TUGAS AKHIR – KS141501

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN MICROSOFT WINDOWS SERVER 2016

NETWORK FORENSICS ANALYSIS ON DOS ATTACK AT UBUNTU SERVER 16.04 AT MICROSOFT WINDOWS SERVER 2016

ALOYSIUS TATUS KRISTANTO
NRP 0511440000186

Dosen Pembimbing
Bekti Cahyo Hidayanto, S.Si, M.Kom.

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

TUGAS AKHIR – KS141501

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN MICROSOFT WINDOWS SERVER 2016

ALOYSIUS TATUS KRISTANTO
NRP 0521144000086

Dosen Pembimbing
Bekti Cahyo Hidayanto, S.Si, M.Kom.

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

FINAL PROJECT – KS141501

NETWORK FORENSICS ANALYSIS ON DOS ATTACK AT UBUNTU SERVER 16.04 AND MICROSOFT WINDOWS SERVER 2016

ALOYSIUS TATUS KRISTANTO
NRP 0521144000086

Advisor
Bekti Cahyo Hidayanto, S.Si, M.Kom.

Departement of Information System
Faculty of Information Technology and Communication
Institut Teknologi Sepuluh Nopember
Surabaya 2018

LEMBAR PENGESAHAN

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN MICROSOFT WINDOWS SERVER 2016

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

ALOYSIUS TATUS KRISTANTO
NRP. 05211440000186

Surabaya, Juli 2018
KEPALA
DEPARTEMEN SISTEM INFORMASI

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP 19650310 199102 1 001

LEMBAR PERSETUJUAN

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN WINDOWS SERVER 2016

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

ALOYSIUS TATUS KRISTANTO
NRP. 05211440000186

Disetujui Tim Penguji: Tanggal Ujian : 5 Juli 2018
Periode Wisuda : September 2018

Bekti Cahyo Hidayanto, S.Si, M.Kom **(Pembimbing I)**

Dr. Ir. Aris Tjahyanto M.Kom **(Penguji I)**

Nisfu Asrul Sani S.Kom, M.Sc **(Penguji II)**

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN MICROSOFT WINDOWS SERVER 2016.

Nama Mahasiswa	**: Aloysius Tatus Kristanto**
NRP	**: 05211440000186**
Departemen	**: Sistem Informasi FTIK-ITS**
Dosen Pembimbing	**: Bekti Cahyo Hidayanto, S.Si, M.kom**

ABSTRAK

Network Forensics merupakan salah satu teknik dalam forensika digital yang digunakan untuk mencatat, menangkap dan menganalisa aktivitas jaringan untuk menemukan bukti digital dari suatu serangan menggunakan jaringan komputer sehingga pelaku bisa dapat dituntut sesuai hokum yang berlaku. Contoh serangan menggunakan jaringan computer adalah Denial Of Service (DoS), Spoofing, Phising, Snifing. Bukti digital pada forensik jaringan dapat diketahui dari pola serangan yang dikenali atau penyimpangan dari kondisi tanpa serangan jaringan. Penelitian ini merupakan analisis dari scenario yang bertujuan untuk menginvestigasi dan menganalisa serangan DoS dengan cara mengumpulkan log data dari wireshark, membuat analisa antar scenario pada Ubuntu Server 16.04 dan Microsoft Windows Server 2016. Terdapat 2 skenario diujikan pada sistem operasi Microsoft Windows Server 2016 dan Ubuntu Server 16.04 yaitu menggunakan protokol TCP dan protokol UDP. Terdapat 2 perlakuan terhadap server yaitu tanpa firewall dan firewall telah dikonfigurasi Server di serang menggunakan tools seperti Low Orbits Ion Cannon (Loic). Setelah dilakukan pembuatan skenario maka tahap selanjutnya adalah Pengujian komunikasi antar komputer dengan Ping. Tahap ini bertujuan untuk mengetahui apakah komputer sudah

terhubung dengan jaringan dan sebagai tahap monitoring jaringan pada kondisi tanpa serangan. Setelah dilakukan pengujian komunikasi, tahap selanjutnya merupakan Penyerangan dos menggunakan loic terhadap target. Penyerangan dos ini ditargetkan pada port 80. Tahap selanjutnya adalah analisa bukti digital. Metode yang digunakan adalah anomaly-based detection. Metode ini bertujuan untuk membandingkan kondisi tanpa serangan traffic jaringan yang tanpa serangan dengan traffic jaringan yang telah dilakukan skenario. Tools yang digunakan pada analisa adalah wireshark. Hal yang dilihat adalah log wireshark dari expert information, conversation antar server dan penyerang lalu kinerja server dengan task manager. Hasil penelitian yang dilakukan adalah log wireshark dan conversation pada Ubuntu Server 16.04 dan Microsoft Windows Server 2016 memiliki karakteristik yang sama. Log wireshark dan conversation pada DOS melalui tcp dan UDP memiliki karakteristik berbeda. Jumlah packet dan jumlah bytes pada saat dilakukan eksperimen tidak mengalami perbedaan yang signifikan. Yang membedakan adalah kinerja. Microsoft Windows Server 2016 lebih unggul dalam kinerja saat dilakukan eksperimen dos TCP maupun UDP. Penggunaan Firewall cukup berpengaruh dalam menangani serangan DoS pada kedua Sistem operasi tersebut.

Kata Kunci: : Network Forensics, Wireshark, DOS, Forensika Digital

NETWORK FORENSICS ANALYSIS ON DOS ATTACK AT UBUNTU SERVER 16.04 AND MICROSOFT WINDOWS SERVER 2016

Nama Mahasiswa	**: Aloysius Tatus Kristanto**
NRP	**: 05211440000186**
Departement	**: Sistem Informasi FTIK-ITS**
Advisor	**: Bekti Cahyo Hidayanto, S.Si, M.Kom**

ABSTRACT

Network Forensics is one of the techniques in digital forensics used to record, capture and analyze network activity to find digital evidence of an attack using a computer network so that the offender can be prosecuted according to applicable law. Examples of attacks using computer networks are Denial Of Service (DoS), Spoofing, Phishing, Snifing. Digital evidence on network forensics can be known from recognizable attack patterns or deviations from tanpa serangan network conditions. This study is an analysis of scenarios that aims to investigate and analyze DoS attacks by collecting logs from wireshark, creating a scenario that analyzes on Ubuntu Server 16.04 and Microsoft Windows Server 2016. There are two scenarios tested on Microsoft Windows Server 2016 and Ubuntu Server operating systems 16.04 are using TCP protocol and UDP protocol.There are 2 treatments to the servers, there are without a firewall and a firewall has been configured Server attacked using tools like Low Orbits Ion Cannon (Loic). After creating scenario is done then the next stage is Testing communication between computers with Ping. This stage aims to determine whether the computer is connected to the network and as a network monitoring stage under tanpa serangan conditions. After communication testing, the next step is Dos attack using loic against target. This attack is targeted at port 80. The next stage is the analysis of digital evidence. The method used is anomaly-

based detection. This method aims to compare tanpa serangan conditions of tanpa serangan network traffic with network traffic that has been done scenario. Tools used in the analysis is wireshark. The thing is seen is the log's wireshark of expert information, conversation between server and attacker then server performance with task manager.
The results of the research is log wireshark and conversation on Ubuntu Server 16.04 and Microsoft Windows Server 2016 have the same characteristics. Characteristics of DOS through TCP and UDP are different. The number of packets and the number of bytes at the time of the experiment were not significantly different. What distinguishes is performance. Microsoft Windows Server 2016 is superior in performance when experiments with both TCP and UDP dos. Firewall usage is quite influential in handling DoS attacks on both operating systems.

Kata Kunci: : Network Forensics, Wireshark, DOS, Forensika Digital.

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas segala petunjuk, pertolongan, rahmat dan kekuatan yang diberikan kepada penulis sehingga dapat menyelesaikan buku tugas akhir dengan judul:

ANALISIS FORENSIK JARINGAN TERHADAP SERANGAN DOS PADA SISTEM OPERASI UBUNTU SERVER 16.04 DAN MICROSOFT WINDOWS SERVER 2016

Pada kesempatan ini, penulis ingin menyampaikan terima kasih kepada semua pihak yang telah memberikan dukungan, bimbingan, arahan, bantuan, dan semangat dalam menyelesaikan tugas akhir ini, yaitu:

- Segenap keluarga besar penulis, terutama Ibu Dra. Sus Handini dan Mas Nikodimus Indra Lukmana yang selalu senantiasa mendoakan, memberikan motivasi, dan kebutuhan materiil maupun non-materiil sehingga penulis mampu untuk menyelesaikan pendidikan S1 ini dengan baik.
- Bapak Bekti Cahyo Hidayanto, S.Si, M.Kom. selaku dosen pembimbing dan dosen wali yang yang telah meluangkan waktu untuk membimbing, mengarahkan dan mendukung dalam penyelesaian tugas akhir.
- Bapak Dr. Ir. Aris Tjahyanto, M.Kom., selaku Ketua Jurusan Sistem Informasi ITS merangkap sebagai Dosen Penguji 1, Bapak Nisfu Asrul Sani, S.Kom, M.Sc selaku KaProdi S1 Sistem Informasi ITS merangkap sebagai Dosen Penguji 2 serta seluruh dosen pengajar beserta staf dan karyawan di Departement Sistem Informasi, FTIK ITS Surabaya selama penulis menjalani kuliah. .
- Keluarga besar E-Home yang telah memberikan pelajaran, hitam dan putihnya dunia kampus ITS, dan mengajarkan bahwa kehidupan adalah dunia yang keras dan perlu kita perjuangkan. Canda, tawa, sedih,

lelah, dan amarah selalu ada, namun kalian adalah yang terbaik.

- Teman-teman PSDM Arogan BEM ITS terutama Pelayanan Tuan Putri, PSDM Berarti HMSI, Family of Science two (FAST), Sahabat Kepompong , Omdoers, Keluarga SMAGA-ITS, dan Food Shall Man Though yang telah yang telah memberikan dukungan emotional dan doa selama ini.
- Untuk Teman Teman Osiris, Mas-mbak FOXIS, Basilisk, Solaris, Beltranis atas petuah-petuah dalam menerjang tugas akhir. Serta adik-adik Lannister, Artemis dan 2017 serta masuknya 2018 karena telah memacu saya untuk segera menyelesaikan tugas akhir saya tepat waktu.
- Untuk teman-teman Lab IKTI Surabaya yang telah memberikan waktu untuk berdiskusi, melepas lelah pikiran dan saling memberikan semangat dalam menyelesaikan tugas akhir.
- Teman-teman dari kampus ITS Surabaya, SMAN 3 Madiun, SMPK Santo Yusuf Madiun Jakarta, dan SD 01 Klegen yang telah menginspirasi, khususnya yang sudah lebih dahulu lulus.

Penyusunan tugas akhir ini masih jauh dari sempurna, untuk itu penulis menerima kritik dan saran yang membangun untuk perbaikan di masa mendatang. Semoga tugas akhir ini dapat menjadi salah satu acuan bagi penelitian-penelitian yang serupa dan bermanfaat bagi pembaca.

DAFTAR ISI

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

DAFTAR TABEL

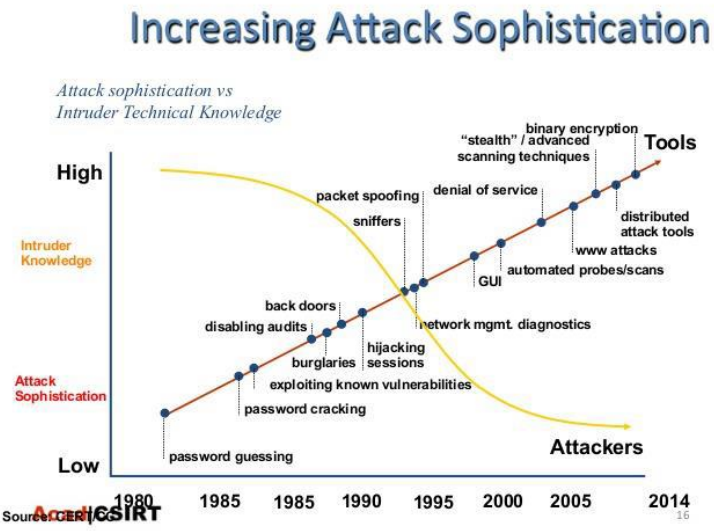
Halaman ini sengaja dikosongkan

BAB I
PENDAHULUAN

Pada bab ini, akan dijelaskan pengenai proses identifikasi masalah yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan, manfaat, dan relevansi tugas akhir. Berdasarkan uraian pada bab ini, diharapkan gambaran umum permasalahan dan pemecahan masalah pada tugas akhir dapat dipahami.

1.1. Latar Belakang

Zaman sekarang teknologi sudah berkembang cepat di negeri ini. Jaringan internet dan teknologi yang dipakai berkomunikasi dengan orang lain di seluruh dunia. Manusia jaman sekarang pun sudah sangat terikat dengan internet dan teknologi. Dari tahun ke tahun pengguna internet meningkat secara pesat[1]. Penggunaan jaringan komputer di instansi-instansi khususnya perusahaan, rumah sakit, perguruan tinggi merupakan suatu kebutuhan yang tidak dapat dielakan lagi. Jaringan komputer dapat diakses banyak orang tanpa terkecuali hacker dan cracker. Dengan berbagai alasan tertentu hacker maupun cracker dapat menyebabkan kerugian kepada pemilik jaringan komputer. Hacker dan cracker menggunakan tools dan metode yang sudah banyak tersebar untuk melakukan penyerangan. Teknologi serangan dan tools pada jaringan komputer berbanding terbalik dengan pengetahuan tentang teknik penyusupan pada jaringan komputer[2]. Hal ini bisa dikarenakan karena semakin modern toolsnya makan semakin otomatis cara penggunaannya. Hal tersebut dapat menjadikan pengetahuan untuk melakukan hal tersebut menjadi rendah. Pada gambar 1 menunjukan Tahun 2000an mulai trend baru yaitu denial of service sampai dengan tahun 2014 yang dimana muncul distributed denial of service.



Gambar 1.1 Teknologi serangan dibanding Pengetahuan[2]

Maka dari itu belakang ini ada istilah forensika digital. Forensika digital adalah proses mengidentifikasi, mengamankan, memulihkan, menganalisa dan mempresentasikan fakta tentang bukti digital yang ditemukan pada komputer atau penyimpanan digital[3]. Di dalam forensika digital , terdapat cabang yang bernama forensika jaringan[4]. Forensik jaringan adalah proses menangkap, mencatat, dan menganalisa aktivitas jaringan guna menemukan bukti digital dari suatu atau kejahatan yang dilakukan atau dijalankan menggunakan jaringan komputer[5]. Forensik jaringan bertujuan untuk menemukan penyerang dan pemulihan tindakan serangan melalui analisis bukti penyusupan[6]. Forensik jaringan berakar dari keamanaan jaringan dan deteksi penyusupan. Tantangan utamanya adalah bagaimana cara mempertahankan bukti, kemudian digunakan dalam proses di pengadilan[7]. Permasalahan yang ada pada jaringan komputer seperti *denial of service, distributed denial of service, spoofing, phising , sniffing.* Penelitian terkait hanya

menggunakan sistem operasi Backtrack[5]. Penelitian yang menggunakan metode *anomaly-based* detection hanya menggunakan sistem operasi windows 7[8].

Dengan melihat permasalahan di atas , penulis berinisiatif untuk melakukan penelitian mengenai analisis forensik jaringan terhadap Denial of Service. Harapannya dengan adanya penelitian ini, dapat memberikan sumbangan pengetahuan bagi akademisi untuk membantu penyelesaian masalah di bidang forensik khususnya pada bidang forensik jaringan.

1.2. **Rumusan Masalah**

Berdasarkan latar belakang di atas, maka rumusan masalah yang akan diteliti pada tugas akhir ini adalah:

1. Bagaimana cara mengimplementasikan Teknik forensik pada jaringan di sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016?
2. Sistem operasi manakah yang paling baik pada server yang sering mengalami serangan *denial of service*?

1.3. Batasan Masalah

Batasan pemasalahan dalam pengerjaan tugas akhir ini adalah :

1. Penelitian menggunakan komputer personal sebagai server yang memiliki sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016.
2. Menggunakan tools Wireshark atau Tshark untuk menangkap paket. Untuk menganalisa menggunakan Wireshark
3. Serangan dilakukan di lingkungan ISnet dengan menyerang server yang telah dibuat skenarionya.
4. Serangan ini menggunakan scenario yaitu serangan melalui TCP dan UDP
5. Serangan menggunakan tools bernama Low Orbit Ion Cannon

6. Perlakuan pada server berupa tanpa firewall, dan menggunakan firewall dan dikonfigurasi.

1.4. Tujuan

Berdasarkan hasil perumusan masalah dan batasan masalah yang telah disebutkan sebelumnya, maka tujuan yang dicapai dari tugas akhir ini:

1. Mengetahui implementasi forensik jaringan pada sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016
2. Mengetahui Sistem operasi mana yang cocok untuk server dalam menangani serangan *DoS*.

1.5. Manfaat

Manfaat yang diharapkan dapat diperoleh dari tugas akhir ini adalah:

1. Memfasilitasi orang tua dan guru dalam mengawasi pergaulan pelajar SMA di Surabaya.
2. Memfasilitasi mahasiswa, khususnya Jurusan Sistem Informasi untuk mempelajari *social media analysis*.
3. Menyediakan data yang dapat digunakan sebagai acuan untuk menentukan tindakan lebih lanjut dalam hal kampanye penggunaan *smartphone* dan *social media* yang bertanggung jawab kepada pelajar di wilayah Surabaya.

1.6. Relevansi

Tugas akhir ini berkaitan dengan mata kuliah forensika digital, keamanan asset informasi dan pengantar sistem operasi. Tugas akhir ini termasuk dalam bidang keilmuan laboratorium infrastuktur dan keamaan teknologi informasi.

BAB II
TINJAUAN PUSTAKA

Bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1. Studi Sebelumnya

Pada subbab ini dijelaskan tentang referensi penelitian yang berkaitan dengan tugas akhir. Pada bagian ini memaparkan acuan penelitian sebelumnya yang digunakan oleh penulis dalam melakukan penelitiannya.

Tabel 2.1 Studi Terkait Penelitian

No	Judul Penelitian	Metode Yang digunakan	Kesimpulan
1.	Investigasi Forensik Jaringan dari Serangan DDOS menggunakan Metode Naïve Bayes. (Y. S. Nugroho 2015)[5]	Penelitiaan menggunakan metode naïve baives yang bertujuan untuk mengklasifikasikan waktu serangan.	Hasil dari penelitian ini adalah dapat mengklasifikasikan kecepatan dan serangan *DDoS* menggunakan tcp atau udp dapat membuat kinerja server lebih berat , perlu adanya hardik berkapasitas besar untuk menyimpan traffic log.
2.	Analisis Forensik Jaringan	Penelitian ini menggunakan metode model	Hasil dari penelitian ini adalah sistem

	Studi Kasus Serangan SQL Injection pada Server Universitas Gajah Mada.(R. U. Putri and J. E. Istiyanto 2012) [4]	proses forensik yang terdiri dari tahap pengkoleksikan, pemeriksaan, analisis dan laporan.	forensik jaringan yang dirancang pada studi kasus ini merupakan sebuah alat untuk menganalisis dari file log. Sistem tersebut dari skrip parsing pcap, skrip port scanning dan skrip untuk merubah log file ke database. Berdasarkan hasil analisis dari log , serangkan sql injection dilakukan menggunakan tools havij dan sql map.
3.	Deteksi Serangan pada Jaringan Komputer dengan Wireshark menggunakan Anomally-Based IDS.(A. B. M. Junaidi Syahputra, Ilham Faisal 2012) [8]	Penelitian ini menggunakan metode Anomally-based ids untuk membandingkan kondisi jaringan sedang di pantau dengan kondisi jaringan yang dianggap tanpa serangan untuk mendeteksi adanya sebuah penyimpangan	Hasil dari penelitian ini adalah semua serangan yang terjadi memiliki karakteristik masing-masing sehingga dapat dikategorikan sebagai serangan karena paket data yang tidak tanpa serangan. Tools wireshark dapat digunakan untuk menerapkan anomaly based ids

.

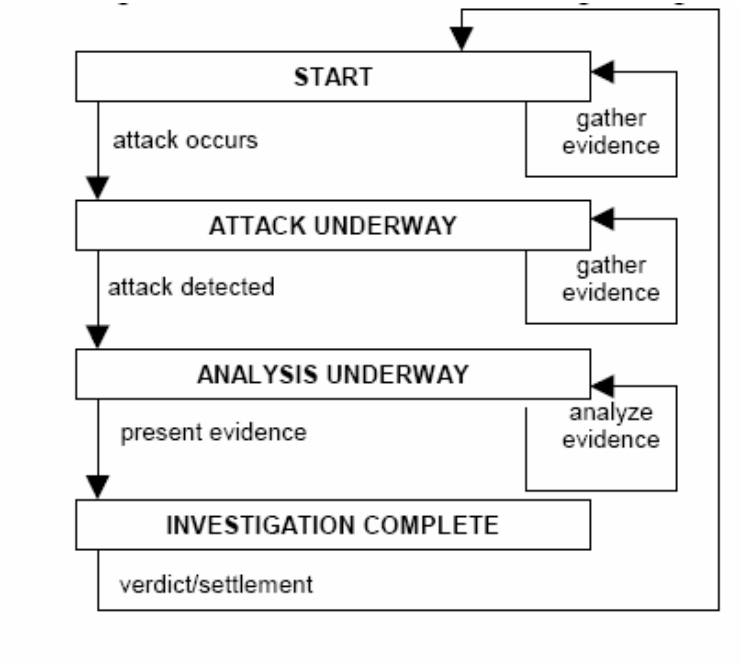
2.2. Dasar Teori

Berisi teori-teori yang mendukung serta berkaitan dengan tugas akhir yang sedang dikerjakan.

2.2.1. Forensika Digital

Forensika digital adalah proses mengidentifikasi, mengamankan, memulihkan, menganalisa dan mempresentasikan fakta tentang bukti digital yang ditemukan pada komputer atau penyimpanan digital[3]. Forensik digital mempunyai cabang yaitu Forensik Jaringan. Forensik jaringan adalah proses menangkap, mencatat, dan menganalisa aktivitas jaringan guna menemukan bukti digital dari suatu atau kejahatan yang dilakukan atau dijalankan menggunakan jaringan komputer [5]. Forensik jaringan biasanya berkutat tentang Ethernet, TCP/IP, Internet, Wireless Forensics. Tahap-tahapan pada forensik jaringan Akuisisi dan pemeriksaan, analisis, dan reporting[9]. Pada forensik jaringan terdapat 2 metode mengenai pengambil data, yaitu *catch-it-as-you-can* dan *stop,look and listen[9]*. *Catch-it-as-you-can* yang mempunyai ciri-ciri adalah semua paket diambil, perlu storage yang besar, menganalisis secara batch, melihat setiap paket. *Stop,look and listen* memiliki ciri-ciri adalah wajib mempunyai processor yang cepat, menganalisis tiap memory, menyimpan yang benar-benar terpakai, dan memfilter secara real time[9].

Di dalam forensik jaringan, terdapat sebuah model forensik yang dari yang berasal dari honeytrap. Honeytrap adalah sebuah tools yang bertujuan untuk melakukan forensik jaringan. Biasanya honeytrap digunakan untuk memancing penyerang untuk menyerang perangkap ini agar host yang asli tidak diserang. Forensik model yang ada pada honeytrap adalah mengumpulkan bukti saat ada indikasi serangan dan saat penyerangan , lalu menganalisa bukti saat penyampaian sebuah bukti[10]. Pada gambar 2 merupakan alur model forensic pada Honeytrap.



