Forensik Jaringan untuk Deteksi Serangan *Flooding* pada Web Server

DESTI MUALFAH

14917114

Tesis diajukan sebagai syarat untuk meraih gelar Magister Komputer

Konsentrasi Forensik Digital

Program Studi Magister Teknik Informatika

Program PascasarjanaFakultas Teknologi Industri

Universitas Islam Indonesia

2017

Lembar Pengesahan Pembimbing

FORENSIK JARINGAN UNTUK DETEKSI SERANGAN *FLOODING*
PADA WEB SERVER

Nama: DESTI MUALFAH
NIM: 14917114

Yogyakarta, Januari 2017

Pembimbing I

Dr. Imam Riadi, M.Kom

Lembar Pengesahan Penguji

FORENSIK JARINGAN UNTUK DETEKSI SERANGAN *FLOODING*
PADA WEB SERVER

Nama: DESTI MUALFAH

NIM: 14917114

Yogyakarta, Januari 2017

Tim Penguji,

Dr. Imam Riadi, M.Kom
Ketua

Dr. Bambang Sugiantoro
Anggota I

Yudi Prayudi, S.Si., M.Kom
Anggota II

Mengetahui,
Direktur Program Pascasarjana Fakultas Teknologi Industri
Universitas Islam Indonesia

Dr. R. Teduh Dirgahayu, ST., M.Sc

Pernyataan keaslian tulisan

Dengan ini saya menyatakan bahwa tesis ini merupakan tulisan asli dari penulis, dan tidak berisi material yang telah diterbitkan sebelumnya atau tulisan dari penulis lain terkecuali referensi atas material tersebut telah disebutkan dalam tesis. Apabila ada kontribusi dari penulis lain dalam tesis ini, maka penulis lain tersebut secara eksplisit telah disebutkan dalam tesis ini.

Dengan ini saya juga menyatakan bahwa segala kontribusi dari pihak lain terhadap tesis ini, termasuk bantuan analisis statistik, desain survei, analisis data, prosedur teknis yang bersifat signifikan, dan segala bentuk aktivitas penelitian yang dipergunakan atau dilaporkan dalam tesis ini telah secara eksplisit disebutkan dalam tesis ini.

Segala bentuk hak cipta yang terdapat dalam material dokumen tesis ini berada dalam kepemilikan pemilik hak cipta masing-masing. Untuk material yang membutuhkan izin, saya juga telah mendapatkan izin dari pemilik hak cipta untuk menggunakan material tersebut dalam tesis ini.

Yogyakarta, Januari 2017

Desti Mualfah

vi

Publikasi selama masa studi

Mualfah, D., Riadi, I., (2017). Network Forensics for Detecting Flooding Attack on Web Server, vol.15 No. 2., pp. 326-331.

Publikasi yang menjadi bagian dari tesis

Publikasi berikut menjadi bagian dari tesis

Kontributor	**Jenis Kontribusi**
Desti Mualfah	Mendesain Eksperimen (80%) Menulis *paper* (80%)
Imam Riadi	Mendesain Eksperiment (20%) Menulis dan mengedit *paper* (20%)

Kontribusi yang diberikan oleh pihak lain dalam tesis ini

Dr. Imam Riadi, M.Kom : Solusi tool yang digunakan berupa Snort.

Desti Mualfah : Deteksi serangan yang diterapkan untuk lingkungan Universitas Muhammadiyah Magelang.

Dr. Bambang Sugiantoro : Lampiran *rule flooding attack.*

Yudi Prayudi, S.Si., M.Kom : Struktur laporan tesis.

Halaman Persembahan

Bapak/Ibu

Terima kasih tak terhingga untuk doa tulus kepada anakmu ini seperti air yang terus mengalir. Terimakasih atas pengorbanan dan motivasi yang selalu diberikan selama ini.

Mas/Mbak

Terima kasih atas doa dan dukungan yang telah kalian berikan. Semoga kita bisa meraih sukses bersama-sama.

Ramadhan

Takdir adalah garis tangan yang meletakkanku di sebuah jalan. Kemanapun aku melangkah, kau ada di setiap ujung persimpangan. Yakinlah bahwa doa yang kita istiqomahkan akan sampai pada sebuah jawaban.

CLYR Activity

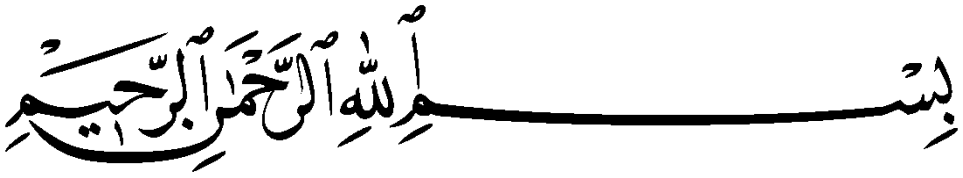
Terima kasih atas hari-hari dimana kita sering berkumpul berbagi suka dan duka. Diana, Alponk, Mas Gede, Abhy, Danang, Alabi, Bang Iwan, Kak Depong, Kakcret Mega dan Mas Afif yang selalu stay kalau di gangguin.

KEHIDUPAN Group

Thanx's A lot buat Kehidupan Group Bang Rizdqi, Mas Adam, Mas Djul, Mas Itqan dan Mas Asep yang selalu membantu dalam berdiskusi laporan tesis ini.

Akhir kata, saya persembahkan tesis ini kepada semua pembaca. Semoga tesis ini dapat bermanfaat, inspirasi, dan tambahan ilmu.

Kata Pengantar



Assalamu'alaikum Wr. Wb.

