS selanjutnya dapat ditanggulangi.

**Rules pertama :**

```
iptables -A INPUT -p tcp --dport 80 -m state --state NEW -m limit --limit 100/second --limit-burst 100 -j ACCEPT
```

**Penjelasan rules pertama :**

1. -p tcp (menjelaskan protokol yang digunakan yaitu tcp)
2. --dport 80 (port yang secara spesifik akan dilimit traffiknya yaitu port 80)
3. -m state NEW (rules ini ditujukan untuk koneksi baru)
4. --limit 100/second (menjelaskan bahwa dalam satu detik hanya diperbolehkan 100 trafik baru yang mengakses server)
5. --limit-burst (ledakan traffiknya dicegah hingga diangka 100 saja)

**Rules kedua :**

```
iptables -A INPUT -m state --state RELATED,ESTABLISHED -m limit --limit 100/second --limit-burst 100 -j ACCEPT
```

**Penjelasan rules kedua :**

Rules kedua tidak jauh berbeda dengan rules pertama, namun bedanya rules ini tertuju untuk semua *port*, sehingga memastikan *port* lain juga terlindungi dari serangan ini.

**Rules ketiga :**

```
iptables -A INPUT -p tcp -m state --state NEW -j LOG --log-prefix 'DDoS Detected'
```

**Penjelasan rules ketiga :**

Rules ketiga digunakan untuk mendeteksi adanya serangan DDoS dengan memasukkan peringatan tersebut ke dalam *log iptables* dan peringatannya berupa 'DDoS Detected'.

**Rules keempat :**

```
Iptables -A INPUT -p tcp -m state --state NEW -j DROP
```

**Penjelasan rules keempat :**

Rules keempat bertujuan memblokk serangan yang dimaksud dalam rules pertama dan kedua sehingga serangan DDoS tersebut bisa diredam dan tidak mengganggu server kembali.

```
root@Ubuntu:/home/saneoth# iptables -F
root@Ubuntu:/home/saneoth# iptables -A INPUT -p tcp -n state --state NEW -n limit --limit 100/second --limit-burst 100 -j ACCEPT
root@Ubuntu:/home/saneoth# iptables -A INPUT -n state --state RELATED,ESTABLISHED -n limit --limit 100/second --limit-burst 100 -j ACCEPT
root@Ubuntu:/home/saneoth# iptables -A INPUT -p tcp -n state --state NEW -j LOG --log-prefix 'DoS Detected'
root@Ubuntu:/home/saneoth# iptables -A INPUT -p tcp -n state --state NEW -j DROP
root@Ubuntu:/home/saneoth# iptables-save
# Generated by iptables-save v1.6.0 on Thu Apr 22 19:39:13 2021
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
- A INPUT -p tcp -n state --state NEW -n limit --limit 100/sec --limit-burst 100 -j ACCEPT
- A INPUT -p tcp -n state --state NEW -j LOG --log-prefix 'DoS Detected'
- A INPUT -p tcp -n state --state NEW -j DROP
COMMIT
# Completed on Thu Apr 22 19:39:13 2021
root@Ubuntu:/home/saneoth# iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination
ACCEPT tcp -- anywhere anywhere state NEW limit: avg 100/sec burst 100
ACCEPT all -- anywhere anywhere state RELATED,ESTABLISHED limit: avg 100/sec burst 100
LOG tcp -- anywhere anywhere state NEW LOG level warning prefix 'DoS Detected'
DROP tcp -- anywhere anywhere state NEW
Chain FORWARD (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
root@Ubuntu:/home/saneoth#
```