Gambar 2.1 Forensik model Honeytrap[10]

2.2.2. Denial of Service

Denial of service merupakan serangan yang bermaksud untuk mematikan mesin atau jaringan yang membuat user tidak dapat mengakses hal tersebut[11]. Serangan ini dijalankan komputer penyerang lebih kuat dari pada target, kuat ini didefinisikan secara *bandwidth*, kecepatan *processor*, dan kapasitas memori. Apabila penyerang lebih lemah dari pada target , maka bisa terjadi sebaliknya[8]. Serangan DoS memiliki beberapa ciri seperti[8]:

1. Logikal: merupakan tipe serangan yang memanfaatkan kelemahan aplikasi, *operating system* atau kesalahan *syntax* pada mesin target. Contohnya smbnuke. 2. *Flooding*: merupakan tipe serangan yang menggunakan protocol TCP, UDP atau ICMP untuk

mengirim packet secara terus menerus ke target dengan request oleh penyerang. Contohnya: Tcp-Flood

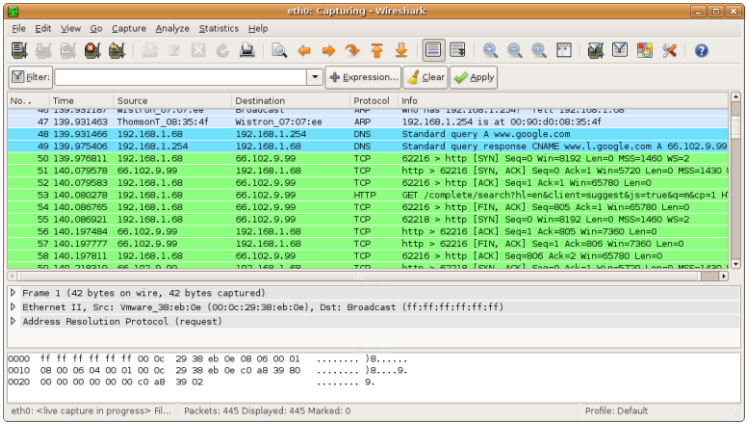
3. UDP *Flood*: merupakan serangan menggunakan UDP untuk melakukan serangan DoS, namun tidak sesederhana dengan TCP. Serangan UDP *flood* dapat dimulai dengan mengirimkan sejumlah besar paket UDP ke *port* acak pada komputer tujuan. Penyerang kerap melakukan *spoofing* pada IP *address* dari paket UDP tersebut sehingga akan terjadi penumpukan paket yang tidak berguna. Untuk menanggulangi UDP *flood*, dapat mematikan semua layanan UDP di semua system pada jaringan atau dengan memfilter semua servis UDP dengan firewall.
4. *Packet Interception*: suatu cara penyerang mendapatkan informasi yang ada didalam paket dengan cara membaca paket disaat paket tersebut dalam perjalanan. Hal ini membutuhkan sebuah aplikasi yang mampu membaca setiap paket yang lewat dalam sebuah jaringan. Cara paling mudah untuk mencegahnya adalah dengan mengenkripsi paket yang akan dikirim.
5. ICMP *Flood:* melakukan ekspolitasi sistem yang bertujuan untuk membuat target hang/down dengan mengirimkan paket icmp/ping dengan ukuran besar sehingga menyebabkan kinerja jaringan menurun.

Berdasarkan tipe, serangan *DoS* terbagi menjadi 2 kategori, yaitu Application layer Attacks dan Network layer attacks[12]. Application layer attack atau dikenal dengan serangan layer ke 7 dapat berupa *DoS* atau *DDoS* yang beroperasi dengan membuat server overload dengan cara mengirimkan request dengan banyak. Network layer attack atau disebut serangan layer ke 3-4 adalah serangan *DDoS* yang memblokir jalannya sebuah jaringan. Perbedaan *DoS* sama *DDoS* terletak pada penggunaan resource. DoS menggunakan 1 koneksi internet untuk melakukan flooding. *DDoS* menggunakan berbagai devices yang tersebar di internet untuk melakukan flooding.

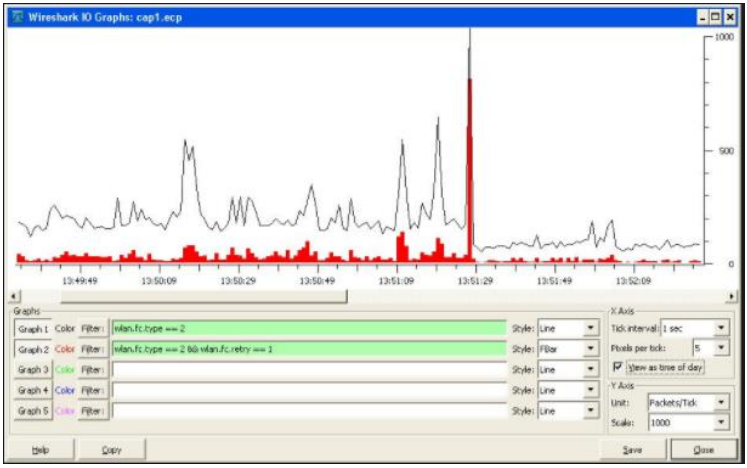
Tools untuk *DoS* biasanya Low orbit ion cannon , sementara *DDoS* biasanya menggunakan *botsnets[12]*.
Parameter untuk mengukur performa server tersebut dalam menangani serangan dos adalah *server CPU processing power* , *memory* dan *storage[13]*. *Proccessing power* merupakan performa atau kecepatan dari sebuah processor dalam menangani suatu aktifitas. Untuk menghitung %*CPU Proccessing power* adalah Penggunaan CPU (GHz) dibagi Daya tampung CPU (GigaHertz). *Memory* merupakan proses menyimpan suatu data pada storage yang bersifat *volatile*. Cara menghitung %Memory adalah Penggunaan Memory (GigaByte) dibagi daya tampung memory (GigaByte). Untuk aktivitas jaringan tidak perlu menggunakan rumus karena sudah terpapar nilai yang dipakai (megabits per detik)[14]. Server dianggap kuat terhadap serangan dos apabila persentase pada CPU, memory , dan trafik tidak lebih dari 70%[15].Hal-hal tersebut dapat dilihat dengan menggunakan Task Manager.

2.2.3. Wireshark

Wireshark merupakan salah satu alat *network analyzer* yang digunakan oleh network administrator untuk menganalisa kinerja jaringan termasuk protocol didalammnya. Wireshark banyak diminati oleh komunitas atau perusahaan karena interfacenya menggunakan *graphical user interface* (GUI). Wireshark mampu menangkap paket-paket data yang melewati jaringan dengan berbagai jenis paket dalam berbagai format protocol. Selain digunakan untuk capture dan analisa , tools ini kerap digunakan untuk memperoleh informasi penting seperti password suatu account dengan menangkap paket-paket yang melewati jaringan[16].*Expert Information* adalah kumpulan anomaly log yang ditangkap oleh wireshark. Tujuan dari *expert information* adalah menemukan kemungkinan masalah jaringan jauh lebih cepat dibandingkan dengan melihat secara manual[17]. *Conversation* merupakan trafik antara 2 endpoints, endpoint ini bisa berupa router, switch ataupun pc. Informasi pada Conversation bisa berupa jumlah bytes ataupun packet dari kedua endpoints



Gambar 2.2 Screenshot Wireshark



Gambar 2.3 Screenshot Wireshark I/O Graph

Ethernet · 5 | IPv4 · 9 | IPv6 | TCP · 5 | UDP · 11

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.61	83.018	5001 k	19	1334	82.999	5000 k	0.163694	16.2346
10.126.10.74	119.81.220.168	453	143 k	249	121 k	204	22 k	0.000000	16.4980
10.126.10.74	10.199.6.166	20	9253	9	1123	11	8130	10.618431	0.0121
1.1.1.2	10.126.10.74	10	1252	5	658	5	594	10.602511	0.0150
10.126.10.74	173.205.14.99	9	1046	5	478	4	568	10.384853	0.2177
10.126.10.1	224.0.0.10	3	222	3	222	0	0	4.703285	9.2348
10.126.10.23	239.255.255.250	3	1161	3	1161	0	0	16.224060	0.2006
10.126.10.74	52.230.3.194	3	364	2	183	1	181	13.758018	0.0762
10.126.10.74	202.46.129.2	2	631	1	83	1	548	10.383140	0.0011

Gambar 2.4 Screenshot Wireshark Conversation

2.2.4. Anomaly-Based Detection

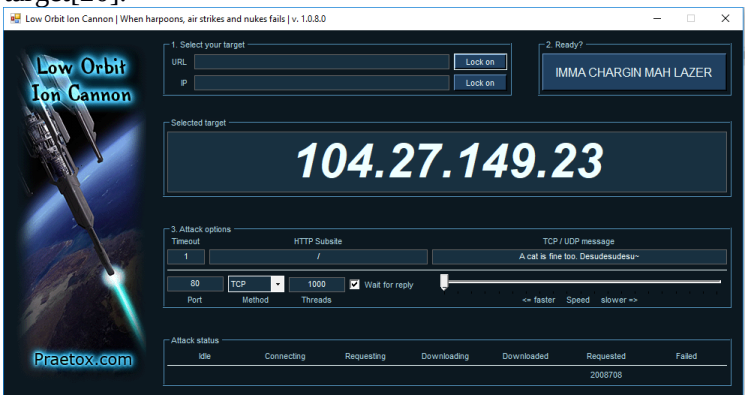
Anomaly-Based Detection merupakan metode yang digunakan untuk membandingkan kegiataan sedang di pantau dengan kegiataan yang dianggap tanpa serangan untuk mendeteksi adanya penyimpangan. Pada metode ini, Intrusion Detection System memiliki profil yang mewakili perilaku yang tanpa serangan dari *user, host,* koneksi jaringan dan aplikasi. Profit tersebut didapat dar hasil pemantauan dalam selang waktu tertentu. Kelebihan dari metode ini adalah efektif dalam mendeteksi ancaman yang belum dikenal. Sedangkan kekurangan dari metode ini adalah dalam beberapa kasus, akan sulit untuk mendapatkan deteksi yang akurat dalam komunikasi yang lebih kompleks. Hal terpenting dalam hal ini adalah menentukan rule. Rule yang dimaksud adalah protocol mana yang dianggap sebagai protocol yang dapat diterima sehingga sistem ini dapat bekerja dengan baik[17].
Serangan Denial of Service (DoS) adalah salah satu contoh jenis serangan yang dapat mengganggu infrastruktur dari jaringan komputer, serangan jenis ini memiliki pola khas, dimana setiap serangannya akan mengirimkan sejumlah paket data secara terus-menerus kepada target serangannya[8].
.

2.2.5. Low Orbit Ion Cannon

Loic atau Low Orbit Ion Cannon merupakan sebuah tools yang digunakan serangan *Denial of Service* yang bersifat

opensource. Pada versi sebelumnnya, loic bernama Ion cannon. Loic awalnya dikembangkan oleh praetox technologies, namun selanjutnya disebar luaskan pada publik. Loic dikategorikan berbahaya karena penggunaannya yang relative mudah. Pengguna tinggal memasukan URL atau IP target lalu memilih metode penyerangan lalu memilih jumlah tread dan timetout dan menunggu hasil maka target tersebut akan down atau tidak bisa diakses. Timeout adalah maksimal waktu yang digunakan untuk meresponse, default pada LOIC adalah 1 (packet tiap detik). Threads adalah jumlah resource yang akan digunakan untuk menyerang (kilobite per detik), hal ini yang akan menjadi variable pnyerangan. Berdasarkan pada penelitian[18] menyimpulkan bahwa pengukuran Dos adalah 0 sampai 1. Berdasarkan referensi ini, apabila loadtime suatu layanan di atas 10 detik maka layanan itu tergolong lamban[19].

Penggunaan loic dapat menyebabkan kerusakan pada server tersebut[8]. Loic dapat menyerang sistem dengan TCP, UDP dan HTTP GET request. Terkadang 1 pengguna loic belum cukup untuk membuat down target, ratusan user pada jaringan yang sama dapat memberikan impact yang signifikan terhadap target[20].



Gambar 2.5 Screenshot LOIC

2.2.6. Microsoft Windows Server 2016

Windows Server 2016 merupakan sistem operasi yang dikembangkan oleh Microsoft pada saat pemgembangan windows 10. Windows Server 2016 *release* pada tanggal 26 September 2016, dan *available* pada tanggal 12 October 2016[21].

2.2.7. Ubuntu Server 16.04

Ubuntu Server 16.04 *release* pada tanggal 21 April 2016 bersamaan dengan Ubuntu 16.04. Ubuntu dibuat oleh Canonical Ltd. Ubuntu merupakan salah satu distro terbaik pada tahun 2018[22].Ubuntu Server 16.04 merupakan versi terbaru yang mendapatkan *long-term support*. Perbedaan Ubuntu Server dengan Ubuntu desktop adalah terletak pada ada atau tidaknya GUI.

2.2.8. Upaya yang dilakukan untuk menangani DoS pada Server

Dalam dunia cybercrime, semua serangan yang ada di dunia digital merupakan senjata. DoS ini dapat diibaratkan dengan *Stun Gun[23]*. Pada dasarnya, dos tidak dapat sepenuhnya di hentikan tetapi dapat dicegah. Berikut 5 cara untuk mencegah serangan dos:

a. Firewall
   Firewal bertujuan untuk memblok ports, protocol dan IP Adresss. Kondisi firewall sekarang hanya dapat memblock port ataupun protocol. Serangan dos yang kompleks pun dapat di cegah dengan Staff IT yang berkompeten. Dalam firewall terdapat outbound dan Inbound Rules. Inbound pada firewall bertujuan untuk melindungi jaringan terhadap jaringan yang masuk dari internet ataupun segment jaringan lain. Inbound dapat memblokir serangan dos. Outbound pada

firewall bertujuan untuk melindungi jaringa terhadap trafik yang keluar[25].

b. Switches dan Routers
   Switch dan router membuat *Access Control List.* ACL ini berfungsi untuk memberikan *permission* untuk mengakses objek-objek tertentu. Hal ini dapat bertujuan untuk membatasi akses sehingga IP-IP tertentu sajalah yang dapat mengakses objek tersebut.
c. Intrusion Prevention System
   Intrusion Prevention System atau IPS bertujuan untuk mengetahui pola serangan dan dapat memblok serangan tersebut.
d. DoS Defense System
   DoS Defense System atau DDS hanya berfokus pada serangan dos. IPS dapat di *counter* dengan packet atau content palsu, tetapi dds dapat mengidentifikasi dan memblok semua serangan yang mencurigakan. Harga Dds memang lebih mahal dari IPS, tetapi merupakan tools paling baik untuk melawan DoS
e. Blackholing dan Sinkholing
   Sinkholing melibatkan pengiriman semua paket yang mencurigakan ke IP address target untuk dijadikan analisa. Blackholing merupakan pengiriman semua paket ke IP Address yang palsu.

Pada Penelitian ini, hal yang akan dilakukan adalah memasang dan mengkonfigurasi firewall bawaan dari Microsoft Windows Server 2016 maupun Ubuntu Server 16.04. Switch dan Router tidak dapat dipraktekan karena bukan wewenang dari penulis karena menggunakan jaringan ISnet, harganya relative tinggi, dan bertentangan dengan skenario karena diposisikan untuk mengcapture kondisi tanpa serangan untuk semua IP. IPS dan DDS juga tidak dapat dilakukan karena harganya sangat mahal. Semua skenario menggunakan sinkholing karena

bertujuan untuk dapat dianalisa pada penelitian ini. Jadi dapat disimpulkan untuk dijadikan uji coba pada penelitian adalah memasang firewall.

2.2.9 Transmission Control Protocol (TCP)

TCP adalah suatu protokol yang berada pada transport layer yang connection-oriented dan reliable. Segmen adalah Unit transmisi di TCP [26]. Segment-segmen pada TCP terdiri dari sebuah header dan segmen data (*payload*). *Payload* adalah data sesungguhnya yang digunakan saat transmisi. Header TCP terdiri dari source port, destination port, sequence number, Acknowledgment Number, Data offset, Reserved, Flags, Windows, Checksum, Urgent Pointer dan option. Pada TCP terdapat istilah TCP Flag. TCP Flag adalah sebuah segment yang mengindikasikan suatu keterangan. Nama-nama flagnya adalah URG, ACK, PSH, RST, SYN, FIN[26]. Berikut merupakan penjelas tiap flag:

- URG(Urgent)
  Mengindikasikan bahwa beberapa bagian dari segmen TCP mengandung data yang sangat penting, dan field Urgent Pointer dalam header TCP harus digunakan untuk menentukan lokasi di mana data penting tersebut berada dalam segmen.
- ACK(Acknowledgment)
  Mengindikasikan field Acknowledgment mengandung segmen selanjutnya yang diharapkan dalam koneksi. Flag ini selalu diset, kecuali pada segmen pertama pada pembuatan sesi koneksi TCP.
- PSH(Push)
  Mengindikasikan bahwa isi dari TCP Receive buffer harus diserahkan kepada protokol lapisan aplikasi. Data dalam receive buffer harus berisi sebuah blok data yang berurutan (kontinyu), dilihat dari ujung paling kiri dari buffer. Dengan kata lain, sebuah segmen yang memiliki flag PSH diset ke nilai 1, tidak bolah ada satu byte pun data yang hilang dari aliran

byte segmen tersebut; data tidak dapat diberikan kepada protokol lapisan aplikasi hingga segmen yang hilang tersebut datang. Normalnya, TCP Receive buffer akan dikosongkan (dengan kata lain, isi dari buffer akan diteruskan kepada protokol lapisan aplikasi) ketika buffer tersebut berisi data yang kontigu atau ketika dalam "proses perawatan". Flag PSH ini dapat mengubah hal seperti itu, dan membuat akan TCP segera mengosongkan TCP Receive buffer. Flag PSH umumnya digunakan dalam protokol lapisan aplikasi yang bersifat interaktif, seperti halnya Telnet, karena setiap penekanan tombol dalam sesi terminal virtual akan dikirimkan dengan sebuah flag PSH diset ke nilai 1. Contoh dari penggunaan lainnya dari flag ini adalah pada segmen terakhir dari berkas yang ditransfer dengan menggunakan protokol FTP. Segmen yang dikirimkan dengan flag PSH aktif tidak harus segera di-acknowledge oleh penerima.

- RST(Reset)
  Mengindikasikan bahwa koneksi yang dibuat akan digagalkan. Untuk sebuah koneksi TCP yang sedang berjalan (aktif), sebuah segmen dengan flag RST diset ke nilai 1 akan dikirimkan sebagai respons terhadap sebuah segmen TCP yang diterima yang ternyata segmen tersebut bukan yang diminta, sehingga koneksi pun menjadi gagal. Pengiriman segmen dengan flag RST diset ke nilai 1 untuk sebuah koneksi aktif akan menutup koneksi secara paksa, sehingga data yang disimpan dalam buffer akan dibuang (dihilangkan). Untuk sebuah koneksi TCP yang sedang dibuat, segmen dengan flag RST aktif akan dikirimkan sebagai respons terhadap request pembuatan koneksi untuk mencegah percobaan pembuatan koneksi.
- SYN(Synchronization)
  Mengindikasikan bahwa segmen TCP yang bersangkutan mengandung Initial Sequence Number

(ISN). Selama proses pembuatan sesi koneksi TCP, TCP akan mengirimkan sebuah segmen dengan flag SYN diset ke nilai 1. Setiap host TCP lainnya akan memberikan jawaban (acknowledgment) dari segmen dengan flag SYN tersebut dengan menganggap bahwa segmen tersebut merupakan sekumpulan byte dari data. Field Acknowledgment Number dari sebuah segmen SYN diatur ke nilai ISN + 1.

- FIN(Finish)
  Menandakan bahwa pengirim segmen TCP telah selesai dalam mengirimkan data dalam sebuah koneksi TCP. Ketika sebuah koneksi TCP akhirnya dihentikan (akibat sudah tidak ada data yang dikirimkan lagi), setiap host TCP akan mengirimkan sebuah segmen TCP dengan flag FIN diset ke nilai 1. Sebuah host TCP tidak akan mengirimkan segmen dengan flag FIN hingga semua data yang dikirimkannya telah diterima dengan baik (menerima paket acknowledgment) oleh penerima. Setiap host akan menganggap sebuah segmen TCP dengan flag FIN sebagai sekumpulan byte dari data. Ketika dua host TCP telah mengirimkan segmen TCP dengan flag FIN dan menerima acknowledgment dari segmen tersebut, maka koneksi TCP pun akan dihentikan.

2.2.10 User Datagram Protocol (UDP)

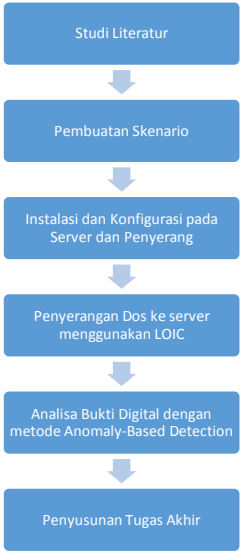
UDP adalah suatu protokol yang berada pada transport layer yang unreliable, connectionless. UDP tidak memiliki segment seperti TCP, tetapi UDP menerapkan UDP Messages.Header UDP adalah source port, destination port, length, checksum. [27].

BAB III
METODOLOGI

Pada bab metode penelitian akan dijelaskan mengenai tahapan – tahapan apa saja yang dilakukan dalam pengerjaan tugas akhir ini beserta deskripsi dan penjelasan tiap tahapan tersebut. Lalu disertakan jadwal pengerjaan tiap tahapanan.

3.1. Tahapan Pelaksanaan Tugas Akhir

Pada sub bab ini akan menjelaskan mengenai metodologi dalam pelaksanaan tugas akhir. Metodologi ini dapat dilihat pada Gambar 3.1



Gambar 3.1 Metodologi Penelitian

3.2. Uraian Metodologi

Pada bagian ini akan dijelaskan secara lebih rinci masing-masing tahapan yang dilakukan untuk penyelesaian tugas akhir ini.

3.2.1 Studi Literatur

Tahap ini merupakan tahap awal yang bertujuan untuk menggali dan menganalisa pengetahuan terkait penelitian yang memiliki topik yang sama, khususnya mengenai metode dan skenario yang akan dicocok untuk studi kasus. Studi literatur yang diambil terkait dengan internet, paper, jurnal, buku dan penelitian yang sudah pernah diteliti sebelumnya sebagai dasar untuk forensik jaringan. Harapannya pada tahap ini dapat membantu penulis dalam membuat kerangka analisa pada tahapan selanjutnya.

3.2.2 Pembuatan Skenario

Pembuatan skenario bertujuan untuk mendapatkan barang bukti digital sebagai langkah awal untuk tahap analisa. Penelitian ini menggunakan skenario yang dibuat semirip mungkin dengan kondisi yang biasa digunakan pada saat melakukan serangan pada suatu jaringan. Skenario yang ini dibagi menjadi 2 model yaitu dos melalui TCP dan dos melalui UDP pada port 80. Untuk membuka port 80, dilakukan instalasi XAMPP pada setiap server. Perlakuan pada server adalah tanpa firewall dan Firewall lalu konfigurasi. Konfigurasi yang dimaksud ini adalah mendeny port yang digunakan oleh penyerang yaitu port 80. Pada pembuatan skenario, terdapat faktor-faktor yang mempengaruh pada server dan penyerang. Faktor-faktor pada server adalah waktu, perlakuan, lingkungan dan ip address. Faktor-faktor pada penyerang adalah waktu, jumlah threads, tipe serangan dan lingkungan.