Alhamdulillah segala puji bagi Allah SWT atas segala rahmat, hidayah, dan kehadirat-Nya, sehingga penulisan laporan tesis sebagai salah satu syarat memperoleh gelar Pascasarjana Magister Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia yang berjudul "FORENSIK JARINGAN UNTUK DETEKSI SERANGAN *FLOODING* PADA WEB SERVER" dapat diselesaikan dengan baik. Shalawat serta salam semoga senantiasa tercurah atas Nabi Muhammad SAW, para sahabat, serta pengikutnya.

Penyusunan tesis ini tidak lepas dari bimbingan, dukungan, dan bantuan dari berbagai pihak. Oleh Karena itu dalam kesempatan ini dan segala kerendahan hati, ucapan terima kasih diucapkan dengan setulus-tulusnya kepada:

1. Allah SWT, yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis selalu diberkan kesehatan dan kemudahan selama masa pengerjaan tesis ini.
2. Bapak, ibu, kakak, beserta keluarga besar yang telah mendoakan dan memberikan restu dan semangatnya.
3. Bapak Rektor dan seluruh jajaran rektorat Universitas Islam Indonesia.
4. Dr. R. Teduh Dirgahayu, ST., M.Sc selaku direktur Program Pascasarjana Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Dr. Imam Riadi, M.Kom selaku dosen pembimbing yang telah memberikan pengarahan, bimbingan, masukan, serta dorongan semangat selama pengerjaan tesis ini.
6. Dosen-dosen Magister Teknik Informatika dan seluruh jajaran staf program Pascasarjana. Terima kasih atas semua ilmu pengetahuan, saran, motivasi, serta bantuannya.
7. Rekan-rekan Forensik Digital UII, terima kasih atas semua dukungan dan kerja samanya selama ini.

8. Keluarga besar Magister Teknik Informatika UII.
9. Sahabat-sahabat yang jauh disana dan selalu mendoakan, terima kasih.
10. Semua pihak yang telah memberikan bantuan dan dorongan yang tidak dapat disebutkan satu-persatu.

Saya menyadari bahwa dalam penulisan dan penyusunan laporan tesis ini masih banyak terdapat kekurangan. Untuk itu saya sampaikan permohonan maaf serta sangat mengharapkan kritik dan saran yang membangun untuk penyempurnaan di masa yang akan datang.

Yogyakarta, 2017

Desti Mualfah

Abstrak

Keamanan jaringan komputer menjadi bagian terpenting untuk menjamin integritas dan validitas layanan bagi pengguna. Suatu serangan ke dalam web server komputer dapat terjadi kapan saja. Salah satunya serangan *Flooding* yang merupakan ancaman serius keamanan jaringan pada web server dapat mengakibatkan kerugian *bandwidth* dan akses web lambat, baik bagi pengguna maupun penyedia layanan web server. Langkah awal untuk meminimalisir terjadinya serangan flooding adalah kemampuan untuk mendeteksi serangan dengan menggunakan *Intrusion Detection System* (IDS). Snort merupakan salah satu *tool* yang dapat digunakan untuk mendeteksi serangan *flooding*.

Snort memiliki kemampuan untuk mendeteksi serangan *flooding* secara *real time* dengan menerapakan *rule* khusus untuk menghasilkan suatu file log yang mencatat aktivitas yang dianggap berbahya. *File log* yang merupakan barisan data untuk menyimpan informasi mengenai segala tindakan, kejadian dan aktifitas yang terjadi di dalam sebuah sistem jaringan. Selanjutnya digunakan untuk proses investigasi analisis forensic jaringan (*network forensic*) yang merupakan ilmu keamanan komputer berkaitan dengan tahap-tahap untuk menemukan sumber serangan. Investigasi yang digunakan berupa model proses forensik. Terdiri dari tahap pengkoleksian, pemeriksaan, analisis dan pelaporan untuk mendapatkan bukti-bukti serangan yang bersumber dari file log.

Hasil penelitian yang telah dilakukan pemasangan *Intrusion Detection System* (IDS) Snort mampu mendeteksi serangan *flooding*. Sejumlah 15 IP *address* yang melakukan tindakan illegal ke dalam *web server*. Hasil analisis penelitian *flooding* dapat menemukan barang bukti investigasi menggunakan *Intrusion Detection System* (IDS) Snort. Berdasarkan hasil pengujian tersebut dapat dinyatakan hasil sudah sesuai dengan tujuan yang diharapkan, sehingga dapat disimpulkan penelitian ini berhasil berjalan dengan baik dan lancar.

Kata Kunci

serangan *flooding*, *intrusion detection system* (IDS), snort, *network forensic*.

Abstract

Computer Network Security Become The most important part to ensure the integrity and validity of the review SERVICE For users. To attack a web server hearts Computers can be Happen Anytime. One of Those Flood Attacks Which is a serious threat ON Network Security web server bandwidth can be resulted Losses And web Access Slow, both users and providers of those web server program service. Initial steps for the review minimize flood attack is the ability to detect attacks BY review using Intrusion Detection System (IDS). Snort is a prayer One That tool can be used to detect review flood attacks.

Snort has the ability to detect flooding attacks in real time by applying a special rule to generate a log file that records the activities that are considered berbahya. The log file which is a sequence of data to store information about all actions, events and activities that occur in a network system. Further investigation process used for forensic analysis of network (network forensic) which is the science of computer security with regard to the steps to find the source of the attack. Investigations are used in the form of the forensic process model. Comprising the step of collecting, examination, analysis and reporting to obtain evidence of attacks originating from the log files.