Gambar 8. Konfigurasi Iptables

Pada gambar 8 menunjukkan tampilan pada saat konfigurasi rules pada iptables, terlihat rules tersebut terdapat pada chain INPUT karena rules yang sudah dibuat ditujukan untuk data yang mengarah kepada server.

```
top - 19:48:40 up 7:19, 1 user, load average: 0.07, 0.03, 0.01
Tasks: 102 total, 1 running, 101 sleeping, 0 stopped, 0 zombie
CPU(s): 1.4 us, 3.1 sy, 0.0 ni, 93.5 id, 0.0 wa, 0.0 hi, 2.1 si, 0.0 st
Mem: 2270860 total, 2270860 free, 71912 used, 699972 buff/cache
Mem Swap: 1003516 total, 1003516 free, 0 used, 1806696 avail Mem

  PID USER      PR  NI  TID    D    S  SHN S %CPU %MEM    TIME+  COMMAND
 12174 www-data  20   0 49424  R    S   226 5  0.0  0.3  2:47.53 apache2
12173 www-data  20   0 561568  R    S   226 5  4.0  0.4  2:44.64 apache2
   1 root      20   0 37732  R    S   226 5  0.0  0.3  0:01.40 systemd
   2 root      20   0 0       S    S   226 0  0.0  0.0  0:00.00 kthreadd
   3 root      20   0 0       S    S   226 0  0.0  0.0  0:03.49 ksoftirqd/0
   5 root      0 -20  0       S    S   226 0  0.0  0.0  0:00.00 kworker/0:0H
   7 root      20   0 0       S    S   226 0  0.0  0.0  0:02.00 rcu_sched
   8 root      20   0 0       S    S   226 0  0.0  0.0  0:00.00 rcu_bh
   9 root      rt   0 0       S    S   226 0  0.0  0.0  0:00.00 migration/0
  10 root    rt   0 0       S    S   226 0  0.0  0.0  0:00.10 watchdog/0
  11 root    20   0 0       S    S   226 0  0.0  0.0  0:00.00 kdevtmpfs
  12 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 netns
  13 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 perf
  14 root    20   0 0       S    S   226 0  0.0  0.0  0:00.00 khungtaskd
  15 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 unttback
  16 root    25   5 0       S    S   226 0  0.0  0.0  0:00.00 ksm
  17 root    39  19 0       S    S   226 0  0.0  0.0  0:00.00 khugepaged
  18 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 crypto
  19 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 kintegrityd
  20 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 bioset
  21 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 kblockd
  22 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 ata_sff
  23 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 md
  24 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 devfreq_wq
  28 root    20   0 0       S    S   226 0  0.0  0.0  0:00.00 kswapd0
  29 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 smitd
  30 root    20   0 0       S    S   226 0  0.0  0.0  0:00.00 fanotify_mark
  31 root    20   0 0       S    S   226 0  0.0  0.0  0:00.00 ecryptfs-kthrea
  47 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 kthrotld
  48 root    0 -20  0       S    S   226 0  0.0  0.0  0:00.00 acpi_thermal_pm
```

Gambar 9. Kondisi Server Akhir

Setelah dimasukkannya rules iptables pada server dilakukan kembali serangan yang sama, dan pada gambar 9 terlihat CPU usage pada server tidak semasih seperti sebelum diterapkannya rules pada server. Terlihat hanya 5% pada CPU usage yang berarti server tidak terlalu terbebani walaupun dalam kondisi serangan DDoS.

```
Apr 22 19:48:50 Ubuntu kernel: [26414.3288491] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=63252 DF FR
DDO-TCP SPT-53133 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26414.328956] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=63257 DF FR
DDO-TCP SPT-53134 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.392430] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64100 DF FR
DDO-TCP SPT-53231 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.392922] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64113 DF FR
DDO-TCP SPT-53232 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.393169] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64125 DF FR
DDO-TCP SPT-53235 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.393691] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64131 DF FR
DDO-TCP SPT-53236 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.393929] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64137 DF FR
DDO-TCP SPT-53237 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.394171] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64141 DF FR
DDO-TCP SPT-53238 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
Apr 22 19:48:50 Ubuntu kernel: [26415.394673] DoS Detected!IN=engp3s3 OUT= MAC=08:00:27:93:48:59:30:19
c:23:af:f4:e4:08:00 SRC=192.168.1.6 DST=192.168.1.9 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=64154 DF FR
DDO-TCP SPT-53240 DPT=80 WINDOW=64240 RES=0x00 SYN URG=0
root@Ubuntu:/home/saneoth#
```

Gambar 10. Log Server

Setelah proses serangan dilakukan selanjutnya cek *log* pada *firewall server*. Pada gambar 10 terlihat bahwa saat serangan terjadi, muncul peringatan dari *iptables* yang menunjukkan adanya serangan DDoS. Peringatan tersebut berupa tulisan 'DDoS Detected' sesuai dengan apa yang telah dibuat pada saat proses pembuatan *rules*. Peringatan ini berguna untuk analisa lebih mendalam dan juga tindakan selanjutnya yang harus dilakukan oleh para *network administrator*.

Tabel 1. Perbandingan CPU Usage

Komponen	Sebelum	Sesudah
CPU Usage	52.0%	5.0%

Pada tabel 1 terlihat bahwa ada perubahan setelah diterapkannya *rules* pada *iptables* dari yang sebelumnya 52% menjadi 5%.