3.2.3 Instalasi dan Konfigurasi pada Server dan Penyerang

Instalasi dan konfigurasi pada server dan penyerang bertujuan untuk mempersiapkan pelaksanaan eksperimen yang telah dibuatkan skenarionya. Hal-hal yang dilakukan adalah Instalasi Software, Konfigurasi pada server dan Pengujian Komunikasi antar Komputer dengan Ping. Instalasi Software bertujuan untuk mempersiapkan software yang dibutuhkan saat eksperimen. Softwarenya adalah Wireshark, Teamviewer, XAMPP untuk server, LOIC dan Teamviewer untuk penyerang. Teamviewer diinstal pada server bertujuan untuk meremote dari jarak jauh karena lokasi peletakkan server tidak dapat dilakukan konfigurasi. Konfigurasi pada server bertujuan untuk mengatur skenario yang ingin dicapai. Contoh konfigurasi adalah pengaturan firewall. Pengaturan firewall ini didasari oleh upaya yang dilakukan terhadap server pada eksperimen kali ini. Pengujian komunikasi antar computer dengan ping bertujuan untuk mengetahui apakah komputer-komputer terkait telah terhubung dengan baik atau tidak dengan menggunakan perintah ping. Pada Tahap ini juga akan dilakukan monitoring jaringan menggunakan wireshark dengan menangkap paket-paket data yang ada pada jaringan dalam keadaan tanpa serangan.

3.2.4 Penyerangan Dos ke server menggunakan LOIC

Penyerangan Dos ke server menggunakan loic adalah penyerangan menggunakan tools Low Orbit Ion Cannon dengan mengirimkan data terus menerus sehingga target lumpuh. Serangan DoS ini nantinya akan menyerang Port 80 yaitu web service. Port 80 digunakan penyerangan karena port tersebut sering digunakan pada server[24]. Skenario pada penyerangan dos adalah dos melalui TCP dan UDP. Jumlah threads pada setiap serangan DOS adalah 10 threads. Kondisi tanpa serangan dibandingkan dengan saat diserang menggunakan DOS melalui TCP maupun UDP merupakan kondisi ideal apabila ingin membandingkan menggunakan metode anomaly-based detection[8].

3.2.5 Analisa Bukti Digital dengan metode Anomaly-Based Detection

Pada tahap ini akan dilakukan analisa dari bukti digital yang didapatkan pada tahap sebelumnnya. Metode yang dipakai adalah Anomaly-Based Detection[8]. Metode ini digunakan untuk membandingkan kondisi tanpa serangan pada jaringan dengan kondisi saat terjadi penyerangan.Tools untuk melakukan analisa ini yaitu wireshark. Aspek yang akan dianalisa berupa *traffic* jaringan dan performa server. Secara detail dapat dilihat sebagai berikut:

a. Trafik Jaringan

Pada trafik jaringan, jumlah packets dan jumlah bytes yang muncul ditujukan untuk membandingkan kondisi trafik pada kondisi tertentu.

Tabel 3.1 Tabel Perbandingan Trafik Jaringan

	Perlakukan	Jumlah Packets (Packets)			Jumlah Bytes (mb)		
		Tanpa serangan	DOS TCP	DOS UDP	Tanpa Serangan	DOS TCP	DOS UDP
Microsoft Windows Server 2016	Tanpa Firewall						
	Firewall dan Konfigurasi						
Ubuntu Serve	Tanpa Firewall						

r 16.04	Firewall dan Konfigurasi						

b. Kinerja Server

Untuk Kinerja server, kriteria yang dilihat adalah CPU (%), Memory (%) dan aktifitas jaringan (mbps). Pada saat pelaksanaan, dibandingkan dengan skenario kondisi tanpa serangan, penyerangan melalui TCP dan UDP

Tabel 3.2 Tabel Perbandingan Kinerja Server

Perlakuan		Microsoft Windows Server 2016		Ubuntu Server 16.04	
		Tanpa Firewall	Firewall dan Konfigurasi	Tanpa Firewall	Firewall dan Konfigurasi
CPU (%)	Tanpa Serangan				
	DOS TCP				
	DOS UDP				
Memory (%)	Tanpa Serangan				
	DOS TCP				
	DOS UDP				

Aktifitas Jaringan (mbps)	Tanpa Serangan				
	DOS TCP				
	DOS UDP				

3.2.6 Penyusunan Tugas Akhir

Penyusunan tugas akhir merupakan tahap akhir dari seluruh rangkaian penelitian ini. Di dalam laporan tersebut mencakup:

b. Bab I Pendahuluan

Pada bab ini menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan relevansi pengerjaan tugas akhir ini.

c. Bab II Dasar Teori

Pada bab ini menjelaskan mengenai teori-teori yang berhubungan dengan permasalahan pada penelitian tugas akhir ini

d. Bab III Metodologi

Pada bab ini berisii tentang tahapan – tahapan pada penelitian dan apa saja yang harus dilakukan dalam pengerjaan tugas akhir ini.

e. Bab IV Perancangan

Bab ini berisi tentang perancangan yang akan dilakukan dalam penyelesaian permasalahan yang dibahas pada pengerjaan tugas akhir.

f. Bab V Implementasi

Bab ini berisi tentang hal yang akan dilakukan berdasarkan perancangan dalam penyelesaian tugas akhir ini

g. Bab VI Hasil dan Pembahasan

Bab ini berisi mengenai hasil dari penelitian yang telah dilakukan. Hasil tersebut nantinya akan dianalisa juga untuk mendapatkan kesimpulan dan saran pada penyelesaian tugas akhir ini.

h. Bab VII Penutup

Bab ini berisi tentang kesimpulan dan saran yang telah diajukan untuk kelengkapan dan penyempurnaan penelitiaan tugas akhir ini.

Halaman ini sengaja dikosongkan

BAB IV
PERANCANGAN

Bab ini akan menjelaskan langkah awal yang dilakukan penulis sebagai persiapan untuk mendapatkan hasil analisis dalam proses analisis forensik jaringan dan kinerja server pada sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016. Perancangan yang akan dilakukan adalah berkisar pada hal-hal seperti pembuatan skenario, peralatan apa saja yang dibutuhkan pada penelitian hingga analisa bukti digital. Analisa bukti digital yang dimaksudkan ini merupakan pemgambilan data sekaligus melakukan analisa dari bukti digital. Pengambilan data yang dimaksudkan adalah data trafik pada jaringan yang telah di tangkap oleh wireshark. Pada pengambilan data ini nantinya pada saat kondisi tanpa serangan, Ngadat dan down. Setelah data diambil maka dilakukan analisis forensic jaringan secara menyeluruh terhadap data terkait dan kinerja pada tiap server

4.1 Pembuatan Skenario

Sebagai penunjang pada penelitian yang dilakukan, maka di perlukan suatu skenario dalam melakukan eksperimen sampai pengambilan data digital. Skenario ini disesuaikan dengan kondisi lingkungan yang diinginkan. Secara umum, nantinya skenarionya akan dilihat pada kondisi tanpa serangan, ngadat dan down. Skenario dibagi manjadi dua yaitu skenario untuk server dan penyerang.

4.1.1 Perancangan Skenario server

Skenario untuk server perlu dituliskan secara detil. Karena pada setiap detilnya sangat mempengaruhi hasil dan analisa pada barang bukti digital. Hal yang di lakukan pada server adalah menangkap data pada trafik jaringan. Skenario untuk server yang akan dibuat adalah dengan memperhatikan faktor seperti:

- Waktu
  Skenario perlu dibuat menyesuaikan dengan waktu antara pelaku dan juga peneliti. Waktu yang cocok untuk melakukan mendefinisikan kondisi tanpa serangan adalah pada saat jam kerja. Jam kerja yang dimaksud adalah pukul 08.00 – 16.00. Karena pada saat itu orang-orang menggunakan jaringan ISnet untuk melakukan kegiatan. .
- Perlakuan pada server
  Perlakuan pada server bertujuan untuk mengetahui kinerja server yang bisa dilakukan. Perlakuan yang dilakukan adalah server tidak diinstal firewall, diinstal firewall, dan dikonfigurasi.
- Lingkungan
  Server diletakkan pada Network Operation Center di lantai 2 Departement Sistem Informasi dan tersambung pada Switch Isnet.
- IP Address
  Ip address pada server dibuat static agar dapat dimasukan kedalam jaringan ISNET. IP address dari 2 server adalah 10.126.10.73 dan 10.126.10.74

4.1.2 Perancangan Skenario Penyerang

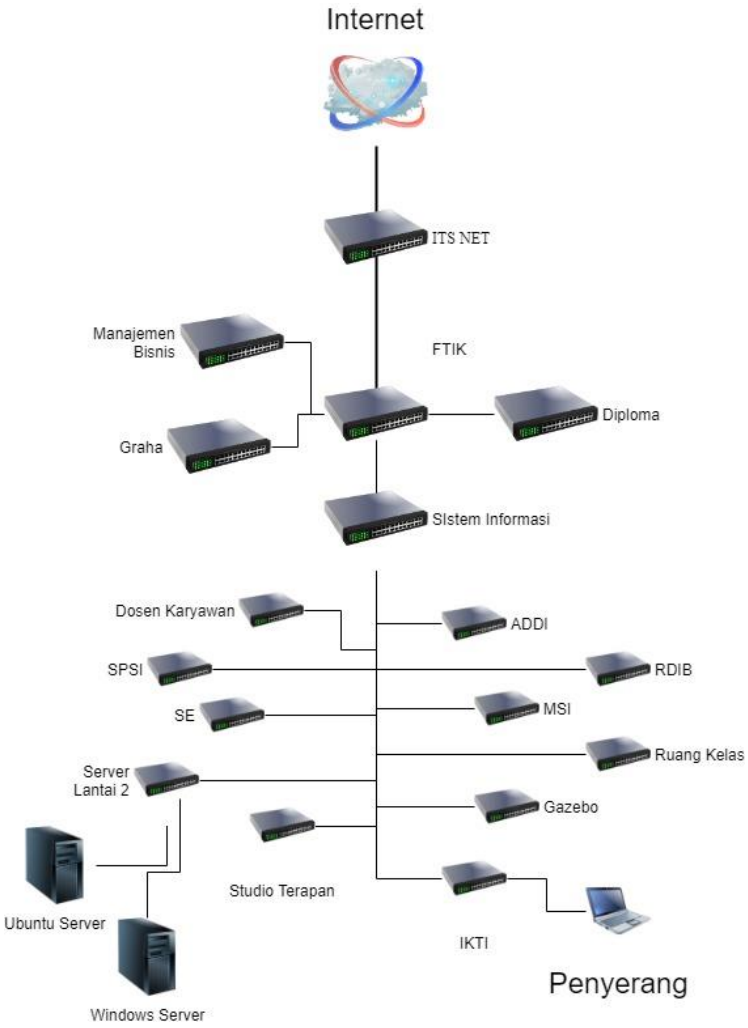
Mirip dengan perancangan skeneario pada server, penyerang juga perlu memerhatikan beberapa faktor untuk mengambil barang bukti digital pada jaringan. Perbedaan pada penyerang adalah faktor-faktor ini tidak serumit dan sebanyak yang ada pada skenario server.

- Waktu
  maksud dari waktu ini adalah waktu yang cocok untuk melakukan simulasi penyerang ke server. Dikarenakan penggunaan pada jaringan ISnet pada jam kerja terlalu pada, maka penyerangan dilakukan setelah jam kerja atau sekitar jam 4 sore sampai jam 7 pagi.

- Jumlah Threads
  Jumlah thread bertujuan untuk membatasi seramgan dos yang diberikan pada server. Jumlah threads ini dibuat tetap yaitu 10 threads.
- Tipe serangan
  Tipe serangan ini bertujuan untuk membedakan jenis serangan yang akan diberikan pada server. Tipe serangan nantinya ada 2 yaitu melalui TCP dan UDP
- Lingkungan
  Penyerang melakukan serangan dos akan disimulasikan pada jaringan IKTI di lantai 2 Departement Sistem Informasi.

4.1.3 Topologi Jaringan

Karena menggunakan jaringan ISnet maka topologi jaringannya pun juga menggunakan topologi pada ISnet. Server dan penyerang diletakan di switch yang berbeda. Server berada pada switch lantai 2 DSI, sedangkan penyerang berada pada switch lab IKTI.



Gambar 4.1 Topologi Jaringan ISNet

4.2 Kebutuhan peralatan

Ada beberapa peralatan dalam melakukan skenario forensik jaringan. Jika dalam kasus forensik jaringan sesungguhnya

membutuhkan peralatan yang rumit, maka untuk penelitian kali ini tidak terlalu banyak. Kebutuhan ini nantinya dibagi menjadi 2 bagian yaitu, kebutuhan untuk software dan kebutuhan untuk hardware. Dan ditiap kebutuhan software maupun hardware dibedakan lagi untuk server dan penyerang.

4.2.1 Kebutuhan untuk Software

Dalam melakukan skenario forensik jaringan perlu memerlukan software agar skenario yang diinginkan berjalan dengan baik. Software yang diperlukan pun dibagi 2 yaitu untuk server dan untuk penyerang. Berikut software yang dibutuhkan untuk menjalankan skenario:

- Server
  Pada Server nantinya akan menggunakan sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016. Setiap sistem operasi akan dijalan pada server yang berbeda. Teruntuk server yang memakai sistem operasi ubuntu, akan diinstal juga ubuntu desktop agar dapat menjalankan wireshark karena wireshark memerlukan GUI dan tidak sekedar CLI. Di setiap servernya akan diinstal wireshark. Nantinya di setiap server akan diinstal firewall karena di dalam skenario forensik jaringan menyebutkan bahwa firewall merupakan salah satu bentuk threatment pada server. Ditiap server akan diinstal XAMPP sebagai port sasaran (port 80) yang akan diujikan serangan dos. Ditiap server akan diinstal teamviewer yang bertujuan untuk melakukan remote desktop.
  - Penyerang
    Kebutuhan software penyerang tidak sebanyak kebutuhan software pada server. Penyerang nantinya akan menggunakan sistem operasi windows 8 bit 64bit. Applikasi yang akan dipakai adalah Low orbit Ion

Cannon dan teamviewer. Low Orbit Ion Cannon bertujuan untuk melakukan serangan dos. Teamviewer bertujuan untuk melakukan remote desktop.

4.2.2 Kebutuhan untuk Hardware

Dalam melakukan penelitian ini diperlukan beberapa hardware dalam skenario forensik jaringan. Beberapa hardware yang akan digunakan dengan spesifikasi sebagai berikut:

- Proccessor: Intel Core i5 & Intel core 2 duo
- Memory: 8gb & 4gb
- Storage: 500gb
- VGA: On board

4.3 Analisa bukti digital

Analisa bukti digital merupakan pengambilan sekaligus menganalisi bukti digital tersebut. Bukti digital ini nantinya akan dijadikan pembanding di tiap skenarionya. Pembanding ini dilihat dari segi forensik dan kinerja server. Pengambilan data diambil melalui tools wireshark saat skenario selesai dilakukan. Untuk Analisa bukti digital menggunakan metode anomaly-based detection. Metode ini bertujuan untuk membandingkan kondisi tanpa serangan dengan kondisi yang dibuat skenario. Aspek yang dilihat adalah aspek forensik dan aspek kinerja server. Untuk aspek forensik dilakukan menggunakaan wireshark, sedangkan untuk aspek kinerja server dilihat dari task manager dan alat tambahan yang akan disesuaikan dengan sistem operasi terkait.

BAB V
IMPLEMENTASI

Bab ini menjelaskan mengenai pelaksanaan dari penelitian yang dilakukan dengan metode dan perangkat yang telah dituliskan pada bagian sebelumnya. Tahap implementasi dimulai dari proses instalasi software, pengujian ping antar komputer, melakukan esperimen serangan dos ke server, pengambilan data digital di server, aplikasi pendukung untuk melaksanaan eksperimen dan hambantan pada saat eksperimen. Proses ini akan menghasilkan *input* terhadap hasil dan pembahasan pada bab berikutnya.

5.1. Proses instalasi software

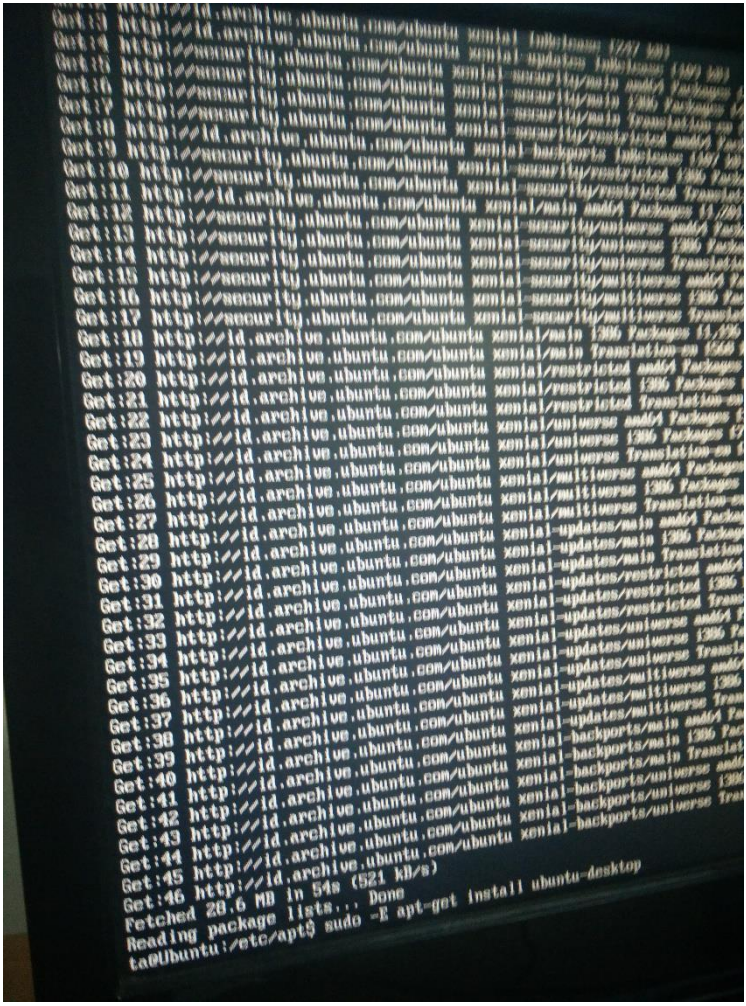
Dikarenakan server harus *fresh install*, maka diperlukan untuk menginstal ulang sistem operasi dan software agar tidak terjadi ketimpangan antar server untuk di jadikan server nantinya. Proses instalasi software ditiap sistem operasi berbeda perlakuannya karena pada Ubuntu Server tidak ada GUI sehingga harus dilakukan instalasi UI ubuntu.

5.1.1 Ubuntu Server 16.04

Hal pertama yang dilakukan adalah menginstall sistem operasi. Instalasi sistem operasi ini menggunakan USB bootable. Aplikasi yang di pakai untuk menjadikan usb bootable adalah rufus. Setelah selesai melakukan instalasi pada sistem operasi, selanjutnya adalah melakukan instalasi software-software yang diperlukan pada penelitian ini.

a. Instalasi Ubuntu Desktop

Pada dasarnya Ubuntu server bentuknya masih *command-line interface* dan wireshark memerlukan GUI untuk menjalankannya. Maka dari itu Ubuntu Server ini harus diinstal Ubuntu-desktop untuk menjalankan aplikasi-aplikasi GUI. Cara menginstal Ubuntu-desktop seperti foto digambar ini.



Gambar 5.1 Gambar Ubuntu Server CLI

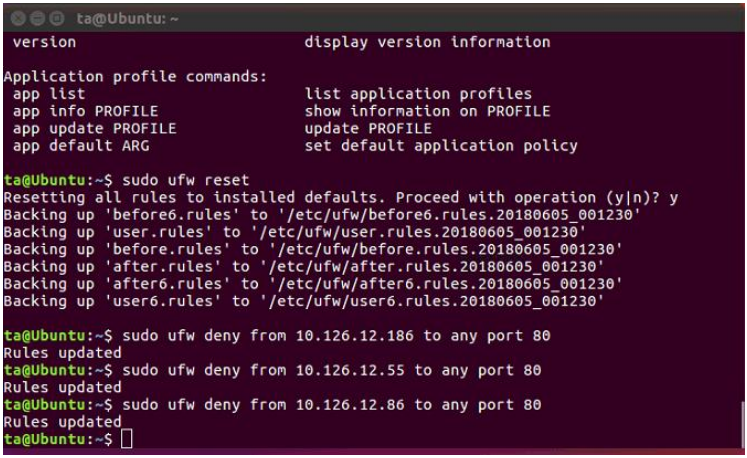
Sepintas Ubuntu server ini mirip dengan Ubuntu seperti biasanya, tetapi sangat berbeda mengenai package-package yang sudah terinstal di sistem operasi ini. Sebut saja package seperti Open SSH dan LAMPP yang sudah terinstal dari Ubuntu server.

b. Instalasi Wireshark pada Ubuntu Server

Setelah menginstal GUI terhadap Ubuntu server, hal yang dilakukan setelah itu adalah melakukan install wireshark. Instalasi wireshark pada Ubuntu server sama saja seperti menginstall aplikasi-aplikasi linux pada umumnnya. Dalam penelitian ini penulis menginstall wireshark melalui Ubuntu software center.

c. Instalasi dan konfigurasi Ubuntu firewall

Ketika instalasi Ubuntu-desktop, secara otomatis Ubuntu firewall akan ikut terinstall juga. Walaupun sudah terinstall, tetapi Ubuntu firewall ini tidak secara langsung aktif. Maka dari itu Ubuntu firewall perlu di-*enable* kan dahulu. Untuk mengaktifkan Ubuntu firewall, gunakan commands "sudo ufw enable". Skenario yang dipakai pada penelitian adalah memblokir IP penyerang pada port 80 khususnya TCP dan UDP. Maka dengan commands "sudo ufw deny from IP to any port 80" sudah cukup mewakili kebutuhan hal tersebut



Gambar 5.2 Screnshot konfigurasi UFW

5.1.2 Microsoft Windows Server 2016

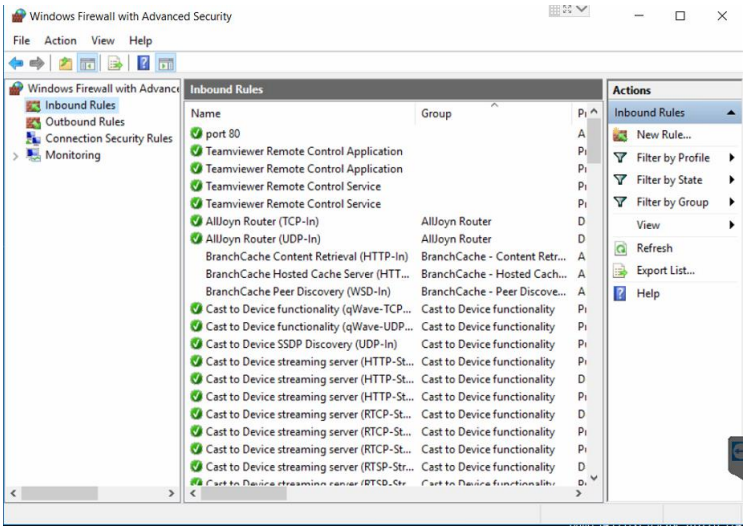
Hal pertama yang harus dilakukan adalah instalasi sistem operasi. Berbeda dengan ubuntu server yang harus melakukan instalasi GUI, Windows Server cukup melakukan instalasi sistem operasi seperti biasa menggunakan usb bootable. Hal ini juga menggunakan rufus untuk menjadikan usb menjadi bootable. Proses membuat usb bootablenya pun sama seperti yang ada di ubuntu server. Setelah selesai install sistem operasi, hal selanjutnya adalah menginstal aplikasi-aplikasi yang dibutuhkan pada penelitian ini.

a. Instalasi wireshark pada Windows Server

Instalasi wireshark pada windows server berbeda dengan apa yang ada di Ubuntu server yang memiliki ubuntu software center. Instalasi wireshark pada umumnya seperti aplikasi-aplikasi windows biasanya.

b. Konfigurasi firewall pada windows server

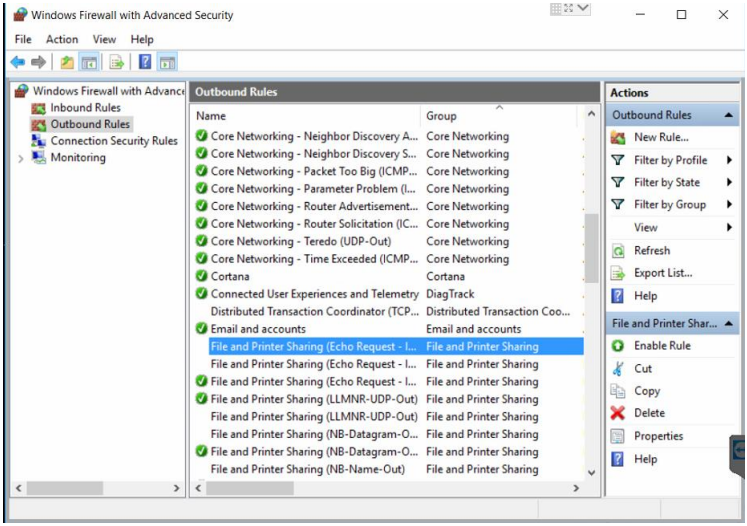
Pada dasarnya windows server sudah menjalankan windows firewall secara aktif. Firewall pada windows server pun sudah mempunyai rule-rule default contohnya seperti memblokir ICMP. Port 80 pun pada firewall windows server secara default sudah deny. Pada kasus ini, penulis menambah rules pada firewall untuk memblokir IP penyerang pada protokol TCP dan UDP.