Results of research conducted installation of Intrusion Detection System (IDS) Snort is able to detect flooding attacks. Some 15 IP addresses that perform illegal actions to the web server. Results of analysis of flooding can find evidence in the investigation using Intrusion Detection System (IDS) Snort. Based on these test results can be declared results are in accordance with the expected goals, so that we can conclude this study managed to run well and smoothly.

Keywords

flooding attacks, intrusion detection system (IDS), snort, forensic network.

Daftar Isi

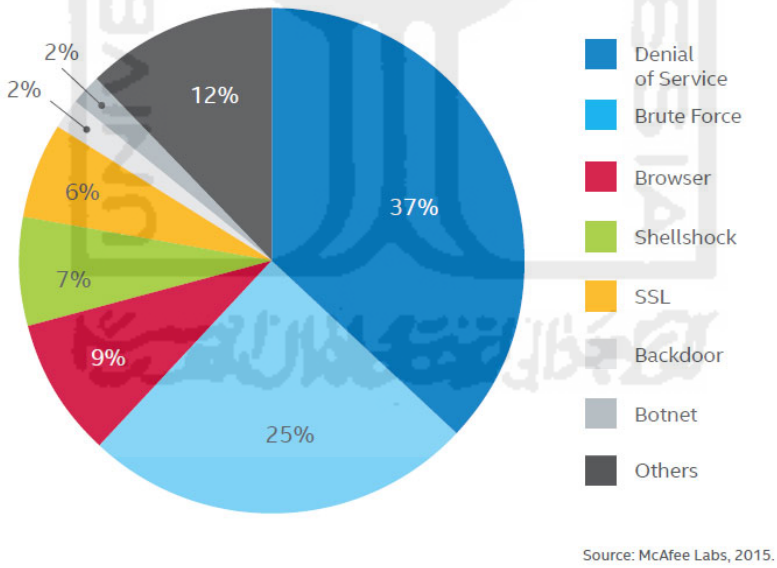
Daftar Gambar

Daftar Tabel

Bab 1 Pendahuluan

1.1 Latar Belakang

Adanya jaringan komputer merupakan salah satu terobosan bagi dunia komputer yang disertai dengan ancaman dan *hacking* jaringan, yang terkait dengan sistem informasi sampai penyalahgunaan data dan informasi. Peningkatan ancaman dan serangan pada keamanan jaringan komputer pada web server dunia seperti terlihat pada Gambar 1.1 dalam laporan (Nguyen, Tran, Ma, & Sharma, 2014) yang masuk di *file log* terlihat paling umum saat ini adalah *Browser, Brute Force, DDoS (Distributed Denial of Service), SSL, Scan, DNS, dan Backdoor*. ("McAfee Labs Threats Report," 2016).



Gambar 1. 1 Top Network Attack

Sumber: Labolatori McAfee ("McAfee Labs Threats Report," 2016)

Serangan web server pada gambar 1.1 membuktikan bahwa semakin banyaknya terjadi kasus *hacking* pada bidang *web server*, serangan yang dapat mengakibatkan akses web lambat,

flooding data, bahkan pencurian informasi dan data melalui jaringan (internet) membuat peretas memiliki banyak waktu untuk melakukan serangan terhadap target serangan (Cahyanto & Prayudi, 2014). Hal tersebut merupakan dampak negatif yang diperoleh dari berkembangnya teknologi jaringan komputer. Berkaitan dengan hal tersebut, muncul suatu bidang teknologi dan komputer yang relatif berkembang pada saat ini, yaitu *network forensic* (Nguyen et al., 2014). *Network forensic* (forensik jaringan) merupakan cabang *digital forensics* yang menggunakan teknik secara ilmiah yang terbukti untuk mengumpulkan, menggunakan, mengidentifikasi, menguji, menganalisis, mendokumentasi ulang dan dapat mempresentasikan barang bukti digital dari beberapa sumber bukti digital dalam memproses dan mengirimkannya dimana bukti ditangkap dari jaringan dan dipresentasikan berdasarkan pengetahuan dari serangan yang di dapat dari *file log* yang berasal dari komputer (forensik komputer) (palmer, 2001).

File log merupakan mekanisme pencatatan yang dilakukan pada sebuah *web server* dengan menyimpan data setiap pengunjung yang mengirimkan permintaan ke *web server* ke dalam suatu file yang dinamakan *file log web server* (Iswardani & Riadi, 2016). Data pengunjung yang terdapat pada *web server log* akan sangat bermanfaat apabila nantinya terdapat suatu permasalahan yang terjadi terhadap *web server*, khususnya apabila terjadi serangan. *Web server* yang sering terindikasi serangan memiliki dampak yang serius maka dalam *web server* harus menerapkan sebuah keamanan jaringan berupa *firewall* yang digunakan untuk mengarahkan paket data yang tidak dikehendaki. Akan tetapi data yang masuk ke dalam *log firewall* mempunyai karakteristik yang tidak dapat dibaca secara langsung, sehingga menyulitkan seorang admin dalam membaca *log file* yang masuk.