Tabel 2. Perbandingan Memory Usage

Komponen	Sebelum	Sesudah
Memory Usage	0.9%	0.3%

Begitu juga dengan *Memory Usage* walaupun perubahannya tidak terlalu besar namun cukup menunjukkan bahwa *rules* berjalan dengan cukup baik.

## KESIMPULAN

Berdasarkan hasil penelitian maka dapat disimpulkan bahwa:

1. *Tools* LOIC terbukti dapat digunakan untuk melakukan serangan DDoS terhadap *server* dengan cara memasukkan *IP address server* serta *port* yang akan digunakan untuk proses serangan.
2. *Wireshark* dapat merekam aktivitas paket data yang lewat sehingga dapat terlihat lalu-lintas apa saja yang sedang terjadi pada saat proses serangan DDoS.
3. Serangan DDoS yang dilakukan pada saat pengujian terbukti dapat menaikkan *load* CPU *server* yang mengakibatkan *server* menjadi lebih berat.
4. Pengimplementasian *rules* pada *iptables* cukup efektif dalam mencegah serangan DDoS yang datang.
5. CPU *usage* sebelum diterapkannya *rules* pada saat diserang adalah sebesar 52% oleh satu penyerang saja. Dan menurun menjadi 5% pada saat serangan setelah diterapkannya *rules* pada *iptables* tersebut.
6. *Memory usage* juga ikut menurun setelah diterapkannya *rules* tersebut dari 0.9% menjadi 0.3%.
7. Tingkat efektivitas serangan bergantung pada jumlah penyerang pada satu waktu, besarnya sumber daya dari komputer *server*, dan jumlah *threads* serangan yang mana semakin

banyak maka akan semakin membebani *server*.

8. Untuk penelitian selanjutnya diharapkan untuk melakukan simulasi dengan jumlah penyerang yang lebih banyak dan juga sumber daya *server* yang lebih besar agar kedepannya dapat muncul metode-metode lain yang lebih efektif dan meningkatkan keamanan jaringan pada *server*.

## DAFTAR PUSTAKA

- [1] Sularno., & Erdisa. 2016. Analisa dan Implementasi Iptables Dengan Debian Server Sebagai Filtering Firewall Web Server. *Jurnal KomTekInfo*, **3(1)**:106-121.
- [2] Rahman, D. A., Irfan, S., & Suksmadana, M. B. 2019. Implementasi Iptables Untuk Packet Filtering Pada Raspberry Pi. *Dielektrika*, **6(1)**:61-66.
- [3] Purwaningrum, F. A., Purwanto, A., & Darmadi, E. A. 2018. Optimalisasi Jaringan Menggunakan Firewall. *Jurnal IKRA-ITH Informatika*, **2(3)**:17-23.
- [4] Hawari, M. S., & Kurniawan, I. F. 2016. Penerapan Iptables Firewall Pada Linux Dengan Menggunakan Fedora. *Jurnal Manajemen Informatika*, **6(1)**:198-207.
- [5] Suwanto, R., Ruslianto, I., & Diponegoro, M. 2019. Implementasi Intrusion Prevention System (IPS) Menggunakan Snort dan Iptable Pada Monitoring Jaringan Lokal Berbasis Website. *Jurnal Komputer dan Aplikasi*, **7(1)**:97-107.
- [6] Santoso, J. D. 2019. Keamanan Jaringan Nirkabel Menggunakan Wireless Intrusion Detection System. *INFOS Journal*, **1(3)**:44-50.
- [7] Amien, J. A. 2020. Implementasi Keamanan Jaringan Dengan Iptables Sebagai Firewall Menggunakan Metode Port Knocking. *Jurnal Fasilkom*, **10(2)**:159-165.
- [8] Mardiyanto, B., Indriyani, T., & Suartana, I. M. 2016. Analisa dan Implementasi Honeypot Dalam Mendeteksi Serangan Distributed Denial-of-Services (DDoS) Pada Jaringan Wireless. *Integer Journal*, **1(2)**:32-42.
- [9] Sugiyono. 2016. Sistem Keamanan Jaringan Komputer Menggunakan Metode Watchguard Firebox Pada PT Guna Karya Indonesia. *Jurnal CKI On SPOT*, **9(1)**:1-8.
- [10] Atmadji, E. S. J., & Susanto, B. M. 2017. Pemanfaatan Iptables Sebagai Intrusion Detection System (IDS) Pada Linux Server. *Prosiding*.