Gambar 5.3 Screenshot Inbound Rules pada Windows Firewall

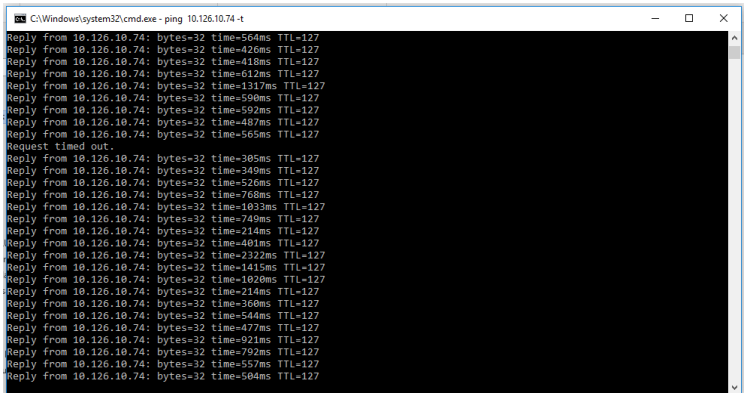
5.2 Pengujian Ping antar Komputer

Pengujian ping bertujuan untuk mengetahui apakah server dan penyerang sudah terhubung baik atau tidak dengan jaringan. Sebelum melakukan ping, server harus dilakukan konfigurasi agar dapat diletakkan pada ruangan server ISNet. Untuk Windows Server perlu melakukan konfigurasi pada windows firewall karena windows server secara otomatis memblokir ICMP lebih tepatnya pada Ip4.



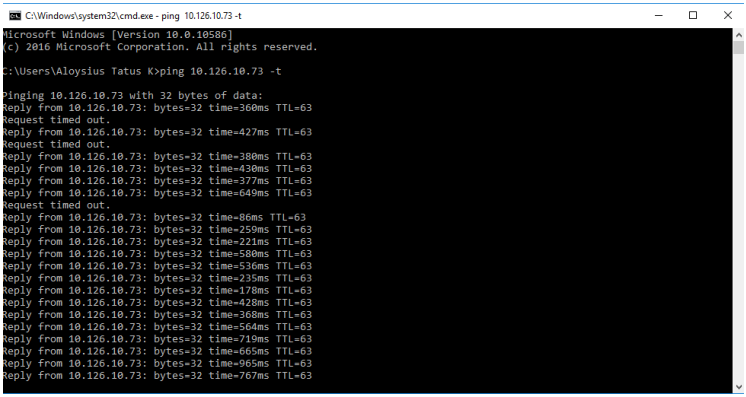
Gambar 5.4 Screenshot salah satu rules pada Outbound Rules

Setelah dilakukan konfigurasi, maka dilakukan ping dari penyerang ke server. Berikut *screenshot* ping ke Windows Server



Gambar 5.5 Ping menuju Ubuntu Server

Pada Ubuntu server tidak memerlukan konfigurasi secara khusus seperti pada windows server. Berikut *screenshot* pengujian ping pada Ubuntu server.



Gambar 5.6 Ping menuju Windows Server

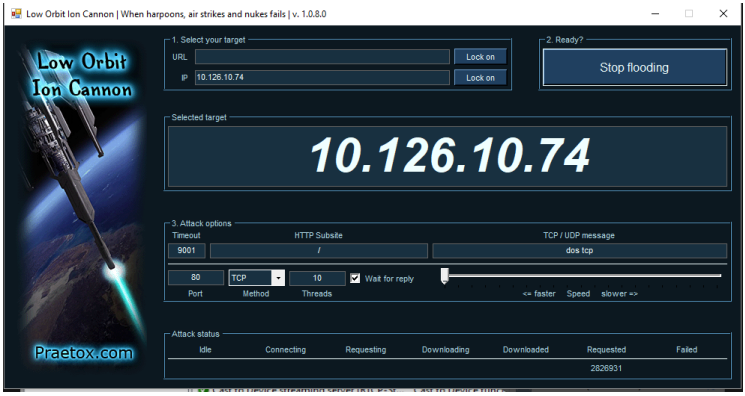
Pada dua gambar di atas sudah menunjukan bahwa ada terjadinya komunikasi antara target dan penyerang.

5.3 Melakukan eksperimen serangan Dos ke server

Setelah melakukan uji komunikasi antar komputer, dilakukan eksperimen serangan dos ke tiap server. Eksperimen serangan dos ini menggunakan aplikasi bernama low ion orbit cannon.

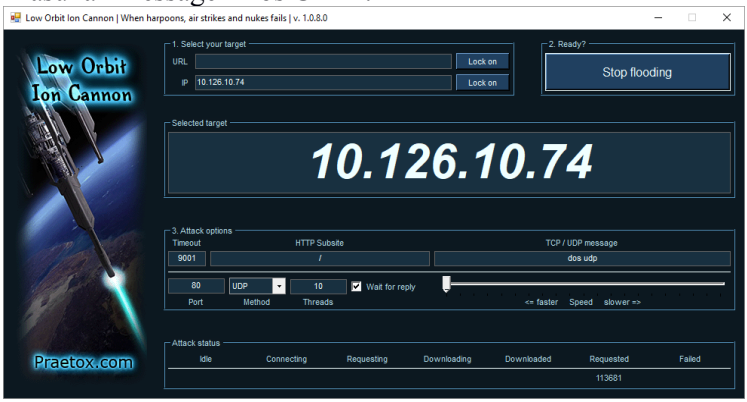
5.3.1 Windows Server

Berikut merupakan eksperimen dos ke server melalui TCP. Masukan 10.126.10.74 sebagai tujuan lalu Lock, selanjutnya masukan threads sejumlah 10 dan pilih TCP sebagai method. Masukan message “Dos TCP”.



Gambar 5.7 Screenshot LOIC protokol TCP target Windows

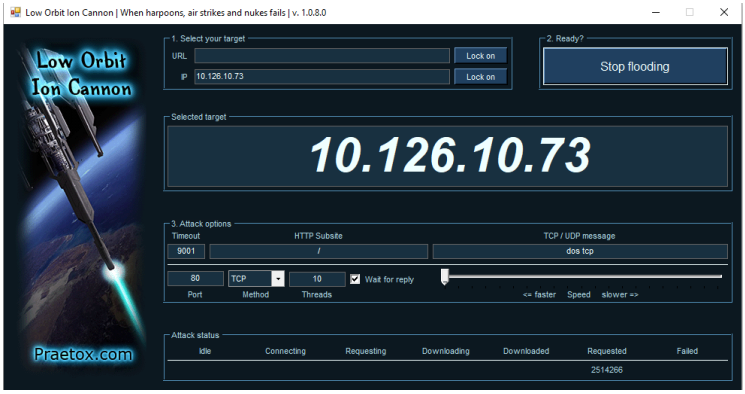
Berikut merupakan eksperimen dos ke server melalui UDP. Masukan 10.126.10.74 sebagai tujuan lalu Lock, selanjutnya masukan threads sejumlah 10 dan pilih UDP sebagai method. Masukan message “Dos UDP”.



Gambar 5.8 Screenshot LOIC protokol UDP target Windows

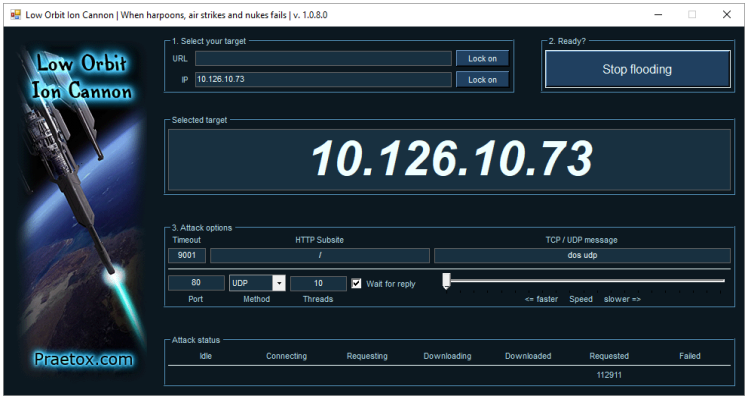
5.3.2 Ubuntu Server

Berikut merupakan eksperimen dos ke server melalui TCP. Masukan 10.126.10.73 sebagai tujuan lalu Lock, selanjutnya masukan threads sejumlah 10 dan pilih TCP sebagai method. Masukan message “Dos TCP”.



Gambar 5.9 Scrnshot LOIC protokol TCP target Ubuntu

Berikut merupakan eksperimen dos ke server melalui UDP. Masukan 10.126.10.73 sebagai tujuan lalu Lock, selanjutnya masukan threads sejumlah 10 dan pilih TCP sebagai method. Masukan message “Dos UDP”.



Gambar 5.10 Screenshot LOIC protokol UDP target Ubuntu

5.4 Pengambilan Data digital di Server

Data digital yang dimaksud adalah hasil dari capture dari wireshark yang telah menjalankan beberapa scenario yang telah di tentukan. File asli capture yang asli masih berada pada server. Untuk menyimpan log yang ada pada wireshark yaitu Files → Saves, lalu masukan nama file.

ubuntu firewall dos tcp	05/06/2018 00.18	Wireshark capture...	752 KB
ubuntu firewall dos udp	05/06/2018 00.22	Wireshark capture...	5.211 KB
ubuntu firewall normal	05/06/2018 00.17	Wireshark capture...	1.331 KB
ubuntu kondisi normal	31/05/2018 21.02	Wireshark capture...	35.552 KB
ubuntu normal dos tcp	31/05/2018 23.48	Wireshark capture...	5.225 KB
ubuntu normal dos udp	31/05/2018 23.49	Wireshark capture...	7.553 KB
windows block dos tcp and udp	02/06/2018 06.40	Wireshark capture...	2.021 KB
windows block dos udp	02/06/2018 06.47	Wireshark capture...	11.756 KB
windows block normal	02/06/2018 06.34	Wireshark capture...	216 KB
windows Kondisi Normal	29/05/2018 04.33	Wireshark capture...	36.802 KB
windows normal dos tcp	29/05/2018 07.31	Wireshark capture...	6.542 KB
windows normal dos udp	29/05/2018 07.34	Wireshark capture...	7.653 KB

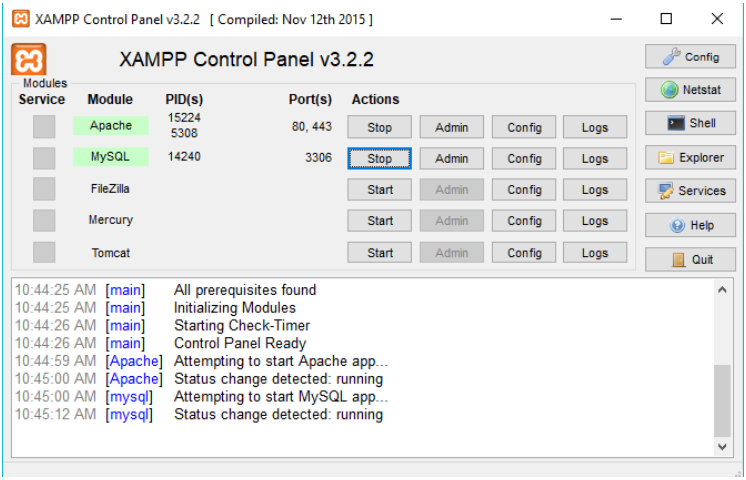
Gambar 5.11 Screenshot File Wireshark

5.5 Aplikasi pendukung

Aplikasi pendukung ini bertujuan agar pelaksanaan penelitian lebih mudah dan sesuai dengan tujuan yang ingin penulis capai. Terdapat 2 aplikasi pendukung pada penelitian ini yaitu

5.5.1 XAMPP

XAMPP bertujuan untuk menjadikan port 80 di Server menjadi Open. XAMPP dapat diinstal pada windows Server dan Ubuntu Server. XAMPP merupakan tools yang bisa digunakakan sebagai aplikasi pendukung pada penelitian ini.



Gambar 5.12 Tampilan XAMPP

Teruntuk Ubuntu server tidak perlu dilakukan install lagi dikarenakan pada saat install ulang sistem operasi dapat memilih untuk mengaktifkan Xampp atau tidak. Pada penelitian ini menggunakan Xampp melalui install ulang pada sistem operasi. Perbedaan yang mendasar XAMPP yang diinstal melalui sistem operasi dan secara manual adalah

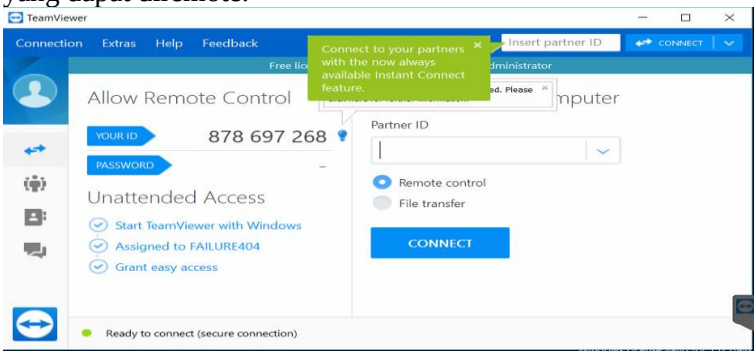
XAMPP yang diinstal melalui sistem operasi secara otomatis menjalankan XAMPP tanpa harus diaktifkan dahulu.

5.5.2 Teamviewer

Teamviewer bertujuan untuk melakukan remote desktop pada server. Keunggulan teamviewer dibanding dengan *ssh* adalah teamviewer mendukung GUI.

a. Teamviewer pada Windows Server

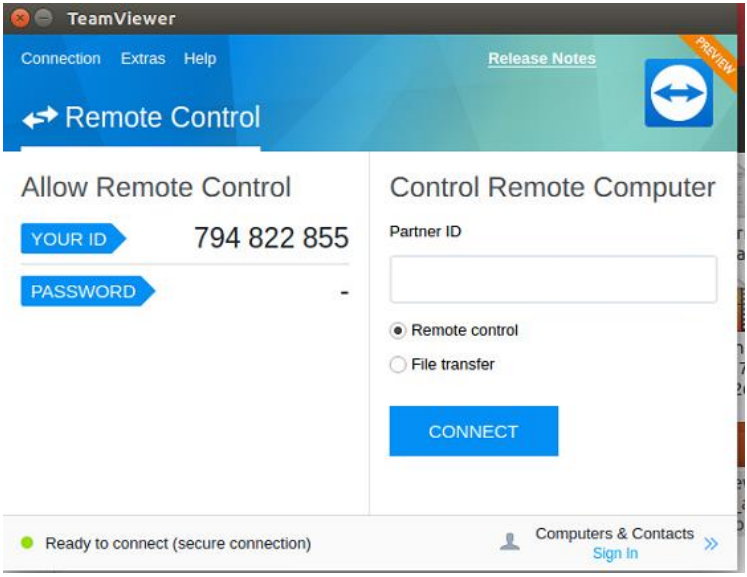
Instalasi teamviewer pada Windows Server relatif mudah karena hanya 2 steps saja. Pada kasus ini, penulis melakukan eksperimen pada jaringan LAN. Dengan kata lain, teamviewer harus diatur agar dapat terkoneksi ke LAN (tanpa jaringan internet). Hal yang dilakukan khususnya di Windows Server adalah melakukan login dan mendaftarkan server ke device yang dapat diremote.



Gambar 5.13 Screenshot tampilan Teamviewer

b. Teamviewer pada Ubuntu Server

Instalasi teamviewer pada Ubuntu server tidak berbeda jauh dengan windows server. Teamviewer pada ubuntu server memiliki konfigurasi yang sedikit berbeda dengan windows server. Pada Ubuntu server tidak perlu login untuk dapat meremote melainkan melakukan whitelist terhadap ID.



Gambar 5.14 Tampilan Teamviewer pada Ubuntu

5.6 Hambatan dan Rintangan

Pada saat melakukan eksperimen serangan dos ke objek, wireshark mengalami not responding. Sehingga perlu dilakukan eksperimen ulang sampai data dapat diambil. Halangan selanjutnya adalah pada saat proses intalasi software terkendala mengenai hardware yang ada digunakan server. Hardware yang digunakan sudah berumur dan pada saat itu harus diganti agar server dapat berjalan. Selain itu, hardware khususnya motherboard pada windows server tidak ada setinggan yang mengatur apabila ada ada arus listrik server bisa menyala otomatis. Sehingga apabila terjadi pemadaman maka harus menghidupkan server secara manual. Rintangan utama pada penelitian ini adalah eksperimen dos harus dilakukan pada jam malam karena apabila dilakukan pada jam kerja maka traffic pada ISnet akan penuh sehingga mengganggu proses bisnis pada isnet. Selain itu, proses capturing dan eksperimen serangan dos tidak boleh terpaut pada waktu yang lama. Ini bertujuan agar data yang diambil

pada saat eksperimen dos realiable, yang dimaksud reliable ini adalah tidak ada perbedaan IP Address penyerang.

BAB VI
HASIL DAN PEMBAHASAN

Pada bab ini akan menjelaskan hasil yang sudah didapatkan dari eksperimen dos yang telah dilakukan dan analisa terhadap permasalahan yang ingin dijawab dalam penelitian ini.
Terdapat tiga macam analisa yang akan dibahas pada bagian ini, yaitu analisa *log*, *statistics* dan kinerja server beserta dengan pembahasannya.

6.1 Hasil Eksperimen Dos ke Server

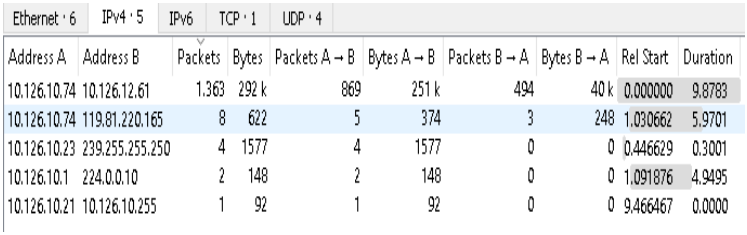
Bagian ini akan menjelaskan mengenai hasil dari eksperimen dos ke server. Terdapat 2 eksperimen dengan perlakuan yang berbeda. Berikut merupakan hasil dari eksperimen yang sudah dilakukan

6.1.1 Hasil Eksperimen Tanpa Firewall

Eksperimen pertama merupakan eksperimen dos ke server tanpa menggunakan firewall. Hasil dibedakan antara windows dan Ubuntu.

a. Windows

Berikut merupakan hasil eksperimen tanpa firewall pada sistem operasi Microsoft Windows Server 2016.

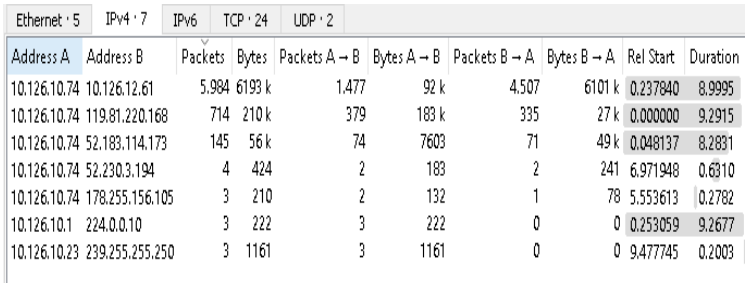


Ethernet · 6 | IPv4 · 5 | IPv6 | TCP · 1 | UDP · 4

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.61	1.363	292 k	869	251 k	494	40 k	0.000000	9.8783
10.126.10.74	119.81.220.165	8	622	5	374	3	248	1.030662	5.9701
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	0.446629	0.3001
10.126.10.1	224.0.0.10	2	148	2	148	0	0	1.091876	4.9495
10.126.10.21	10.126.10.255	1	92	1	92	0	0	9.466467	0.0000

Gambar 6.1 Screenshot *conversation K*ondisi Tanpa Serangan dan Tanpa Firewall pada Windows Server

Untuk melihat *conversation*, buka wireshark→statistics→conversation.
Gambar di atas adalah *conversation* antar IP yang telah ditangkap oleh wireshark. Terlihat banyak pertukaran data terhadap beberapa IP. IP yang terlihat tidak selalu IP terhadap suatu PC, melainkan router atau switch. Gambar di atas adalah kondisi ketika tanpa serangan dan tanpa firewall dimana jumlah packets dan bytesnya relatif kecil yaitu 1363 packet dan 292kb.



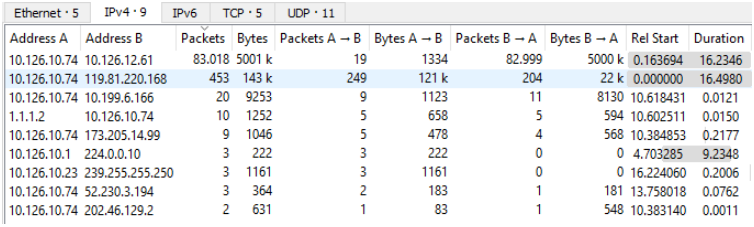
Ethernet · 5 | IPv4 · 7 | IPv6 | TCP · 24 | UDP · 2

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.61	5.984	6193 k	1.477	92 k	4.507	6101 k	0.237840	8.9995
10.126.10.74	119.81.220.168	714	210 k	379	183 k	335	27 k	0.000000	9.2915
10.126.10.74	52.183.114.173	145	56 k	74	7603	71	49 k	0.048137	8.2831
10.126.10.74	52.230.3.194	4	424	2	183	2	241	6.971948	0.6310
10.126.10.74	178.255.156.105	3	210	2	132	1	78	5.553613	0.2782
10.126.10.1	224.0.0.10	3	222	3	222	0	0	0.253059	9.2677
10.126.10.23	239.255.255.250	3	1161	3	1161	0	0	9.477745	0.2003

Gambar 6.2 Screenshot *conversation* DoS melalui TCP dan Tanpa Firewall pada Windows Server

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui TCP. Terlihat IP penyerang melakukan pertukaran data yang banyak. Apabila diurutkan berdasarkan packet dan byte, IP penyerang menduduki posisi

paling atas. IP penyerang pun lebih banyak mengirimkan daripada menerima server. Terdapat 5984 paket dan 6193 kb pada saat komunikasi antara penyerang dan server. Berbeda jauh dengan kondisi tanpa serangan yang hanya 1363 packet dan 292 kb. Hal ini menandakan ada aktifitas yang mencurigakan.