Berdasarkan permasalahan tersebut, ada beberapa konsep yang digunakan untuk keamanan jaringan adalah NIDS (*Network Intrusion Detection System*) yang berdasarkan *anomaly* jaringan, HIDS (*Host Intrusion Detection System*) yang berdasarkan *anomaly host* dan khusus untuk forensik digunakan konsep NFAT (*Network Forensik Analysis Tools*). NFAT adalah alat untuk menangkap lalu lintas jaringan, menganalisis lalu lintas jaringan sesuai dengan kebutuhan pengguna, dan memungkinkan pengguna sistem untuk menemukan hal-hal yang berguna dan menarik tentang lalu lintas yang dianalisis (Sindhu & Meshram, 2012)

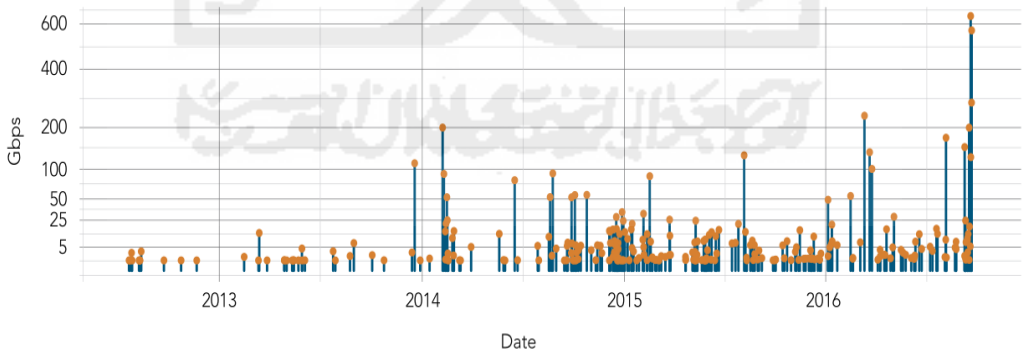
Dalam implementasi NFAT (*Network Forensic Analysis Tool*) bukti pengintaian yang akan dilakukan adalah dengan cara memeriksa protokol TCP (*Transmission Control Protocol*), UDP (*User Datagram Protocol*), IP (*Internet Protocol*), yang melewati trafik jaringan. Sedangkan untuk bukti serangan dan pelanggaran di jaringan akan diketahui jika ada aktivitas yang tidak biasa seperti dalam hal komunikasi jaringan, protokol dan *port*, koneksi ke dalam,

koneksi ke luar, gagal koneksi, dan trafik *pee-to-peer*. (“Guide to Integrating Forensic Techniques into Incident Response,” n.d.)

Untuk memerangi kejahatan pada *web server* digunakan perangkat lunak yang dapat digunakan untuk melakukan identifikasi pelaku terkait dengan serangan web server diantaranya IDS (*Intruction Detection System*) (Stiawan et al., 2012) salah satunya Snort (“Introduction to Snort A . Sniffer Mode,” n.d.). Snort dapat melakukan deteksi adanya penyusupan terhadap *web server* dengan cara menganalisis data log. IDS digunakan sebagai salah satu solusi yang dapat digunakan untuk membantu dan menganalisa paket-paket yang berbahaya.

Pada penelitian forensik jaringan pada web server ini dihususkan untuk mendeteksi serangan *flooding* yang dengan sengaja penyerang mengirimkan serangan untuk mengacaukan atau menghentikan sebuah layanan. Serangan *flooding* berupa serangan DoS (*Denial of Service*) dan DDoS (*Distributed Desnial of Service*) terhadap sebuah server di dalam jaringan (“Design & Deployment Of Testbed Based On ICMPv6 Flooding Attack,” 2014). Namun sayangnya saat ini Snort belum tentu terpasang dalam suatu web server, padahal untuk melakukan konfigurasi Snort bersama *firewall* dapat berkontribusi menjadi solusi permasalahan dalam mendeteksi dan membaca *log* serangan *web server*.

Untuk melakukan konfigurasi Snort tersebut akan diterapkan dalam lingkungan Universitas Muhammadiyah Magelang. Yang mana serangan *flooding* terjadi semakin meningkat cukup signifikan dari tahun ke tahun. Peningkatan statistik serangan *flooding* dapat dilihat pada gambar 1.2.



Gambar 1. 2 Statistik Flooding Attack

Gambar 1.2 menjelaskan statistik grafik peningkatan signifikan antara tahun 2013 hingga 2016. Untuk itu peneliti akan melakukan penelitian yang berfokus pada deteksi serangan fooding, kemudian analisis forensik akan digunakan untuk membantu menganalisis hasil dari deteksi

serangan flooding pada web server pada jaringan komputer di biro Teknologi Informasi dan Komunikasi (TIK) yang merupakan pusat jaringan di Universitas Muhammadiyah Magelang. Maka dari itu diperlukan mekanisme untuk mengkonfigurasikan Snort dalam mendeteksi serangan dan investigasi penyelidikan kasus-kasus yang sering terjadi berdasarkan semua *file log flooding* yang diambil selama penelitian berlangsung.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah:

a. Apakah pemasangan Snort mampu memberikan informasi dalam mendeteksi serangan *flooding*?
b. Bagaimana hasil analisis *file log* Snort dalam menemukan barang bukti digital forensik?

1.3 Batasan Masalah

Didalam melaksanakan kegiatan penelitian ini ada beberapa batasan masalah, yaitu:

a. Uji coba dilakukan dalam lingkungan Universitas Muhammadiyah Magelang.
b. Simulasi serangan diambil dari rekaman Snort yang telah terpasang.
c. Dalam mendapatkan log file serangan ini di ambil di ruang lab biro TIK Universitas Muhammadiyah Magelang.
d. Mendeteksi kinerja serangan *web server* menggunakan Snort.
e. Bukti digital forensik ditinjau dari file log Snort selama simulasi berlangsung.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dibuat maka dapat iambil tujuan penelitian ini sebagai berikut:

a. Pemasangan Snort untuk memberikan informasi dalam mendeteksi serangan *flooding*.
b. Menganalisis *file log* Snort untuk mendapatkan barang bukti digital forensik.

1.5 Manfaat Penelitian

Manfaat yang dapat diperoleh dari hasil penelitian adalah sebagai berikut:

a. Pemasangan Snort untuk memberikan informasi dalam mendeteksi serangan.

b. Mengetahui bukti digital forensik dari *file log* Snort.

1.6 Review Penelitian

Resi Utami Putri (Putri, R U. 2012) forensik jaringan merupakan ilmu keamanan komputer yang berkaitan dengan investigasi untuk menemukan sumber serangan pada jaringan berdasarkan bukti log, mengidentifikasi, menganalisa serta merekontruksi ulang kejadian tersebut. Metode yang digunakan adalah model proses forensik *(The Forensic Proces Model)* yang terdiri dari tahap pengoleksian, pemeriksaan analisis dan pelaporan. Penelitian ini telah mendapatkan 68 IP *Address* yang melakukan tindakan illegal SQL Injection pada server www.ugm.ac.id. Dan kebanyakan penyerang menggunakan tools SQL *injection* yaitu Havij dan SQLMap.

Ismi Junita Rahmawati (Rahmawati, IJ. 2012) *Intruction Berbasis System* (IDS) berbasis jaringan NIDS untuk layananan *Infrastrukture as aService* (IaaS) yang diimplementasikan pada *open cloud computing*. Dengan menggunakan mirroring traffic pada switch, traffic akan diarahkan *traffic* akan diarahkan ke *NIDS* sehingga *NIDS* mampu memantau semua traffic jaringan yang berasal dari luar *server cloud* maupun *traffic* yang antar *virtual machine* di dalam *server cloud*. Tugas utamanya adalah memantau aktivitas yang mencurigakan dari luar *cloud computing* dan antar *host* di dalam *cloud computing* dan memberikan laporan ke administrator jaringan jika ada serangan yang terjadi di lingkungan sistem (Junita Rahmawati, 2012).

Penelitian yang dilakukan Ira Vaoliya Shafitri (Shafitri , I. 2012) Serangan Pada Keamanan Sistem Jaringan Komputer Melalui *Email* , menyatakan bahwa Potensi serangan dapat memyebabkan ancaman yaitu adanya akses tidak terotorisasi pada informasi data dimana hal ini terjadi kelemahan sistem keamanan jaringan komputer dan adanya kesalahan sistem atau kerusakan sistem komputer karena adanya serangan dari *Hacker*. Oleh karena itu dibutuhkan sebuah sistem yang bisa mendeteksi adanya serangan secara *realtime respons*, Intrusion *Detection System (IDS)* adalah sebuah aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS dapat melakukan inspeksi terhadap lalu lintas inbound dan outbound dalam sebuah sistem atau jaringan, melakukan analisis dan mencari bukti dari percobaan intrusi (penyusupan).

(Lipeng, Xingyuan, Huilin, & Wang, 2013) teknologi IDS bertujuan untuk mengidentifikasi intruksi ilegal yang tersembunyi pada jaringan lalu lintas yang diserang, penelitian ini berkomitmen untuk menyediakan sutu metode generasi yang sistematis dan dan ilmiah untuk menghindari berbagai bentuk serangan dengan menggunakan kerangka kerja dan rekomendasi untuk pertahanan dari serangan yang masuk.

Adi Cahyanto Log merupakan sebuah file yang berisi data atau informasi mengenai daftar tindakan, kejadian dan aktifitas yang telah terjadi di dalam suatu sistem. Dari beberapa log yang ada data log tersebut belum tentu sesuai dengan yang diinginkan dan dicari, maka pada saat ini tersedia beberapa aplikasi perangakat lunak salah satunya IDS atau program signature lain untuk dapat melacak keberadaan pelaku yang menggunakan *IP Address log* yang sudah tersimpan untuk menemukan bukti digital dan penelitian ini juga menggunakan DNS *(Domain Name System) blacklist* dan informasi GeoIP untuk mengidentifikasi identitas penyerang yang potensial (Cahyanto & Prayudi, 2014).

Suteva, Natasa Mileva, Aleksandra, Loleski, Mario (Suteva, Mileva, & Loleski, 2014) menggunakan post-mortem komputer analisis forensik baik dari penyerang dan mesin korban, menemukan beberapa artefak. Skenario dari tiga jenis serangan: SQL Injection, XSS, dan inklusi file jarak jauh dengan injeksi byte null. Penyerang yang menggunaan shells untuk serangan, meningalkan bukti pada kedua mesin. Pada mesin penyerang, jejak yang ditemukan di file sejarah browser, browser sementara penyimpanan, dan berkas bash_history. Pada mesin korban, jejak ditemukan dalam sistem file dan file log. Ini artefak dapat membantu untuk mengidentifikasi dan kadang-kadang untuk merekonstruksi serangan, dan bahkan lebih valid untuk bukti pengadilan.

Nattawat Khamphakdee, Nunnapus Benjamas, Saiyan Saiyod. (Khamphakdee, N. Dkk. 2014 dalam keamanan sebuah data dalam suatu organisasi berperan sangat penting yang harus di lidungi untuk mrngurangi resiko dari berbagai macam serangan. Snort *Intrucsion Detection System* adalah salah satu alat keamanan jaringan yang telah banak digunakan untuk mencocokkan lalu lintas paket data dalam penyelidikan berbagai serangan yang masuk. Snort jugan digunakan untuk membandingkan efektifitas serangan yang masuk agar mendapatkan hasil akurasi dengan membandingkan deteksi *scoring* untuk medeteksi waktu. (Khamphakdee, 2014).