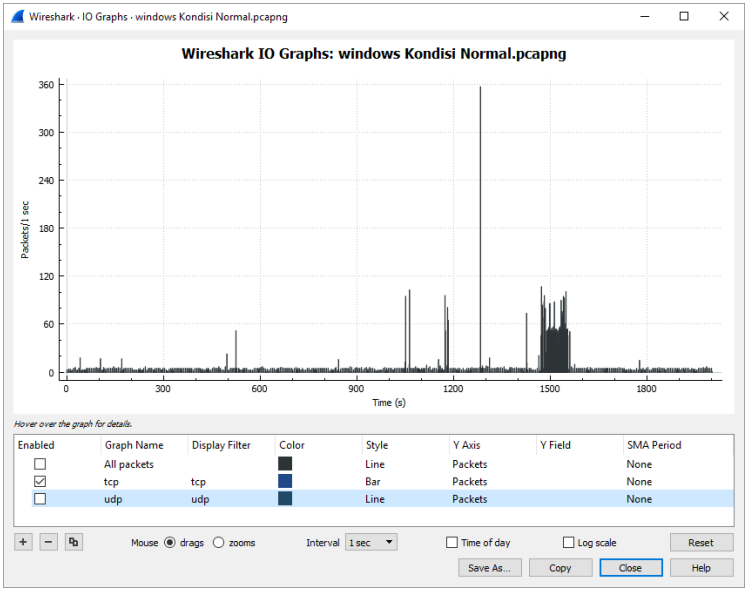


Ethernet · 5 | IPv4 · 9 | IPv6 | TCP · 5 | UDP · 11

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.61	83.018	5001 k	19	1334	82.999	5000 k	0.163694	16.2346
10.126.10.74	119.81.220.168	453	143 k	249	121 k	204	22 k	0.000000	16.4980
10.126.10.74	10.199.6.166	20	9253	9	1123	11	8130	10.618431	0.0121
1.1.1.2	10.126.10.74	10	1252	5	658	5	594	10.602511	0.0150
10.126.10.74	173.205.14.99	9	1046	5	478	4	568	10.384853	0.2177
10.126.10.1	224.0.0.10	3	222	3	222	0	0	4.703285	9.2348
10.126.10.23	239.255.255.250	3	1161	3	1161	0	0	16.224060	0.2006
10.126.10.74	52.230.3.194	3	364	2	183	1	181	13.758018	0.0762
10.126.10.74	202.46.129.2	2	631	1	83	1	548	10.383140	0.0011

Gambar 6.3 Screenshot *conversation* DoS melalui UDP dan Tanpa Firewall pada Windows Server

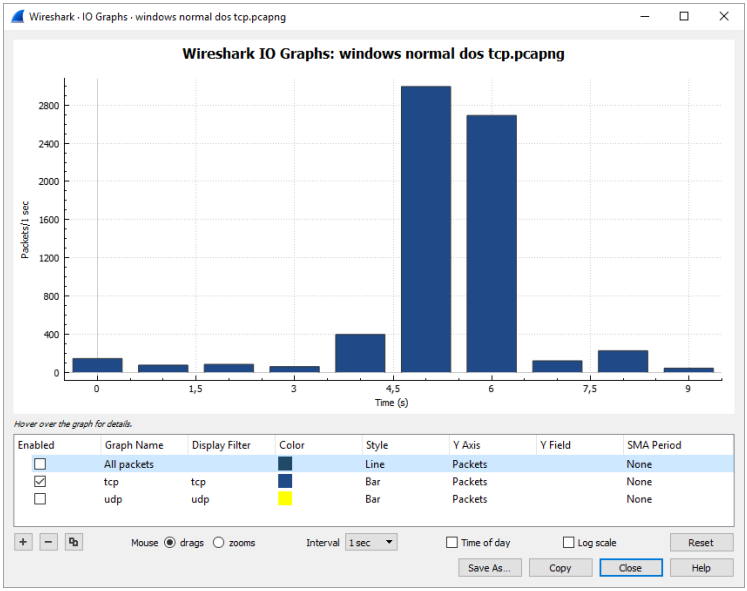
Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui UDP. IP penyerang pun lebih banyak mengirimkan daripada menerima server. Jumlah paket pada *conversation* antara server dan penyerang adalah 83018 paket. Jumlah bytes pada *conversation* kali ini adalah 5001 kb. Perbandingannya pun cukup jauh apabila dibandingkan dengan dos melalui TCP.Sekilas terlihat bahwa eksperimen dos menggunakan UDP lebih sedikit mengirimkan byte tetapi lebih banyak mengirimkan packet.



Gambar 6.4 Screenshot I/O Graph Kondisi Tanpa serangan dan Tanpa Firewall pada port TCP di Windows Server

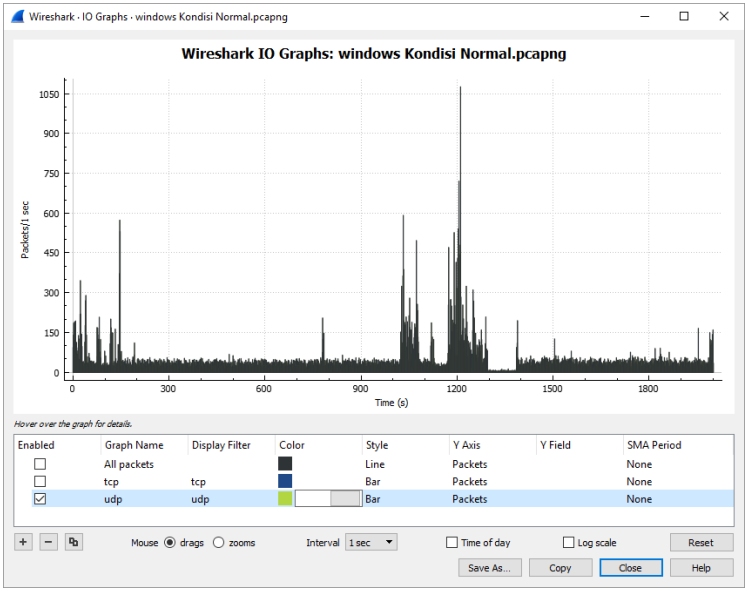
Untuk melihat grafik tersebut. Buka wireshark→statistics→I/O graph.

Gambar di atas adalah grafik dari trafik jaringan khususnya protokol tcp. Dilihat sekilas, nilai paket tertinggi adalah 360 paket/detik.Grafik di atas diambil ketika keadaan tanpa serangan atau tanpa eksperimen dos. Pada kondisi ini tidak ada aktifitas yang mencurigakan, grafik menandakan jaringan berjalan normal karena trafik cenderung sedikit dalam pengiriman paket.



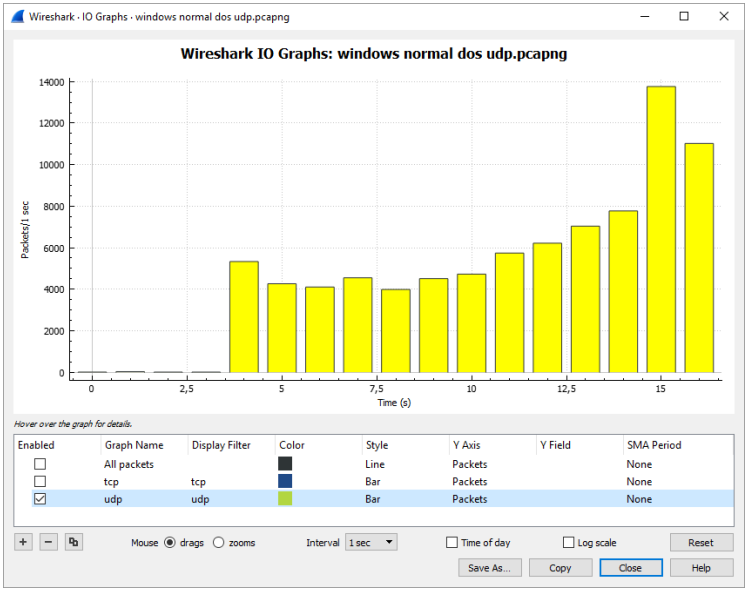
Gambar 6.5 Screenshot I/O Graph DoS melalui TCP dan Tanpa Firewall pada Windows Server

Gambar di atas merupakan grafik dari trafik jaringan ketika dilakukan eksperimen dos melalui TCP. Jumlah tertinggi packet per detik pada saat dilakukan dos adalah 2800 packet/detik. Berbeda jauh dengan kondisi tanpa serangan yaitu 360 packet/detik. Ini menandakan bahwa eksperimen dos cukup berpengaruh pada trafik jaringan.



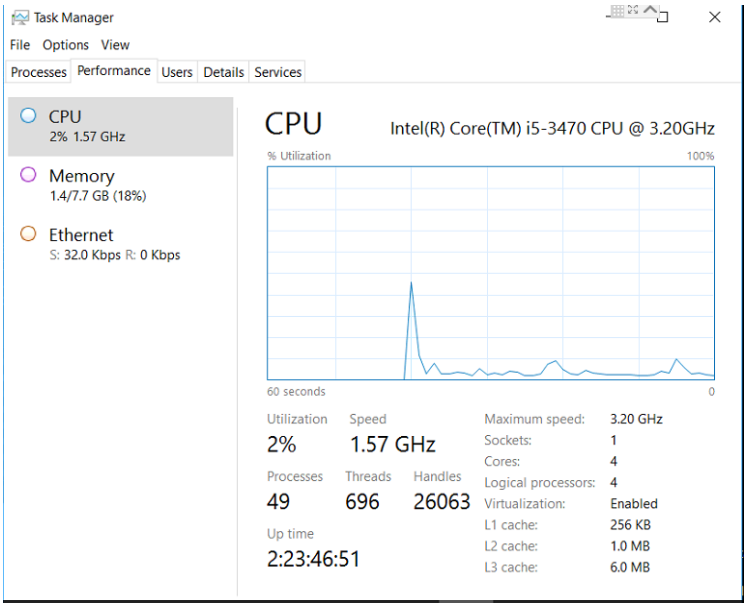
Gambar 6.6 Screenshot I/O Graph Kondisi Tanpa serangan dan Tanpa Firewall pada port UDP di Windows Server

Gambar di atas adalah kondisi trafik jaringan khususnya protokol udp saat kondisi tanpa serangan. Jumlah paket terbanyak adalah 3250 packet per detik. Grafik UDP lebih banyak dalam pengiriman paket ketimbang grafik TCP. Grafik di atas diambil ketika keadaan tanpa serangan atau tanpa eksperimen dos. Pada kondisi ini tidak ada aktifitas yang mencurigakan, grafik menandakan jaringan berjalan normal karena trafik cenderung sedikit dalam pengiriman paket.



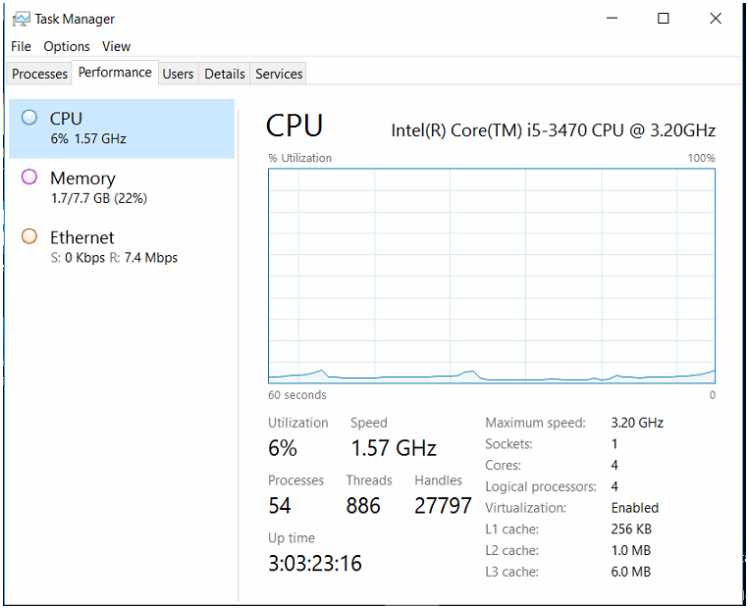
Gambar 6.7 Screenshot I/O Graph DoS melalui UDP dan Tanpa Firewall pada Windows Server

Gambar di atas adalah grafik dari trafik jaringan ketika dilakukan dos menggunakan udp. Nilai tertinggi pada saat melakukan dos adalah 14000 packet per detik. Berbeda jauh dengan kondisi tanpa serangan yang hanya 3000an paket per detik. Rata-rata packet per detik saat dilakukan dos pun di atas 4000 packet per detik. Setelah melihat hasil-hasil di atas, paket per detik pada UDP lebih banyak daripada TCP. Ini memandakan dos melalui UDP cukup berpengaruh.



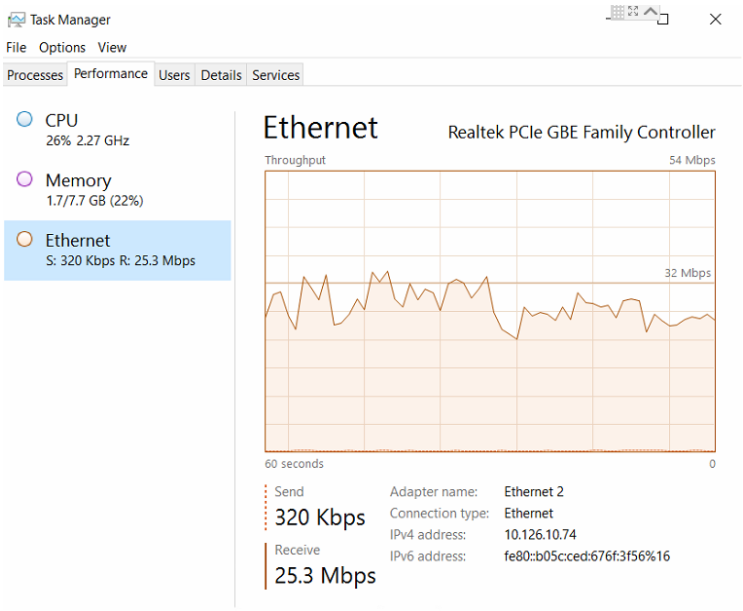
Gambar 6.8 Screenshot kinerja Windows Server saat Tanpa Serangan dan Tanpa Firewall

Gambar di atas adalah Screenshot saat kondisi tanpa serangan. Pada kondisi tanpa serangan, performa server tidak terlalu memakan resource yang ada. Gambar diatas diambil dari Task Manager pada feature Windows Server. Resource ini adalah CPU, memory, dan aktifitas jaringan. Pada kondisi ini, CPU hanya memakan 2% (1.5ghz), memory 18% (1,4gb) dan aktifitas jaringan hanya 32kbps



Gambar 6.9 Screenshot kinerja Windows Server saat DoS melalui TCP dan Tanpa Firewall

Gambar di atas adalah screnshoot saat dilakukan eksperimen dos ke server melalui protokol tcp. Pada eksperimen ini, CPU naik (4%), memory naik (6%) dan akitifitas jaringan naik pesat dimana yang sebelumnya maksimal 320kbps sekarang menjadi 7.4mbps. Ini menandakan bahwa dos melalui TCP cukup berpengaruh pada kinerja server. Selain itu, Windows Server cukup handal dalam menangani dos ini karena CPU dan memory tidak sampai 70%.

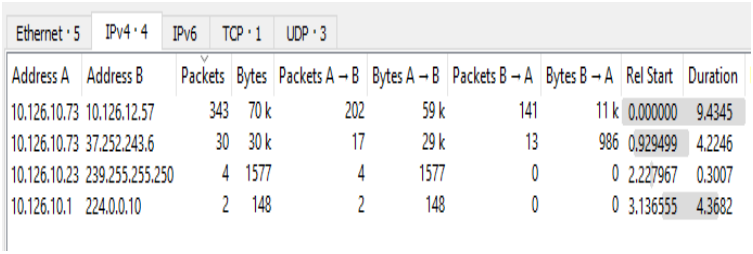


Gambar 6.10 Screenshot kinerja Windows Server saat DoS melalui UDP dan Tanpa Firewall

Gambar di atas adalah screenshot saat melakukan eksperimen dos ke server menggunakan protokol udp. CPU naik drastis (dari 2% menuju 26%), memory sama seperti dos melalui tcp (18% menjadi 22%), Ethernet sedikit dibawah dos melalui tcp (320Kbps menjadi 25.3mbps). Hal ini menandakan bahwa DOS melalui UDP lebih berpengaruh daripada DOS melalui TCP pada Windows Server.

b. Ubuntu

Berikut merupakan hasil eksperimen tanpa firewall pada sistem operasi Ubuntu Server 16.04.



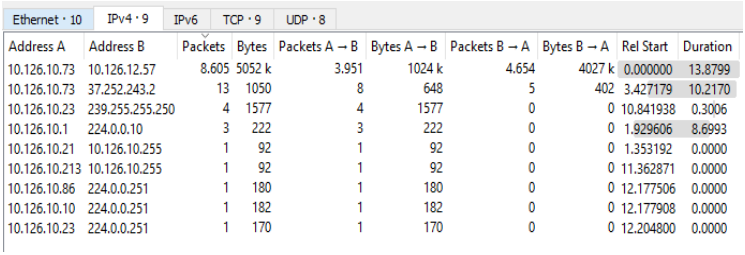
Ethernet · 5 | IPv4 · 4 | IPv6 | TCP · 1 | UDP · 3

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.57	343	70 k	202	59 k	141	11 k	0.000000	9.4345
10.126.10.73	37.252.243.6	30	30 k	17	29 k	13	986	0.929499	4.2246
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	2.227967	0.3007
10.126.10.1	224.0.0.10	2	148	2	148	0	0	3.136555	4.3682

Gambar 6.11 Screenshot conversation Kondisi Tanpa Serangan dan Tanpa Firewall di Ubuntu Server

Untuk melihat *conversation*, buka wireshark→statistics→conversation

Gambar di atas adalah *conversation* antar IP yang telah ditangkap oleh wireshark. Terlihat banyak pertukaran data terhadap beberapa IP. IP yang terlihat tidak selalu IP terhadap suatu PC, melainkan router atau switch. Gambar di atas merupakan kondisi tanpa serangan atau tanpa eksperimen dos. Jumlah bytes dan packets pada Ubuntu Server lebih sedikit karena pertukaran informasi pada Ubuntu server cenderung sedikit. Hal ini diawali dengan layanan yang menggunakan pada Ubuntu server yang sedikit ketimbang Windows Server.

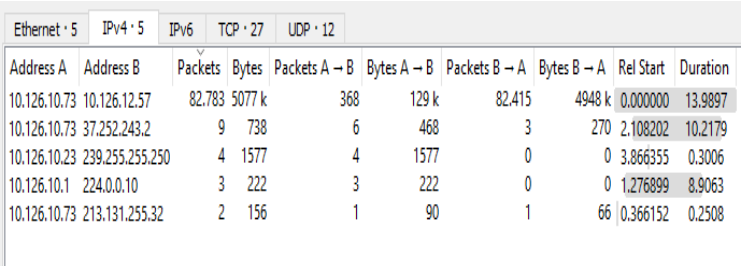


Ethernet · 10 | IPv4 · 9 | IPv6 | TCP · 9 | UDP · 8

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.57	8.605	5052 k	3.951	1024 k	4.654	4027 k	0.000000	13.8799
10.126.10.73	37.252.243.2	13	1050	8	648	5	402	3.427179	10.2170
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	10.841938	0.3006
10.126.10.1	224.0.0.10	3	222	3	222	0	0	1.929606	8.6993
10.126.10.21	10.126.10.255	1	92	1	92	0	0	1.353192	0.0000
10.126.10.213	10.126.10.255	1	92	1	92	0	0	11.362871	0.0000
10.126.10.86	224.0.0.251	1	180	1	180	0	0	12.177506	0.0000
10.126.10.10	224.0.0.251	1	182	1	182	0	0	12.177908	0.0000
10.126.10.23	224.0.0.251	1	170	1	170	0	0	12.204800	0.0000

Gambar 6.12 Screenshot conversation DoS melalui TCP dan Tanpa Firewall di Ubuntu Server

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui TCP. Apabila diurutkan berdasarkan packet dan byte, IP penyerang menduduki posisi paling atas. Jumlah paket pada conversation antara server dan penyerang adalah 8605 paket, sedangkan jumlah bytesnya adalah 5052 kb. Pada Jumlah paket naik drastis dari 343 menjadi 8605. Sedangkan jumlah byte naik dari dari 70 kb menjadi 5052 kb. Hal ini menandakan bahwa dos melalui TCP sangat berpengaruh pada jumlah bytes dan jumlah packets.

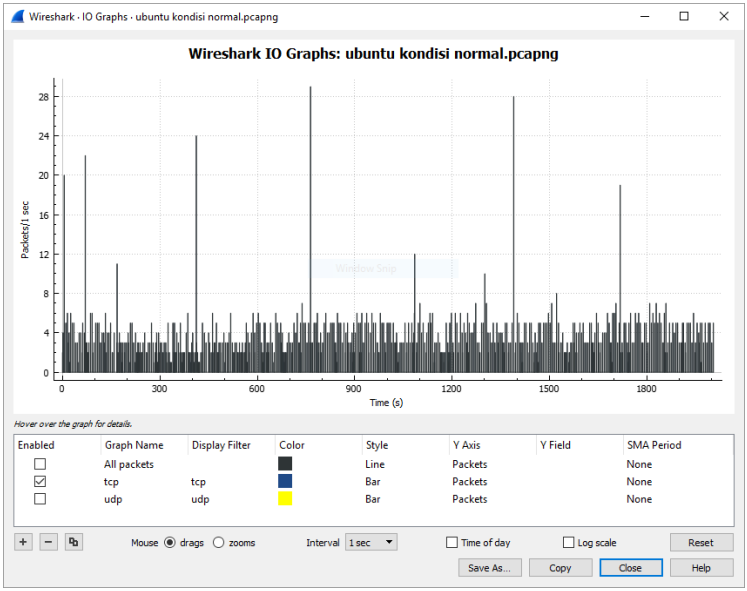


Ethernet · 5 | IPv4 · 5 | IPv6 | TCP · 27 | UDP · 12

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.57	82.783	5077 k	368	129 k	82.415	4948 k	0.000000	13.9897
10.126.10.73	37.252.243.2	9	738	6	468	3	270	2.108202	10.2179
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	3.866355	0.3006
10.126.10.1	224.0.0.10	3	222	3	222	0	0	1.276899	8.9063
10.126.10.73	213.131.255.32	2	156	1	90	1	66	0.366152	0.2508

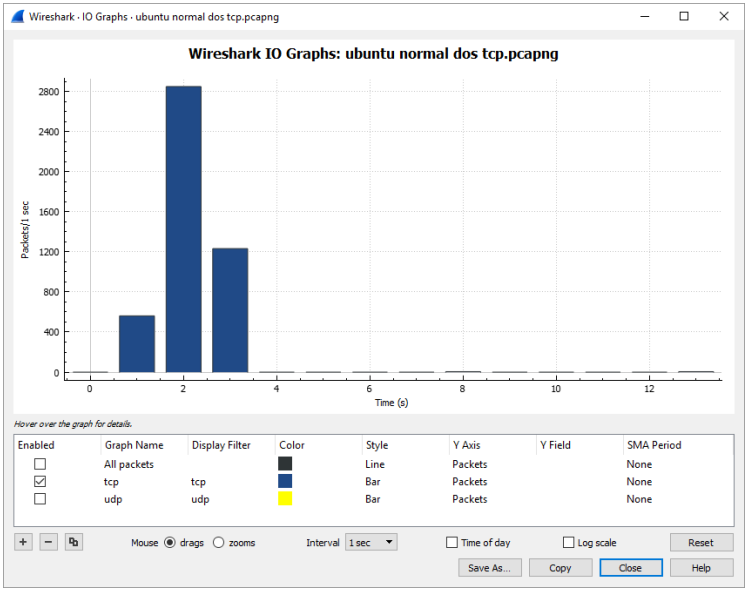
Gambar 6.13 Screenshot *conversation* DoS melalui UDP dan Tanpa Firewall di Ubuntu Server

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui UDP. Apabila diurutkan berdasarkan packet dan byte, IP penyerang ada pada posisi paling atas. Jumlah packet pada conversation ini adalah adalah 82783 paket sedangkan jumlah bytesnya dalah 5077kb. IP penyerang pun lebih banyak mengirimkan daripada menerima server. Perbandingannya pun cukup jauh apabila dibandingkan dengan dos melalui TCP.