Rami Al-Dalky (Al-Dalky, R. 2014) dalam penelitian NIDS dalam beberapa tahun terahir memeliki tingkat kecepatan pemrosesan paket dan *real time* mendeteksi lalu lintas yang berbahaya. Snort yang digunakan sebagai aplikasi threaded satu-satunya pengolahan dalam mendeteksi kemampuan lalu lintas yang masuk. Dalam desain aturan snort diturunkan ke dalam

lapisan hardware berbasis NetFPGA. Dan digunakan brdasarkan Bloom menyaring dalam menganalisis dan menyaring paket ang datang dengan *field header* dengan aturan yang digunakan.

Yogi Surya Nugrogo (Nugroho, Y S. 2015) dalam penelitian analisis dimana tujuannya untuk menginvesigasi dan menganalisis serangan DDOS dengan menggunakan metode Naive Bayes dengan cara mengumpulkan semua file log dan mengklarifikasikan waktu serangan. Digunakan metode Naive Bayes, dalam penelitian ini menunjukkan bahwa:

1. Penelitian ini telah mengklarifikasikan kecepatan dari serangan DDOS baik menggunakan TCP maupun UDP.
2. Penelitian ini dapat menyimpulkan bahwa serangan DDOS menggunakan mengguakan TCP maupun UDP dapat membuat kinerja server lebih berat karena server dikirimkan paket berulang kali.

Penelitian ini dapat menyimpulkan bahwa agar CPU dapat menyimpan traffic log ang akan digunakan sebagai barang bukti maka harus memiliki hardisk yang besar, karena file log yang tersimpa sangatlah besar. (Studi, Informatika, Sains, Teknologi, & Kalijaga, 2015)

Penelitian yang dilakukan Mr. Vrushank Shah, Dr. A.K Aggarwal (Shah, V., Aggarwal. 2015) bahawa serangan DOS adalah situasi penyerang dimana penyerang mencoba untuk mencegah penggunaan dari layanan tertentu untuk merusak layanan target. IDS sistem detksi yang lebih efisien dibandingkan dengan firewall dalam mendeteksi serangan DOS karena lalu lintas internal, namun sistem IDS tunggal terkadang agal dalam mendeteksi serangan baru dan memberikan peringatan palsu yang lebih besar. Penelitian ini menggunakan metode heterogen untun mendeteksi serangan DOS karena lebih efisien dalam mendeteksi alert ang masuk ke dalam sistem IDS. (Shah & Aggarwal, 2015). Agar lebih jelas maka literatur riview dijelaskan pada tabel dibawah ini:

Tabel 1. 1 Tabel Literatur Review

No.	Paper Utama	Metode dan Pendeteksian	Penelitian Serangan Tertinggi		Metode Investigasi	Kesimpulan
			Brute Force	*Flooding Attack*		
1	Resi Utami Putri (2012)	Log untuk mengidentifikasi serangan SQL Injection	–	–	Model Proses Forensik	Mengetahui serangan SQL Injection
2	Ismi Junita Rahmawati (2012)	mirroring traffic pada switch	–	–	–	Memantau aktifitas di luar cloud Computing
3	Ira Vaoliya Shafitri (2012)	NIDS (Network Intrution Detection System)	–	–	–	IDS dapat melakukan inspeksi terhadap lalu lintas inbound dan outbound dalam sebuah sistem atau jaringan
4	Lipeng, Xingyuan, Huilin, & Wang (2013)	Kerangka kerja, dan rekomendasi	√	–	–	menormalkan lalu lintas jaringan dan mendeteksi lalu lintas yang abnormal dengan bebagai teknik serangan
5	Adi Cahyanto (2013)	Mengambil file log	–	–	Hidden Marcov	algorima *forward-backward*, dan algoritma *baum-welch*

Tabel 1. 2 Cont'd Litelatur Review

No.	Paper Utama	Metode dan Pendeteksian	Penelitian Serangan Tertinggi		Metode Investigasi	Kesimpulan
			Brute Force	*Flooding Attack*		
6	Suteva, Natasa Mileva, Aleksandra, Loleski, Mario (2014)	–	–	–	Post-mortem Komputer	Serangan SQL Injection XXS meninggalkan bukti di mesin penyerang dan korban
7	Nattawat Khamphakdee, Nunnapus Benjamas, Saiyan Saiyod (2014)	*Rules Of Network Probe Attack Detection*	–	–	–	Mengelompokkan hasil klarifikasi serangan yang masuk ke dalam lalu lintas web server
8	Rami Al-Dalky (2014)	*NetFPGA-based Bloom Filter*	√	–	–	Penggunaan CPU dalam paket loss saat menggunakan Snort NetFPGA
9	Mr. Vrushank Shah, Dr. A.K Aggarwal (2015)	*IDS alerts for Detecting DOS Attacks*	–	√	–	(DOS, -DOS,) apabila prosentasi $^{\delta}$ DOS atau –DOS
10	Yogi Surya Nugrogo (2015	Mendeteksi DDoS	–	√	Naïve Bayes	Kecepatan dalam menangkal serangan DDoS

Tabel 1. 3 Con't Litelatur Review

No.	Paper Utama	Metode dan Pendeteksian	Penelitian Serangan Tertinggi		Metode Investigasi	Kesimpulan
			Brute Force	*Flooding Attack*		
Usulan Penelitian		NIDS (Network Intrution Detection System) File Log dari snort	√	√	Model Proses Forensik	Menganalisis karakteristik file log NIDS jenis serangan yang dihasilkan dari Snort, Pengujian pemasangan snort dapat memberikan informasi kepada administrator dalam membantu mendeteksi serangan/penyusupan dalam mendapatkan bukti digital forensik.
		Untuk mengurangi resiko serangan, maka diperlukan mekanisme pengaman dalam mengurangi tingkat kerentanan terhadap sebuah sistem *intruction detection* (IDS). IDS ini menggunakan aplikasi *snort* berbasis *open source* yang dapat memberikan *file output* dari *snort* berupa *log Network Intruction Detection System* hususnya Flooding Attack yang mempunyai tingkat serangan tertinggi, maka Snort dapat membantu seorang admin dalam mengimprementasikan *file text* serangan yang sulit dipahami oleh masyarakat awam dan membantu menemukan bukti-bukti digital serangan yang menuju ke server Maka model proses forensik akan membantu dalam memvisualisasikan *file log* serangan agar dapat dipahami oleh masyarakat awam				

1.7 Metode Penelitian

Susunan laporan penelitian ini perlu metodologi penyelesaian secara sistematis, peneltian ini menggunakan beberapa tahap berupa:

Gambar 1. 3 Metodologi Penelitian

1. *Literatur Review*

   Studi litelatur dilakukan untuk mendapatkan informasi mengenai topic penelitian yang dapat bersumber dari dokumen, buku, artikel, atau bahan tertulis lainya yang berupa teori, laporan penelitian, atau penemuan sebelumnya, baik bersifat *online* maupun *offline source*.
2. Identifikasi sistem

   Merupakan tahap perancangan dan implementasi snort yang akan digunakan sebagai objek penelitian.
3. Konfigurasi Snort

   Tahap konfigurasi snort dimulai dari instalasi snort kemudian melakukan konfigurasi snort.
4. Simulasi Kasus

   Merupakan tahap dilakukannya simulasi kasus penggunaan snort dalam mendeteksi adanya penyusupan atau serangan. Simulasi kasus bertujuan untuk melakukan pengujian snort seberapa besar manfaat snort untuk memberikan informasi kepada administrator dalam mendeteksi penyusup atau serangan.
5. Analisis

   Tahap ini dilakukan untuk melakukan investigasi dalam menemukan bukti serangan, mengelompokkan jenis serangan apa yang terjadi, IP siapa yang melakukan serangan, kapan serangan itu terjadi, dimana serangan itu terjadi, bagaimana serangan tersebut bisa terjadi, dan mengapa itu terjadi.
6. Laporan

   Tahap ini dilakukan untuk mereport semua data yang telah di analisis yang digunakan sebagai bukti digital yang sah dan dapat diterima secara umum.

1.8 Sistematika Penulisan

Untuk mempermudah proses pembahasan dalam penelitian, maka dibuat sistematika penulisan pada penelitian ini:

BAB I PENDAHULUAN

Pendahuluan, merupakan pengantar terhadap permasalahan yang akan dibahas. Didalamnya menguraikan tentang gambaran suatu penelitian yang terdiri dari latar belakang, rumusan masalah, batasan masalah, manfaat penelitian, tujuan penelitian, metodologi penelitian, serta sistematika penulisan.

BAB II LANDASAN TEORI

Pada Bab ini menjelaskan teori-teori yang digunakan untuk memecahkan masalah dalam penelitian ini. Teori yang dibahas pada bagian ini merupakan teori yang berhubungan dengan *web foensics*

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang langlah-langkah penelitian dan gambaran umum langkah penyelesaian.

BAB IV PEMBAHASAN

Hasil dan Pembahasan, berisi tentang pembahasan penyelesaian masalah yang diangkat yaitu dengan melakukan analisis dan uji coba.

BAB V KESIMPULAN DAN SARAN

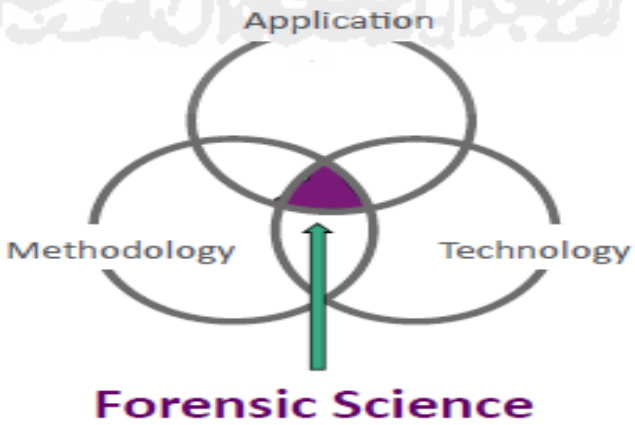
Kesimpulan dan Saran, memuat kesimpulan-kesimpulan dari hasil penelitian dan saran-saran yang perlu diperhatikan berdasar keterbatasan yang ditemukan serta asumsi-asumsi yang dibuat selama melakukan penelitian dan juga rekomendasi yang dibuat untuk pengembangan penelitian selanjutnya.