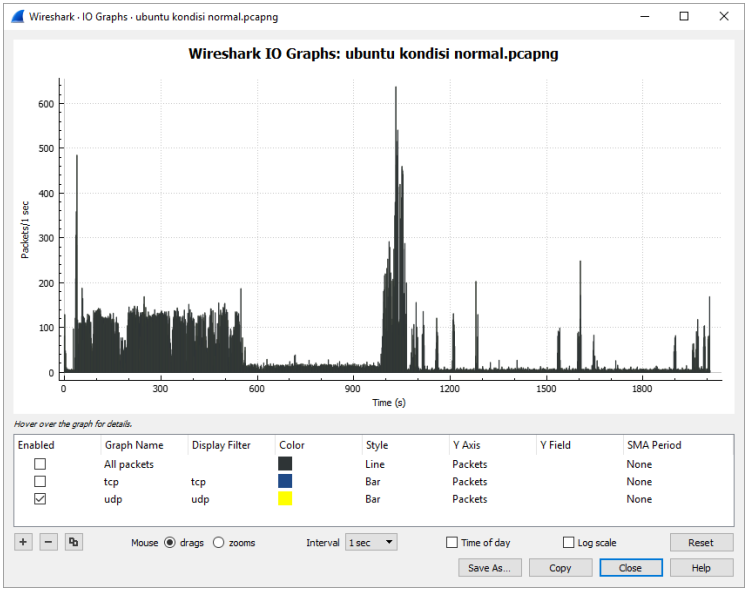
Gambar 6.14 Screenshot I/O graph TCP Kondisi Tanpa Serangan dan Tanpa Firewall di Ubuntu Server

Gambar di atas adalah kondisi trafik jaringan khususnya protokol tcp. Jumlah paket tertinggi pada kondisi ini adalah 28 paket per detik. Kondisi di atas merupakan kondisi tanpa serangan atau tanpa eksperimen dos apapun. Pada kondisi ini, jaringan cenderung lebih tenang dan jarak maksimal packet dan minimal yang kecil. Hal ini menandakan bahwa tidak ada aktifitas yang mencurigakan pada kondisi ini.



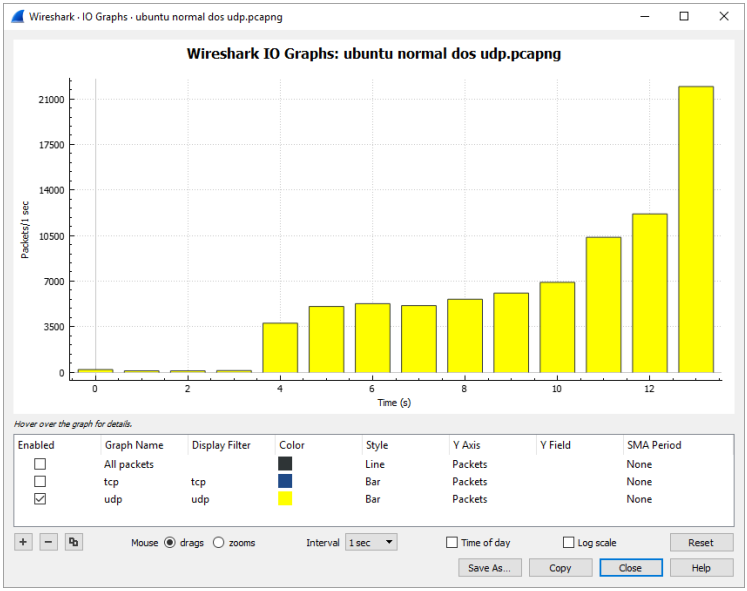
Gambar 6.15 Screenshot I/O graph saat DoS melalui TCP dan tanpa firewall di Ubuntu Server

Gambar di atas adalah kondisi trafik ketika dilakukan eksperimen serangan dos melalui tcp. Nilai tertinggi pada kondisi ini adalah 2800 paket per detik. Jumlah packet naik drastis dari yang hanya berkisar 0-28 menjadi 0-2800. Nilai ini sama seperti apa yang ada pada windows. Hal ini menandakan bahwa dos melalui TCP cukup berpengaruh pada jumlah bytes dan packet di Ubuntu Server.



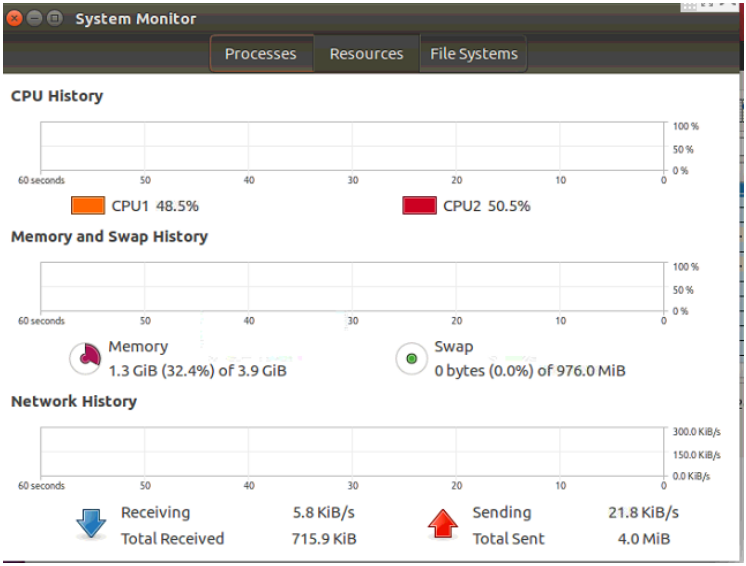
Gambar 6.16 Screenshot I/O Graph UDP dan Tanpa Firewall di Ubuntu Server

Gambar di atas adalah kondisi trafik jaringan khususnya protokol udp saat kondisi tanpa serangan. Jumlah paket terbanyak adalah 600 packet per detik.Pada kondisi ini, jaringan cenderung lebih tenang dan tidak ada jarak maksimal-minimal tiap paket tidak terlalu besar. Hal ini menandakan bahwa tidak ada aktifitas yang mencurigakan.



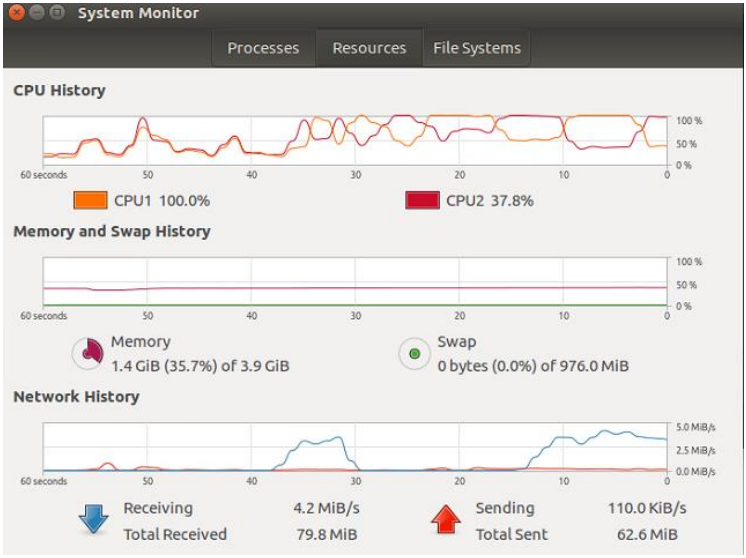
Gambar 6.17 Screenshot I/O Graph DoS melalui UDP dan Tanpa Firewall di Ubuntu Server

Gambar di atas adalah grafik dari trafik jaringan ketika dilakukan dos menggunakan udp. Nilai tertinggi pada saat melakukan dos adalah 21000 packet per detik. Berbeda jauh dengan kondisi tanpa serangan yang nilai maksimal hanya 600an paket per detik. Jumlah DOS melalui UDP berbeda dengan DOS melalui TCP, dimana DOS melalui nilai maksimal hanya 2400. Hal ini menandakan bahwa dos melalui UDP cukup berpengaruh pada jumlah packet dan bytes di Ubuntu Server



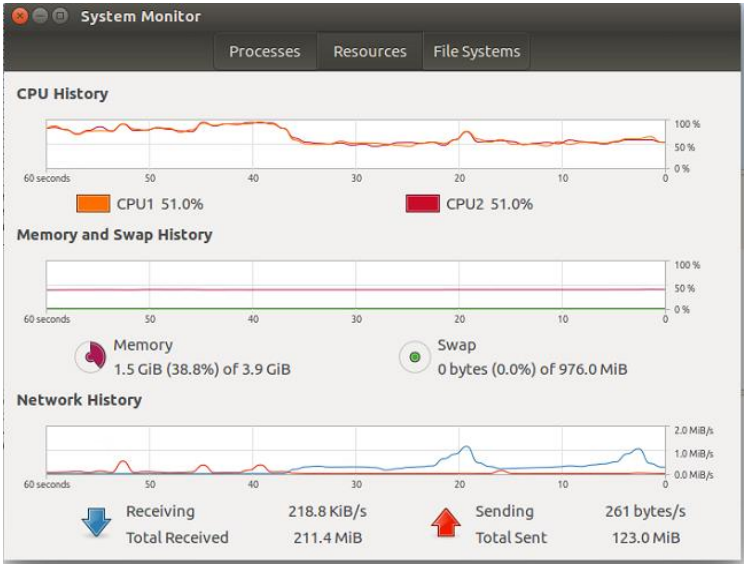
Gambar 6.18 Screenshot Kinerja saat Kondisi Tanpa Serangan dan Tanpa Firewall pada Ubuntu Server

Gambar di atas adalah Screenshot saat kondisi tanpa serangan. Pada kondisi tanpa serangan, performa server tidak terlalu memakan resource yang ada. Resource ini adalah CPU, memory, dan jaringan. Berbeda dengan windows, kondisi tanpa serangan pada Ubuntu cukup memakan CPU. Dibandingkan dengan windows yang hanya menggunakan 2%, Ubuntu server menggunakan 48%. Untuk memory, Ubuntu server unggul 0.1 gb dari Windows Server.



Gambar 6.19 Screenshot Kinerja saat DoS melalui TCP dan Tanpa Firewall pada Ubuntu Server

Gambar di atas adalah Screenshot saat eksperimen serangan dos melalui tcp. Pada kondisi ini, CPU 1 menjadi 100% tetapi CPU2 menjadi 37%. Memory menjadi 35.7%. Aktifitas jaringan pada kondisi ini berkisar 62 sampai 79 mbps. Dibandingkan dengan kondisi tanpa serangan, CPU kali ini naik 50%, memory menjadi 1.4gb dan aktifitas jaringan naik sekitar 60an mbps. Hal ini menandakan bahwa DoS melalui TCP sangat berpengaruh pada CPU, memory dan Aktifitas jaringan di Ubuntu Server. CPU pada Ubuntu server pun melebihi 70% yang artinya melebihi kinerja pada server sesungguhnya.



Gambar 6.20 Screenshot Kinerja saat DoS melalui UDP dan Tanpa Firewall pada Ubuntu Server

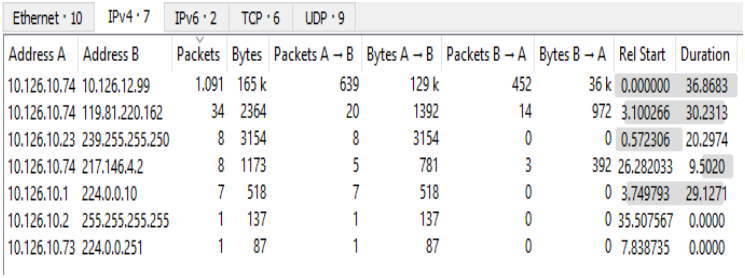
Gambar di atas adalah Screenshot saat eksperimen dos melalui udp. Pada kondisi ini, performa server tidak terlalu memakan resource yang ada. Perbedaanya adalah konsumsi RAM naik dan trafik menjadi padat (123mbps – 211mbps). Pada kondisi ini CPU tidak naik drastis seperti DOS melalui TCP. Akan tetapi aktifitas jaringan jauh lebih banyak daripad DOS melalui TCP. Hal ini menandakan bahwa DOS melalui UDP cukup berpengaruh pada kinerja server di Ubuntu Server walaupun tidak signifikan.

6.1.2 Hasil Eksperimen dengan Firewall dan Konfigurasi

Eksperimen kedua merupakan eksperimen dos ke server dengan mengkonfigurasi firewall. Hasil dibedakan antara Windows Server dan Ubuntu Server

a. Windows

Berikut merupakan hasil eksperimen dengan firewall dan konfigurasi pada sistem operasi Microsoft Windows Server 2016.



Ethernet · 10 | IPv4 · 7 | IPv6 · 2 | TCP · 6 | UDP · 9

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.99	1.091	165 k	639	129 k	452	36 k	0.000000	36.8683
10.126.10.74	119.81.220.162	34	2364	20	1392	14	972	3.100266	30.2313
10.126.10.23	239.255.255.250	8	3154	8	3154	0	0	0.572306	20.2974
10.126.10.74	217.146.4.2	8	1173	5	781	3	392	26.282033	9.5020
10.126.10.1	224.0.0.10	7	518	7	518	0	0	3.749793	29.1271
10.126.10.2	255.255.255.255	1	137	1	137	0	0	35.507567	0.0000
10.126.10.73	224.0.0.251	1	87	1	87	0	0	7.838735	0.0000

Gambar 6.21 Screenshot Conversation saat Kondisi Tanpa serangan pada Windows Server dengan Firewall dan Konfigurasi

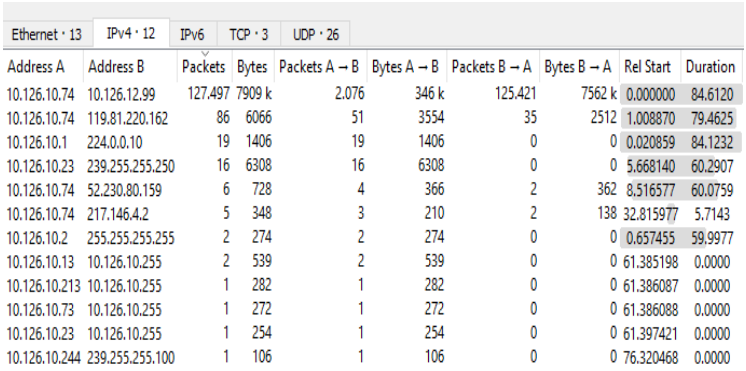
Gambar di atas adalah *conversation* antar IP yang telah ditangkap oleh wireshark. Terlihat banyak pertukaran data terhadap beberapa IP. IP yang terlihat tidak selalui IP terhadap suatu PC, melainkan router atau switch. Pada dasarnya *conversation* pada eksperimen kali ini tidak beda jauh dengan apa yang ada pada eksperimen tanpa firewall di Windows Server. Hal ini dikarenakan jumlah packet dan bytes pada eksperimen kali ini tidak beda jauh dari eksperimen tanpa firewall. Jumlah packet dan bytes pada eksperimen tanpa firewall adalah 1363 packet dan 292 kb. Hal ini menandakan bahwa penggunaan firewall tidak mempengaruhi secara signifikan pada jumlah bytes dan packets.

Ethernet · 15 | IPv4 · 14 | IPv6 | TCP · 19 | UDP · 16

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.99	9.591	1703 k	5.596	1377 k	3.995	326 k	0.000000	79.8936
10.126.10.74	119.81.220.162	81	5692	47	3290	34	2402	0.850964	76.4665
10.126.10.1	224.0.0.10	18	1332	18	1332	0	0	0.714506	79.2246
10.126.10.23	239.255.255.250	16	6308	16	6308	0	0	10.624600	60.2915
10.126.10.74	52.230.80.159	10	1244	8	882	2	362	13.426920	60.0767
10.126.10.74	217.146.4.2	10	696	6	420	4	276	15.049001	61.0508
10.126.10.2	255.255.255.255	2	274	2	274	0	0	5.587713	59.9921
10.126.10.10	224.0.0.251	1	182	1	182	0	0	29.671815	0.0000
10.126.10.21	10.126.10.255	1	92	1	92	0	0	40.349302	0.0000
10.126.10.23	224.0.0.251	1	170	1	170	0	0	29.767610	0.0000
10.126.10.73	10.126.10.255	1	92	1	92	0	0	35.810206	0.0000
10.126.10.86	224.0.0.251	1	180	1	180	0	0	29.670930	0.0000
10.126.10.213	10.126.10.255	1	92	1	92	0	0	35.809362	0.0000
10.126.10.244	239.255.255.100	1	106	1	106	0	0	21.139394	0.0000

Gambar 6.22 Screenshot Conversation saat DoS melalui TCP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui TCP. Terlihat IP penyerang melakukan pertukaran data yang banyak. Apabila diurutkan berdasarkan packet dan byte, IP penyerang menduduki posisi paling atas. IP penyerang pun lebih banyak mengirimkan daripada menerima server. Terdapat 9391 paket dan 1703 kb pada saat komunikasi antara penyerang dan server. Berbeda jauh pada kondisi DOS melalui TCP tetapi tanpa firewall yaitu 5984 packet dan 6193kb. Hal ini menandakan bahwa firewall berpengaruh pada jumlah packet dan bytes saat DOS melalui TCP di Windows Server.

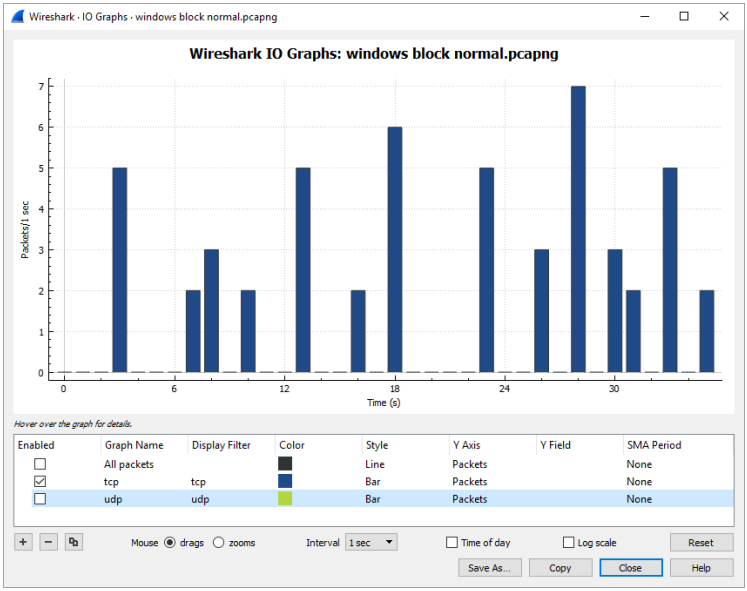


Ethernet · 13 | IPv4 · 12 | IPv6 | TCP · 3 | UDP · 26

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.74	10.126.12.99	127.497	7909 k	2.076	346 k	125.421	7562 k	0.000000	84.6120
10.126.10.74	119.81.220.162	86	6066	51	3554	35	2512	1.008870	79.4625
10.126.10.1	224.0.0.10	19	1406	19	1406	0	0	0.020859	84.1232
10.126.10.23	239.255.255.250	16	6308	16	6308	0	0	5.668140	60.2907
10.126.10.74	52.230.80.159	6	728	4	366	2	362	8.516577	60.0759
10.126.10.74	217.146.4.2	5	348	3	210	2	138	32.815977	5.7143
10.126.10.2	255.255.255.255	2	274	2	274	0	0	0.657455	59.9977
10.126.10.13	10.126.10.255	2	539	2	539	0	0	61.385198	0.0000
10.126.10.213	10.126.10.255	1	282	1	282	0	0	61.386087	0.0000
10.126.10.73	10.126.10.255	1	272	1	272	0	0	61.386088	0.0000
10.126.10.23	10.126.10.255	1	254	1	254	0	0	61.397421	0.0000
10.126.10.244	239.255.255.100	1	106	1	106	0	0	76.320468	0.0000

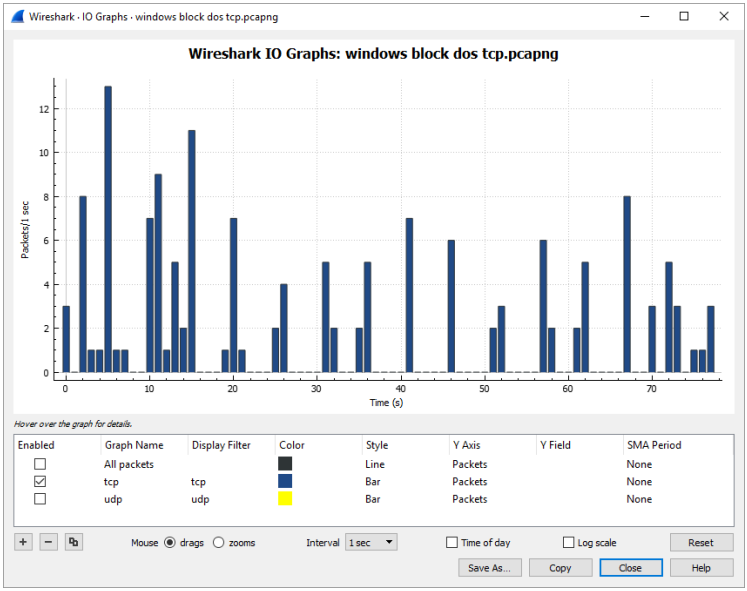
Gambar 6.23 Screenshot Conversation saat DoS melalui UDP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui UDP. IP penyerang pun lebih banyak mengirimkan daripada menerima server. Jumlah paket pada *conversation* antara server dan penyerang adalah 127497 paket. Jumlah bytes pada *conversation* kali ini adalah 7909 kb. Perbandingannya pun cukup jauh apabila dibandingkan dengan dos melalui TCP. Pada dasarnya eksperimen kali ini tidak berbeda jauh apabila dibandingkan dengan tanpa firewall. Jumlah packet dan bytes pada eksperimen tanpa firewall adalah 83018 packet dan 5001kb. Hal ini menandakan bahwa tidak ada perbedaan antara eksperimen tanpa firewall dan menggunakan firewall.



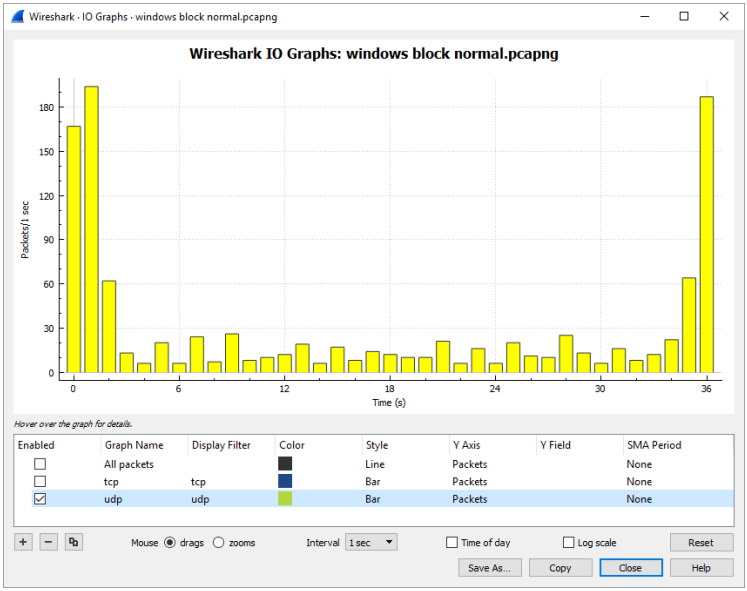
Gambar 6.24 Screenshot I/O Graph TCP saat Kondisi Tanpa Serangan pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah grafik dari trafik jaringan khususnya protokol tcp. Dilihat sekilas, nilai paket tertinggi adalah 7 paket/detik. Grafik di atas diambil ketika keadaan tanpa serangan dan menggunakan firewall. Apabila dibandingkan dengan eksperimen firewall, jumlah packet tertingginya adalah 360 packets/detik. Perbandingannya cukup signifikan, hal ini menandakan bahwa penggunaan firewall membuat jaringan lebih tenang dan jarak maksimal-minimal tiap packet menjadi kecil.



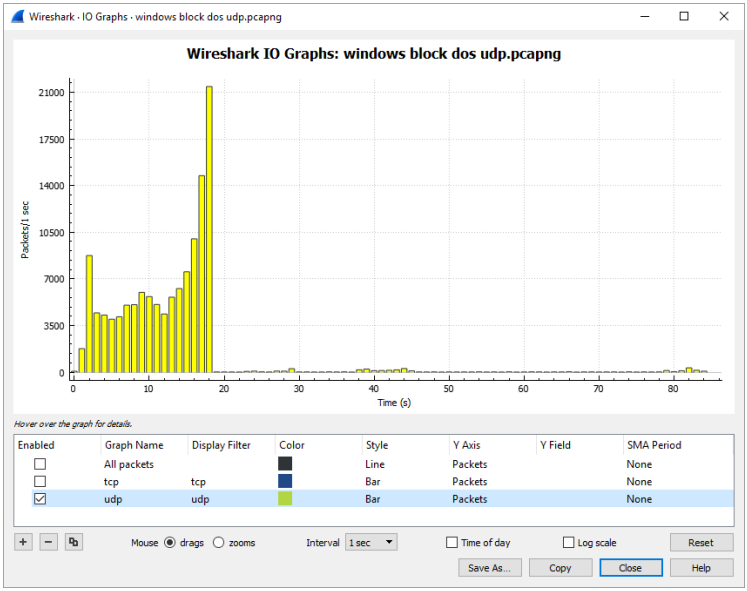
Gambar 6.25 Screenshot I/O Graph saat DoS melalui TCP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas merupakan grafik dari trafik jaringan ketika dilakukan eksperimen dos melalui TCP. Jumlah tertinggi packet per detik pada saat dilakukan dos adalah 12 packet/detik. Berbeda sedikit dengan kondisi tanpa serangan yaitu 7 packet/detik. Apabila dibandingkan dengan kondisi tanpa firewall, jumlah packet tertingginya adalah 2800 packet/detik. Hal ini menandakan bahwa penggunaan firewall santa berpengaruh pada DOS melalui TCP di Windows Server.



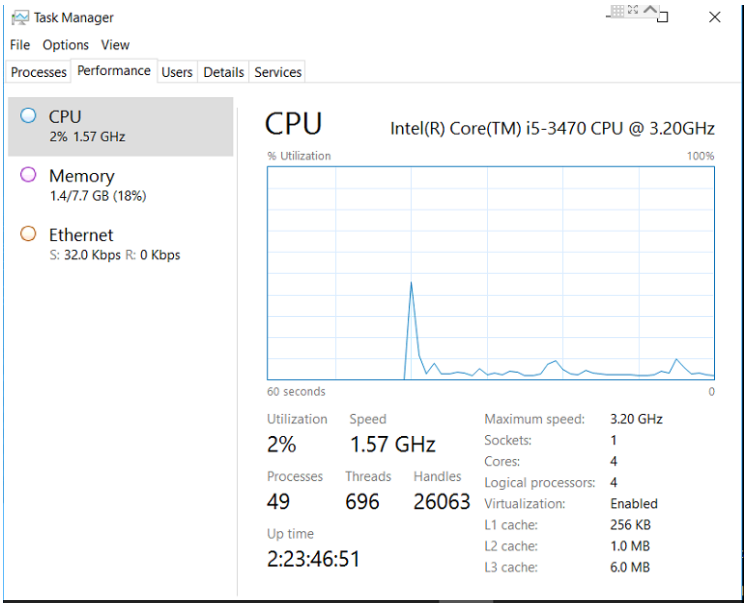
Gambar 6.26 Screenshot I/O Graph UDP saat Kondisi Tanpa serangan pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah kondisi trafik jaringan khususnya protokol udp saat kondisi tanpa serangan. Jumlah paket terbanyak adalah 180 packet per detik. Grafik UDP lebih banyak apabila dibandingkan dengan grafik TCP pada kasus yang sama. Apabila dibandingkan dengan eksperimen tanpa firewall, jumlah packetnya adalah 3250 packet per detik. Ini menandakan bahwa firewall berpengaruh pada protokol udp saat kondisi tanpa serangan.



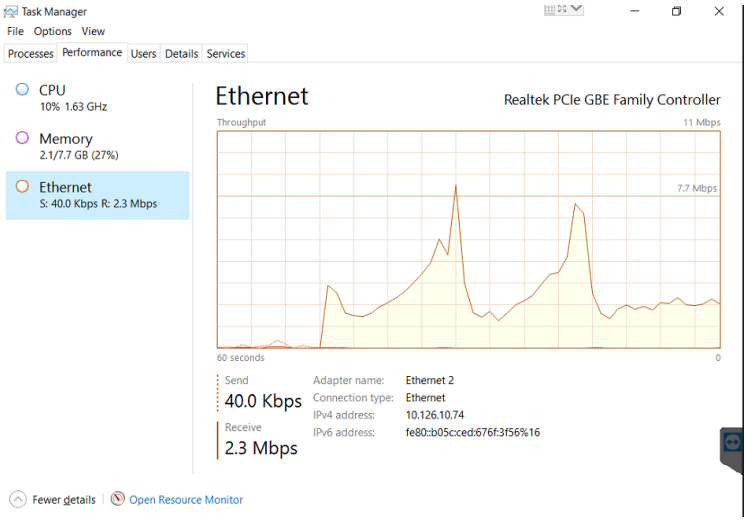
Gambar 6.27 Screenshot I/O Graph saat DoS melalui UDP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah grafik dari trafik jaringan ketika dilakukan dos menggunakan udp. Nilai tertinggi pada saat melakukan dos adalah 21000 packet per detik. Berbeda jauh dengan kondisi tanpa serangan yang hanya 180 paket per detik. Apabila dibandingkan dengan DOS tcp dengan firewall, jumlah packetnya sangat jauh yaitu 12 packet per detik. Ketika dibandingkan dengan kondisi tanpa firewall, jumlah packetnya adalah 14000 packet per detik. Hal ini menandakan bahwa firewall tidak berpengaruh secara signifikan terhadap serangan dos melalui UDP di Windows Server.



Gambar 6.28 Screenshot Kinerja saat Kondisi Tanpa serangan dan DoS melalui TCP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah Screenshot saat kondisi tanpa serangan dan firewall dikonfigurasi untuk protokol tcp. Pada kondisi ini performa server tidak terlalu memakan resource yang ada, bahkan dos melalui tcp hasilnya sama seperti tanpa serangan pada performa server di windows server. Hal ini menandakan bahwa penggunaan firewall cukup signifikan terhadpa serangan DOS melalui TCP di Windows Server.

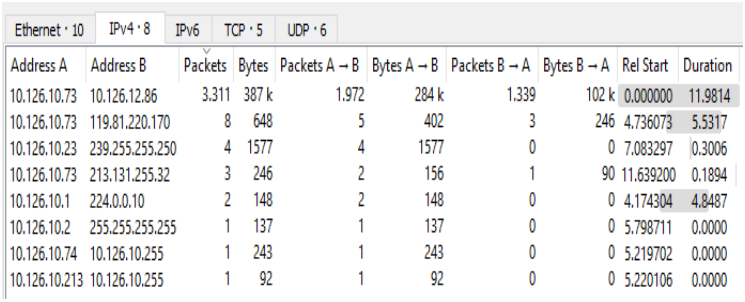


Gambar 6.29 Screenshot Kinerja saat DoS melalui UDP pada Windows Server dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot saat melakukan eksperimen dos ke server menggunakan protokol udp. CPU naik dari 2% menuju 10%, memory naik dari 18% menjadi 27%, Ethernet berbeda dengan kondisi-kondisi sebelumnya yaitu 40Kbps sampai 2.3mbps. Hal ini menandakan bahwa konfigurasi firewall dengan memblokir protokol udp tidak berpengaruh signifikan pada serangan dos melalui UDP.

b. Ubuntu

Berikut merupakan hasil eksperimen dengan firewall dan konfigurasi pada sistem operasi Ubuntu Server 16.04.

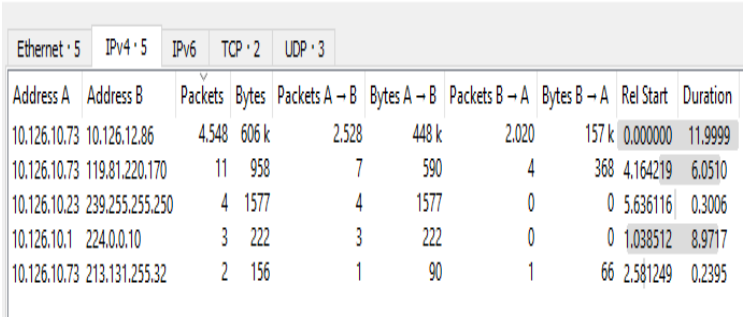


Ethernet · 10 | IPv4 · 8 | IPv6 | TCP · 5 | UDP · 6

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.86	3.311	387 k	1.972	284 k	1.339	102 k	0.000000	11.9814
10.126.10.73	119.81.220.170	8	648	5	402	3	246	4.736073	5.5317
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	7.083297	0.3006
10.126.10.73	213.131.255.32	3	246	2	156	1	90	11.639200	0.1894
10.126.10.1	224.0.0.10	2	148	2	148	0	0	4.174304	4.8487
10.126.10.2	255.255.255.255	1	137	1	137	0	0	5.798711	0.0000
10.126.10.74	10.126.10.255	1	243	1	243	0	0	5.219702	0.0000
10.126.10.213	10.126.10.255	1	92	1	92	0	0	5.220106	0.0000

Gambar 6.30 Screenshot Conversation saat Kondisi Tanpa serangan pada Ubuntu Server Eksperimen dengan Firewall dan Konfigurasi

Gambar di atas adalah *conversation* antar IP yang telah ditangkap oleh wireshark saat kondisi tanpa serangan. Terlihat banyak pertukaran data terhadap beberapa IP. IP yang terlihat tidak selalu IP terhadap suatu PC, melainkan router atau switch. Gambar di atas adalah kondisi ketika tanpa serangan dan menggunakan firewall dimana jumlah packets dan bytesnya relatif kecil yaitu 3311 packet dan 387kb.



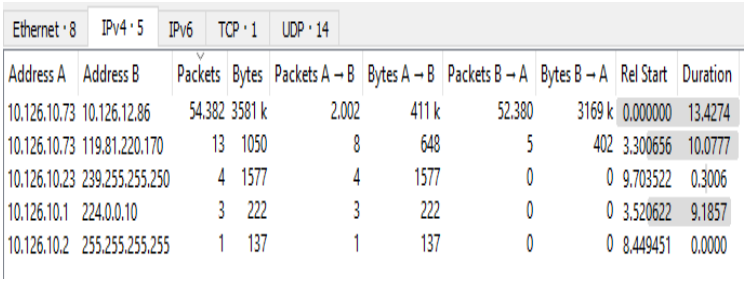
Ethernet · 5 | IPv4 · 5 | IPv6 | TCP · 2 | UDP · 3

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.86	4.548	606 k	2.528	448 k	2.020	157 k	0.000000	11.9999
10.126.10.73	119.81.220.170	11	958	7	590	4	368	4.164219	6.0510
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	5.636116	0.3006
10.126.10.1	224.0.0.10	3	222	3	222	0	0	1.038512	8.9717
10.126.10.73	213.131.255.32	2	156	1	90	1	66	2.581249	0.2395

Gambar 6.31 Screenshot Conversation saat DoS melalui TCP pada Ubuntu Server Eksperimen dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui TCP. Terlihat IP penyerang

melakukan pertukaran data yang tidak begitu banyak. Terdapat 4548 paket dan 606 kb pada saat komunikasi antara penyerang dan server. Apabila dibandingkan dengan tanpa firewall, tidak ada selisih yang signifikan. Dibandingkan dengan tanpa serangan pun tidak ada perbedaan juga. Ini terjadi karena serangan dos melalui TCP dapat diblokir oleh firewall. Hal ini menandakan bahwa penggunaan firewall cukup berpengaruh pada kasus DOS melalui TCP di Ubuntu Server.

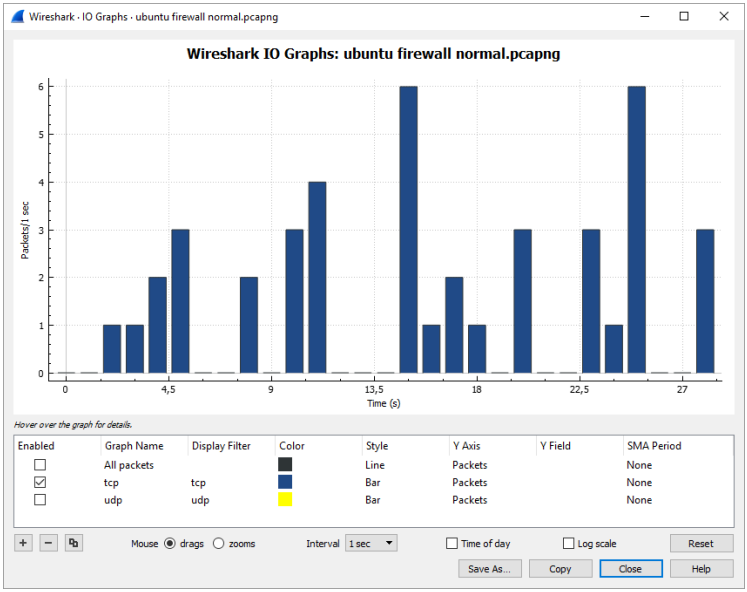


Ethernet · 8 | IPv4 · 5 | IPv6 | TCP · 1 | UDP · 14

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration
10.126.10.73	10.126.12.86	54.382	3581 k	2.002	411 k	52.380	3169 k	0.000000	13.4274
10.126.10.73	119.81.220.170	13	1050	8	648	5	402	3.300656	10.0777
10.126.10.23	239.255.255.250	4	1577	4	1577	0	0	9.703522	0.3006
10.126.10.1	224.0.0.10	3	222	3	222	0	0	3.520622	9.1857
10.126.10.2	255.255.255.255	1	137	1	137	0	0	8.449451	0.0000

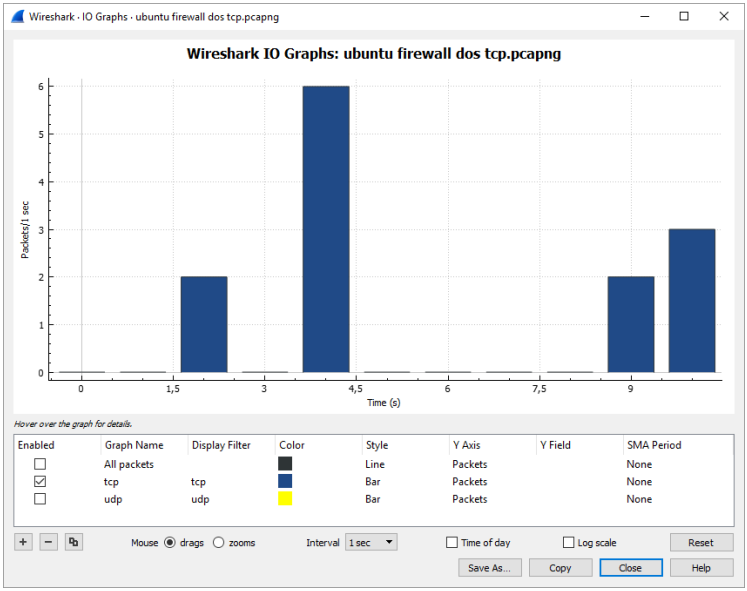
Gambar 6.32 Screenshot Conversation saat DoS melalui UDP pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot *conversation* pada saat dilakukan eksperimen dos melalui UDP. IP penyerrang terlihat melakukan pertukaran data yang banyak.Terdapat 54382 paket dan 3153 kb pada saat komunikasi antara penyerang dan server. Kondisi ini tidak jauh berbeda dengan Ubuntu tanpa firewall. Ini menandakan bahwa firewall tidak dapat menangkal serangan dos melalui UDP. Tetapi hal ini sangat berbeda dengan kondisi tanpa serangan dimana jumlah packet dan bytesnya adalah 3311 packet dan 387kb.



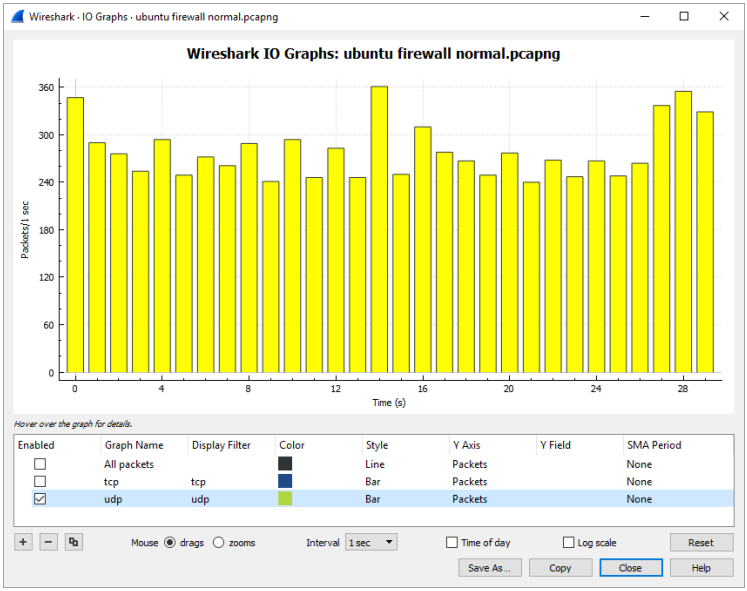
Gambar 6.33 Screenshot I/0 Graph TCP saat Kondisi Tanpa serangan pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah grafik dari trafik jaringan khususnya protokol tcp. Dilihat sekilas, nilai paket tertinggi adalah 6 paket per detik. Grafik di atas diambil ketika keadaan tanpa serangan dengan firewall terpasang. Sedikit berbeda dengan kondisi tanpa firewall, yaitu 28 paket per detik. Pada kondisi ini tidak ada aktifitas yang mencurigakan, selain itu dengan adanya firewall jumlah packet pada kondisi ini sedikit berbeda.



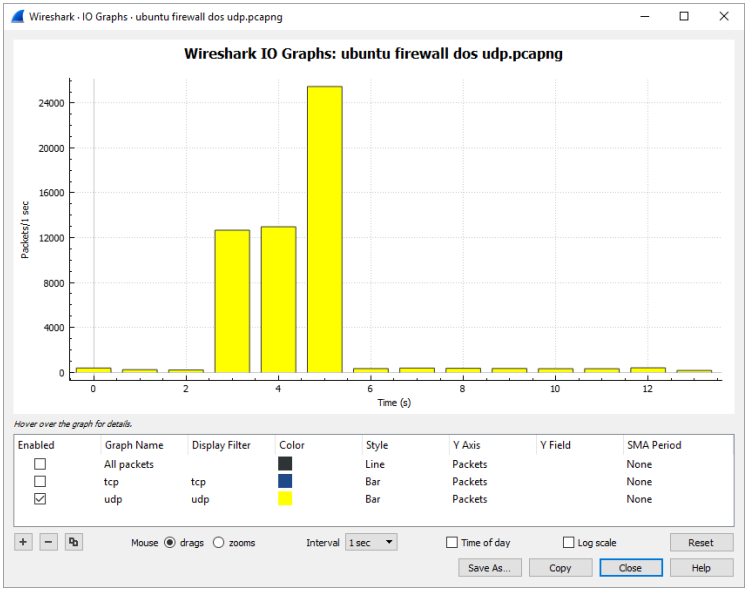
Gambar 6.34 Screenshot I/0 Graph saat DoS melalui TCP pada Ubuntu Server Eksperimen dengan Firewall dan Konfigurasi

Gambar di atas merupakan grafik dari trafik jaringan ketika dilakukan eksperimen dos melalui TCP. Jumlah tertinggi packet per detik pada saat dilakukan dos adalah 6 packet/detik. Tidak ada perbedaan signifikan apabila dibandingkan dengan kondisi tanpa serangan. Apabila dibandingkan dengan tanpa firewall, jumlah paketnya antara 0-2800 paket per detik. Ini menandakan firewall efektif menangkal serangan dos melalui TCP di Windows Server.



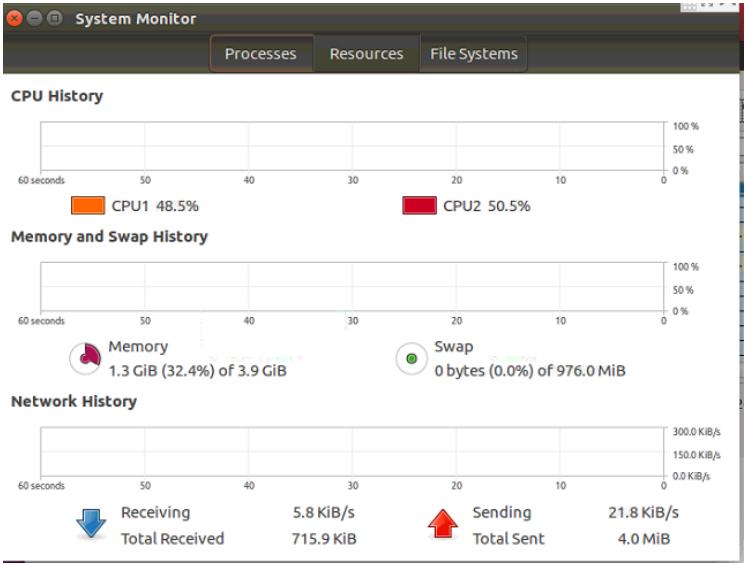
Gambar 6.35 Screenshot I/0 Graph UDP saat Kondisi Tanpa serangan pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah kondisi trafik jaringan khususnya protokol udp saat kondisi tanpa serangan. Jumlah paket terbanyak adalah 360 packet per detik. Grafik UDP lebih banyak dalam pengiriman paket ketimbang grafik TCP. Hal ini berbeda jauh apabila dibandingkan pada nilai maksimal Grafik UDP tanpa firewall yaitu 600 packet per detik. Hal ini menandakan bahwa firewall sedikit mengurangi packet per detik pada Ubuntu Server. Grafik di atas juga menandakan bahwa tidak ada aktifitas yang mencurigakan dan pengiriman paket cenderung sedikit.



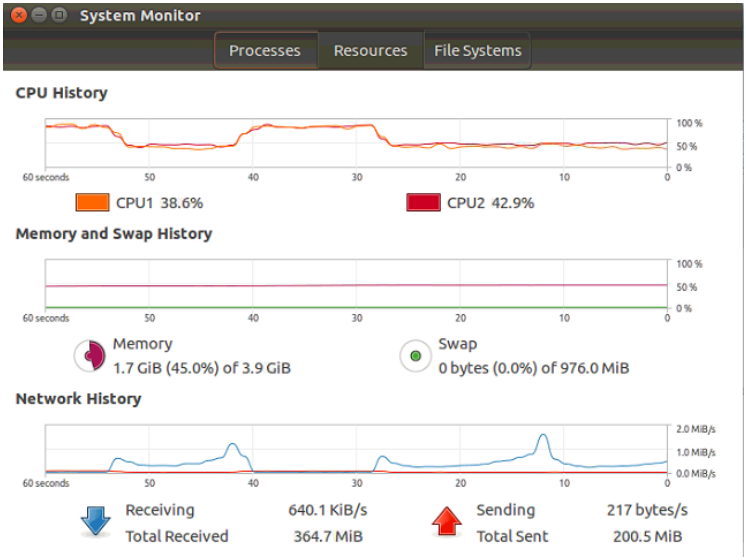
Gambar 6.36 Screenshot I/0 Graph saat DoS melalui UDP pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah grafik dari trafik jaringan ketika dilakukan dos menggunakan udp. Nilai tertinggi pada saat melakukan dos adalah 24000 packet per detik. Berbeda jauh dengan kondisi tanpa serangan yang hanya 360 paket per detik. Berbeda sedikit dengan kondisi tanpa firewall yaitu 21000 packet per detik. Apabila dibandingkan dengan DoS melalui TCP, jumlah packet pada DoS melalui UDP lebih banyak. Disamping itu, penggunaan firewall pada kasus ini tidak berpengaruh secara signifikan.



Gambar 6.37 Screenshot Kinerja saat Kondisi Tanpa serangan dan DoS melalui TCP pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah Screenshot saat kondisi tanpa serangan dan firewall saat memblokir tcp. Pada kondisi ini, Pada kondisi ini performa server tidak terlalu memakan resource yang ada, bahkan dos melalui tcp hasilnya sama seperti tanpa serangan pada performa server di windows server. Hal ini menandakan bahwa penggunaan firewall cukup signifikan terhadap serangan dos melalui TCP di Ubuntu Server.



Gambar 6.38 Screenshot Kinerja saat DoS melalui UDP pada Ubuntu Server dengan Firewall dan Konfigurasi

Gambar di atas adalah screenshot saat melakukan eksperimen dos ke server menggunakan protokol udp. CPU cukup berpengaruh signifikan yaitu naik 80%, penggunaan memory naik menjadi 45%, trafik jaringan menjadi tinggi yaitu 364mbps. Hal ini menandakan bahwa penggunaan firewall tidak signifikan dalam menangani serangan dos melalui UDP. Selain itu, Ubuntu Server memakan resource lebih dari 70% dalam penggunaan CPU pada kasus ini. Hal ini dapat membuat server menjadi down.

6.2 Perbandingan Hasil Eksperimen

Berdasarkan penelitian yang telah dilakukan, berikut merupakan perbandingan hasil eksperimen dalam penelitian ini:

Log wireshark pada kondisi tanpa serangan tidak ada perbedaan karakteristik dengan firewall maupun tidak. Pada kasus TCP, terdapat segment SYN—SYN-ACK—ACK. Log wireshark pada kondisi serangan DOS melalui TCP bisa dilihat pada lampiran A. Tidak ada perbedaan karakteristik pada log wireshark di Windows Server dan Ubuntu Server. Disamping itu, karakteristik pada DOS melalui TCP berbeda dengan kondisi normal. DOS melalui TCP memliki segment PSH-ACK, length dari tiap packetnya 1514, dan waktu antara packet tergolong kecil. Karakteristik pada DOS melalui TCP tetapi menggunakan firewall sama seperti kondisi tanpa serangan, meskipun terdapat eksperimen Serangan DOS melalui TCP. Hal ini dikarenakan firewall memblokir TCP. Log wireshark pada kondisi serangan DOS melalui UDP bisa dilihat lampiran B.. Karakteristik pada serangan DOS melalui UDP mempunyai, lengthnya 60 dan waktu antara packet yang tergolong kecil

.

Tabel 6.1 Trafik Jaringan hasil eksperimen

	Perlakukan	Jumlah Packets(Packets)			Jumlah Bytes(mb)		
		Tanpa serangan	DOS TCP	DOS UDP	Tanpa Serangan	DOS TCP	DOS UDP
Microsoft Windows Server 2016	Tanpa Firewall	1363	5984	83018	0,292	6,193	5,001
	Firewall dan Konfigurasi	1091	9391	127497	0,165	1,703	7,909
Ubuntu	Tanpa Firew	343	8605	82783	0,072	5,502	5,077

Server 16.04	all						
	Firewall dan Konfigurasi	3311	4548	54382	0,387	0,606	3,581

Pada Microsoft Windows Server 2016, saat kondisi tanpa serangan lalu firewall dimatikan jumlah paket dan bytesnya adalah 1363 paket dan 0.291 mb. Sedangkan saat firewall diaktifkan, jumlah paket dan bytesnya adalah 1091 paket dan 0.165 mb. Ketika firewall dimatikan dan dos melalui TCP, jumlah paket dan bytesnya adalah 5984 pakets dan 6,193 mb. Sedangkan saat firewall dinyalakan, jumlah paket dan bytesnya adalah 9391 paket dan 1,703 mb. Dan saat firewall dimatikan lalu dilakukan dos melalui udp, jumlah paket dan bytesnya adalah 83018 paket dan 5,001 mb. Lalu firewall diaktifkan, jumlah paket dan bytesnya adalah 127497 paket dan 7,909 mb.

Pada Ubuntu Server 16.04, pada kondisi tanpa serangan lalu firewall dimatikan jumlah paket dan bytesnya adalah 90733 paket dan 32 mb. Sedangkan saat firewall diaktifkan, jumlah paket dan bytesnya adalah 8385 paket dan 1.061 mb. Ketika firewall dimatikan dan dos melalui TCP, jumlah paket dan bytesnya adalah 8605 pakets dan 5,052 mb. Sedangkan saat firewall dinyalakan, jumlah paket dan bytesnya adalah 4577 paket dan 0,609 mb. Dan saat firewall dimatikan lalu dilakukan dos melalui udp, jumlah paket dan bytesnya adalah 82783 paket dan 5,077 mb. Lalu firewall diaktifkan, jumlah paket dan bytesnya adalah 543282 paket dan 3,581 mb. Sama seperti Microsoft Windows Server 2016, firewall cukup signifikan dalam karena dapat menurunkan jumlah bytes pada dos TCP dari 5,052 mb ke 0,609 mb

Jumlah packet dan bytes tidak berbeda jauh antara Ubuntu dan windows pada saat tanpa serangan. Pada eksperimen DoS melalui TCP, firewall mempunyai perang penting karena dapat menangkal serangan dos,

Pada eksperimen DoS melalui UDP, firewall tidak cukup berpengaruh pada dos melalui UDP. Untuk kasus ini tidak terlalu jauh perbedaan antara Ubuntu dan windows server.

Tabel 6.2 Kinerja Server hasil eksperimen

Perlakuan		Microsoft Windows Server 2016		Ubuntu Server 16.04	
		Tanpa Firewall	Firewall dan Konfigurasi	Tanpa Firewall	Firewall dan Konfigurasi
CPU (%)	Tanpa Serangan	2	2	48.5	48.5
	DOS TCP	6	2	68.9	48.5
	DOS UDP	26	10	90	80
Memory (%)	Tanpa Serangan	18	18	32.4	32.4
	DOS TCP	22	18	35.7	32.4
	DOS UDP	22	27	38.8	45
Aktifitas Jaringan (mbps)	Tanpa Serangan	0.032	0.032	0.715	0.715
	DOS TCP	7.4	0.032	79.8	0.715
	DOS UDP	25.3	2.3	211.8	364.7

Kinerja Windows Server saat dilakukan Dos melalui TCP tanpa firewall adalah CPU dan Memory naik 4% lalu Trafik jaringan naik 7,368 mbps dari keadaan semula. Saat firewall diaktifkan pun kembali ke kinerja semula yaitu CPU 2%, Memory 18% dan 0.032 mbps. Sedangkan kinerja Ubuntu Server saat dilakukan DoS melalui TCP tanpa firewall adalah CPU naik 20.4%, memory naik 0.33% dan trafik jaringan naik 7.265 mbps dari keadaan semula. 3Saat firewall diaktifkan juga kembali seperti semula yaitu CPU 48.5%, memory 32.4%. Kinerja Windows Server saat dilakukan dos melalui UDP tanpa firewall adalah CPU naik 22%, memory naik 4% dan trafik jaringan 2,498 mbps dari keadaan semula. Saat Firewall diaktifkan kinerjanya adalah CPU turun 16%, memory naik 5% dan trafik jaringan turun 22mbps dari kondisi dos melalui udp. Sedangkan kinerja Ubuntu Server saat dilakukan dos melalui UDP adalah CPU naik 88%, memory naik 6.4% dan trafik jaringan naik 211.085 mbps dari kondisi semula. Saat firewall diaktifkan kinerjanya adalah CPU turun 10%, memory naik 6,7% dan trafik jaringan naik 152.9 mbps dari kondisi dos melalui udp

Pada kondisi tanpa serangan, windows tidak terlalu makan resource. Pada dasarnya Ubuntu server lebih ringan dariapda windows server. Dikarenakan pada penelitian ini harus menggunakan GUI maka server harus diinstal Ubuntu-desktop sehingga cukup memakan resource pada proses idle.

Pada eksperimen dos melalui TCP. Windows server jauh lebih unggul daripada Ubuntu server. Firewall pun mempunyai pengaruh pada kasus ini.

Pada eksperimen dos melalui UDP, windows server juga unggul terhadap Ubuntu server. Dilihat dari tabel di atas, penggunaan firewall tidak cukup berpengaruh untuk Ubuntu server, berbanding terbalik dengan windows server.

BAB VII
KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan mengenai kesimpulan dari hasil penelitian pada pengerjaan tugas akhir dan saran perbaikan untuk penelitian selanjutnya.

7.1 Kesimpulan

Berdasarkan hasil penelitian pada analisis forensik jaringan terhadap serangan DoS pada sistem operasi Ubuntu server 16.04 dan Microsoft Windows Server 2016, didapatkan beberapa kesimpulan:

1. Pada Microsoft Windows Server 2016, firewall berpengaruh pada DOS melalui TCP karena dapat menurunkan jumlah bytes sejumlah 4,490 mb, CPU dan memory sejumlah 4%. Sedangkan DOS melalui UDP tidak ada perbedaan signifikan pada jumlah bytes pada penggunaan firewall walaupun menurunkan CPU sejumlah 16% dan menaikkan memory sebanyak 5%.
2. Pada Ubuntu Server 16.04, firewall berpengaruh pada DOS melalui TCP karena dapat menurunkan jumlah bytes sejumlah 4,9 mb, CPU sejumlah 20,4% dan memory sejumlah 3,3%. Sedangkan DOS melalui UDP tidak ada perbedaan signifikan pada jumlah bytes pada penggunaan firewall walaupun menurunkan CPU sejumlah 10% dan menaikkan memory sebanyak 6,2%.
3. Kinerja Microsoft Windows Server 2016 lebih unggul dibandingkan Ubuntu Server 16.04 saat dilakukan eksperimen DoS melalui TCP maupun UDP karena Penggunaan CPU dam Memory yang lebih sedikit. Sehingga direkomendasikan untuk dijadikan sistem operasi pada server.

7.2 Saran

Berdasarkan hasil penelitian pada analisis forensic jaringan terhadap serangan DoS pada sistem operasi Ubuntu Server 16.04 dan Microsoft Windows Server 2016, dibutuhkan penyempurnaan lebih lanjut agar didapatkan hasil yang maksimal. Berikut saran yang dapat disampaikan penulis untuk penelitian selanjutnya adalah sebagai berikut:

1. Penggunaan atau membandingkan dengan metode lain seperti signature-based detection atau stateful/protocol analysis. hal ini berguna untuk mendapatkan hasil yang lebih baik
2. Penggunaan tools dos lain seperti Hoic atau Tor Hammer untuk dijadikan perbandingan dalam penelitian.
3. Penggunaan tools penangkap packet lain seperti Tshark agar analisa dari hasil yang sudah didapatkan dapat dijadikan pertimbangan.
4. Membandingkan dengan sistem operasi lain yang serupa seperti fedora atau red hat agar mengetahui sistem operasi mana yang lebih bagus dan gratis.

DAFTAR PUSTAKA

[1] S. Kemp, "DIGITAL IN SOUTHEAST ASIA IN 2017," 2017. [Online]. Available: https://wearesocial.com/special-reports/digital-southeast-asia-2017.

[2] M. I, "Peranan CERT/CSIRT Untuk melindungi Data Pribadi dan Institusi. Seminar Cyber Defence Teknik Informatika Universitas Jendral Sudirman Purwokerto 21 September 2014," 2014.

[3] B. Stephens, "What Is Digital Forensics?," 2016. [Online]. Available: https://www.interworks.com/blog/bstephens/2016/02/05/what-digital-forensics.

[4] R. U. Putri and J. E. Istiyanto, "Analisis Forensik Jaringan Studi Kasus Serangan SQL Injection pada Server Universitas Gadjah Mada," *Ijccs*, vol. 6, no. 2, pp. 1978–1520, 2012.

[5] Y. S. Nugroho, "Investigasi Forensik Jaringan dari Serangan DDOS menggunakan Metode Naïve Bayes," *Investig. Forensik Jar. Dari Serangan Ddos Menggunakan Metod. Naïve Bayes*, 2015.

[6] B. Ruchandani, M. Kumar, A. Kumar, K. Kumari, A. K. Sinha, and P. Pawar, "Experimentation in network forensics analysis," *Proc. Term Pap. Ser. under CDAC-CNIE Bangalore*, 2006.

[7] L. Volonino and R. Anzaldua, *Computer Forensics For Dummies*. 2007.

[8] A. B. M. Junaidi Syahputra, Ilham Faisal, "Deteksi Serangan Pada Jaringan Komputer Dengan Wireshark Menggunakan Metode Anomaly-Based IDS," *J. Tek. Elektro Terap.*, vol. 1 (2), 2012.

[9] M. I. Cohen, "Network Forensics," pp. 279–306.

[10] Y. T. Aburabie and M. Omari, *Network Forensics Tool "Honeytraps Project."* 2006.

[11] P. A. Networks, "WHAT IS A DENIAL OF SERVICE

ATTACK (DoS)?" [Online]. Available: https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos.

[12] Incapsula, "DISTRIBUTED DENIAL OF SERVICE ATTACKS." [Online]. Available: https://www.incapsula.com/ddos/denial-of-service.html.

[13] C.-M. Cheng, H. T. Kung, and K.-S. Tan, "Use of spectral analysis in defense against DoS attacks," *Glob. Telecommun. Conf. 2002. GLOBECOM '02. IEEE*, vol. 3, pp. 2143--2148 vol.3, 2002.

[14] S. Geges and W. Wibisono, "Pengembangan Pencegahan Serangan Distributed Denial Of Service(DDOS) pada Sumber Daya jaringan dengan Integrasi Network Behavior Analysis dan Client Puzzle," pp. 53–67.

[15] T. Willhalm, R. Dementiev, and P. Fay, "A better way to measure CPU utilization," 2017. [Online]. Available: https://software.intel.com/en-us/articles/intel-performance-counter-monitor#cpu_utilization.

[16] S. Widodo, "Pemantauan Jaringan Komputer dengan DNS Server Berbasis Routing Statis Menggunakan Wireshark," *J. Tek. Elektro*, vol. 1, no. 2, pp. 1–7, 2013.

[17] Wireshark, "7.4. Expert Information." [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChAdvExpert.html.

[18] V. Jyothsna, "A Review of Anomaly based IntrusionDetection Systems," *Int. J. Comput. Appl.*, vol. 28, no. 7, pp. 975–8887, 2011.

[19] J. Mirkovic, A. Hussain, S. Fahmy, P. Reiher, and R. K. Thomas, "Accurately measuring denial of service in simulation and testbed experiments," *IEEE Trans. Dependable Secur. Comput.*, vol. 6, no. 2, pp. 81–95, 2009.

[20] MachMetric, “Average Page Load Times for 2018,” 2018. .

[21] Incapsula, “LOW ORBIT ION CANNON (LOIC).” [Online]. Available: https://www.incapsula.com/ddos/attack-glossary/low-orbit-ion-cannon.html.

[22] M. J. Foley, “Microsoft’s Windows Server 2016 hits general availability,” 2016. [Online]. Available: http://www.zdnet.com/article/microsofts-windows-server-2016-hits-general-availability/.

[23] A. Williams, “The best Linux distros of 2018,” 2018. [Online]. Available: https://www.techradar.com/news/best-linux-distro.

[24] MYDIGITALSHIELD, “Denial of Service – DoS Attack Protection Firewall,” 2014. [Online]. Available: http://www.mydigitalshield.com/denial-of-service-dos-attack-protection-firewall/.

[25] K. Beaver, “Inbound vs. outbound firewall rules: Comparing the differences.” [Online]. Available: https://searchsecurity.techtarget.com/answer/Comparing-firewalls-Differences-between-an-inbound-outbound-firewall.

[26] M. Rouse, “TCP/IP (Transmission Control Protocol/Internet Protocol).” [Online]. Available: https://searchnetworking.techtarget.com/definition/TCP-IP.

[27] M. Rouse, “UDP (User Datagram Protocol).” [Online]. Available: https://searchnetworking.techtarget.com/definition/UDP-User-Datagram-Protocol.

[28] S. Wilkins, “TCP/IP Ports and Protocols,” 2012. [Online]. Available: http://www.pearsonitcertification.com/articles/article.aspx?p=1868080.

BIODATA PENULIS

Penulis lahir di Kota Perdagangan dan Industri, Madiun pada tanggal 9 Juni 1995. Anak kedua dari dua bersaudara yang telah menempuh pendidikan formal yaitu; SD Negeri Klegen 1 Madiun, SMPK Santo Yusuf Madiun, dan SMA Negeri 3 Madiun.

Pada tahun 2014, penulis melanjutkan pendidikan ke jenjang yang lebih tinggi, yaitu di Institut Teknologi Sepuluh Nopember (ITS) Surabaya, sebagai mahasiswa departemen Sistem Informasi, Fakultas Tekonologi Informasi dan Komunikasi (FTIK). Terdaftar sebagai pemilik NRP 05211440000186. Selama menjadi mahasiswa, penulis banyak mengikuti kegiatan kemahasiswaan, antara lain seminar, organisasi dan Pelatihan. Penulis pernah menjadi Staff Kementrian Pengembangan Sumber Daya Mahasiswa (PSDM) di BEM ITS pada tahun 2015/2016. Penulis pernah diberikan tanggung jawab oleh organisasi Himpunan Mahasiswa Sistem Informasi (HMSI) ITS sebagai Kepala Divisi Pemetaan PSDM pada tahun kepengurusan 2016/2017. Selain itu penulis pernah diberikan tanggung jawab sebagai Steering Committee pada Masa Pengembangan Generasi HMSI pada tahun 2015-2016. Disamping aktif dalam kegiatan kemahasiswaan, penulis juga pernah menjadi Asisten Praktikum mata Keamanaan Aset Informasi. Pada tahun ke-4, penulis tertarik dengan bidang Forensika Digital, sehingga mengambil bidang minat laboratorium Infrastruktur dan Keamanaan Teknologi Informasi (IKTI) dan lulus dalam waktu 4 tahun atau 8

semester. Penulis dapat dihubungi melalui email Aloysius.tatus@gmail.com atau whatsapp 085895160039.

LAMPIRAN A – Log Wireshark dari DOS melalui TCP

Berikut merupakan log wireshark saat dilakukan DOS melalui TCP. Log seperti ini akan dijumpai saat DOS melalui TCP tanpa firewall. Apabila menggunaan firewall, tidak akan muncul log seperti ini

Time	Source	Destination	Protocol	Length	Info
2.047.784.105	10.126.12.57	10.126.10.73	TCP	1514	59164 > 80 [PSH, ACK] Seq=43875 Ack=516 Win=65024 Len=1460 [TCP segment of a reassembled PDU]
2.047.818.212	10.126.10.73	10.126.12.57	TCP	54	80 > 59164 [ACK] Seq=516 Ack=45335 Win=122752 Len=0
2.047.823.620	10.126.12.57	10.126.10.73	TCP	1514	59164 > 80 [PSH, ACK] Seq=45335 Ack=516 Win=65024 Len=1460 [TCP segment of a reassembled PDU]
2.047.837.768	10.126.12.57	10.126.10.73	TCP	60	59164 > 80 [PSH, ACK]

					Seq=46795 Ack=516 Win=65024 Len=6 [TCP segment of a reassembled PDU]
2.047.840.397	10.126.12.57	10.126.10.73	TCP	1514	59165 > 80 [PSH, ACK] Seq=46801 Ack=516 Win=65024 Len=1460 [TCP segment of a reassembled PDU]
2.047.848.370	10.126.10.73	10.126.12.57	TCP	54	80 > 59164 [ACK] Seq=516 Ack=46801 Win=125568 Len=0
2.047.851.756	10.126.12.57	10.126.10.73	TCP	1514	59165 > 80 [PSH, ACK] Seq=48261 Ack=516 Win=65024 Len=1460 [TCP segment of a reassembled PDU]
2.047.852.135	10.126.10.73	10.126.12.57	TCP	54	80 > 59165 [ACK] Seq=516 Ack=48261 Win=125568 Len=0

LAMPIRAN B – Log Wireshark dari DOS melalui UDP

Berikut merupakan log wireshark dari DOS melalui UDP. Log seperti ini dapat dijumpai saat melakukan dos melalui UDP dengan firewall maupun tanpa firewall.

Time	Source	Destination	Protocol	Length	Info
6.366.330.065	10.126.12.57	10.126.10.73	UDP	60	52501 > 80 Len=7
6.366.332.855	10.126.12.57	10.126.10.73	UDP	60	52499 > 80 Len=7
6.366.335.424	10.126.12.57	10.126.10.73	UDP	60	52504 > 80 Len=7
6.366.338.274	10.126.12.57	10.126.10.73	UDP	60	52505 > 80 Len=7
6.366.341.439	10.126.12.57	10.126.10.73	UDP	60	52502 > 80 Len=7
6.366.344.600	10.126.12.57	10.126.10.73	UDP	60	52498 > 80 Len=7
6.366.347.473	10.126.12.57	10.126.10.73	UDP	60	52500 > 80 Len=7
6.366.350.450	10.126.12.57	10.126.10.73	UDP	60	52503 > 80 Len=7
6.376.994.143	10.126.12.57	10.126.10.73	UDP	60	52501 > 80 Len=7
6.377.054.916	10.126.12.57	10.126.10.73	UDP	60	52505 > 80 Len=7
6.377.058.970	10.126.12.57	10.126.10.73	UDP	60	52499 > 80 Len=7
6.377.062.360	10.126.12.57	10.126.10.73	UDP	60	52504 > 80 Len=7

6.377.065.390	10.126.12.57	10.126.10.73	UDP	60	52498 > 80 Len=7
6.377.068.709	10.126.12.57	10.126.10.73	UDP	60	52500 > 80 Len=7
6.377.071.682	10.126.12.57	10.126.10.73	UDP	60	52502 > 80 Len=7
6.377.074.911	10.126.12.57	10.126.10.73	UDP	60	52503 > 80 Len=7
6.377.078.005	10.126.12.57	10.126.10.73	UDP	60	52501 > 80 Len=7