Bab 2 Landasan Teori

2.1 Forensik dan Forensik Jaringan

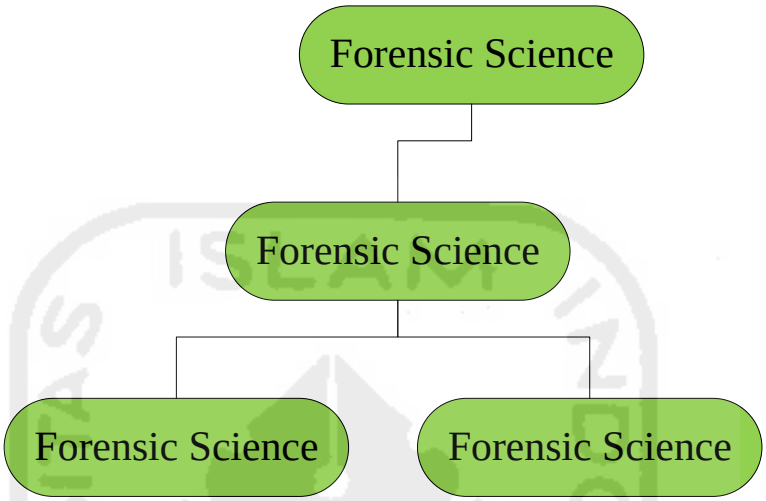
Forensik adalah suatu proses ilmiah atau suatu usaha ilmiah yang didasari ilmu pengetahuan dalam mengumpulkan, mengnalisa dan menghadirkan bukti dalam suatu persidangan di pengadilan untuk membantu pengungkapan suatu kejahatan melalui pengungkapan bukti-bukti yang sah menurut undangn-undang dan peraturan yang berlaku. Digital forensik merupakan aplikasi bidang ilmu pengetahuan dan teknologi komputer untuk kepentingan pembuktian hukum (pro justice), yang dalam hal ini adalah untuk membuktikan kejahatan berteknologi tinggi atau computer crime secara ilmiah (scientific) sehingga bisa mendapatkan bukti-bukti digital yang dapat digunakan untuk menjerat pelaku kejahatan tersebut.

Menurut Franke, 2010 (Franke, n.d.) yang terlihat dari Gambar 2.1 menunjukan bahwa metode forensik dari pendekatan untuk melakukan tugas seperti: (1) Menyelidiki TKP, (2) Mengumpulkan jenis data-data dan menganalisis jenis bukti yang telah ditemukan, (3) Mengidentifikasi, mengelompokkan, mengukur, memisahkan antara pelaku, objek dan proses penanganan, (4) Membangun hubungan, mengasosiasi, dan merekontruksi ulang kejadian, (5) Menggunakan barang bukti di pengadilan.



Gambar 2. 1 Forensic Science

Forensic jaringan merupakan turunan dari forensic digital yang merupakan salah satu ilmu forensic seperti istilah pada bidang kedokteran. *Forensic science* ini mempunyai beberapa cabang turunan ilmu forensik yang dikembangkan menjadi digital forensik, computer forensik dan network forensik. Berikut Gambar 2.2 yang menjelaskan turunan ilmu forensik :



Gambar 2. 2 Turunan Ilmu Forensik

Penelitian T.Charles dan M. Pollock (2015) menyebutkan bahwa digital forensic merupakan metode ilmiah untuk melestarikan, mengoleksi, validasi, identifikasi, analisis, interpretasi, dokumentasi, dan presentasi digital untuk tujuan menfasilitasi atau merekontruksi peristiwa ditemukan tindak kriminal, atau membatu untuk mengantisipasi tindakan yang tidak sah atau terbukti mengganggu proses perencanaan operasi. (Charles & Pollock, 2015)

Penelitian Aleksandar V, Heis S, Mellisa I (2014) mendefinisikan *digital forensic* sebagai penggunaan ilmiah yang diturunkan dengan bukti metode identifikasi, pengumpulan, transportasi, penyimpanan, analisis, diartikan, dipresentasikan dan didistribusikan kembali dari bukti digital yang berasal dari sumbernya. Dengan mendapatkan otorisasi untuk semua kegiatan yang berinteraksi langsung dengan penyelidikan, melestarikan barang bukti, melacak barang bukti atau melakukan rekontruksi peristiwa ditemukannya insiden.(Server & Aktivitas, 2013).

2.2 Model Proses *Forensic*

Model proses *forensic* dijelaskan dalam empat komponenen tahapan dalam menanganinya berupa:

1. Tahap Pengoleksian (*Collection*)

   Yaitu pada tahap ini yang dilakukan meneliti dan mencari bukti-bukti, pengenaln terhadap bukti-bukti penyusupan, dan pengumpulan bukti. Sistem IDS *snort* digunakan untuk mendeteksi serangan. Pada *snort* terdapat aturan yang mengekstrak ciri dari paket yang melewati jaringan, sehingga jika ada paket ang mencurigakan dan sesuai dengan aturan lalu lintas mengirimkan pesan *alert* dan menyimpannya sebagai *log*.

2. Tahap Pemeriksaan (*Examination*)

   Adalah tahap pencarian informasi yang tersembunyi dan mengungkapkan dokumentasi yang relevan. Pemeriksaan dilakukan pada *file log* yang telah diambil menggunakan IDS *snort*. Setelah log tersimpan sebagai *alert*, maka *log* diteliti dan diperiksa.

3. Tahap Analsis (*Analysis*)

   Dari tahap pemeriksaan terlihat hasil untuk nilai pembuktian pada kasus yang ada. Tahap ini digunakan untuk menjawab pertanyaan forensik, yaitu serangan **apa** yang terjadi, IP **siapa** yang melakukan serangan, **kapan** serangan itu terjadi, **dimana** serangan itu terjadi, **bagaimana** serangan tersebut bisa terjadi, dan **mengapa** itu terjadi.

4. Tahap Pelaporan (*Reporting*)

   Penulisan laporan mengenai proses pemeriksaan dan data yang diperoleh dari semua penyelidikan, untuk membuat laporan tentang serangan yang terjadi pada jaringan dari hasil analisis bukti *log* dan setelah itu dilakukan rekontruksi aliran data dari kejadian tersebut dengan tidak merusak *file log* tersebut.

Empat komponen dalam *Digital Forensic* dari bukti digital selanjutnya akan membahas hal yang dangat penting yaitu *Chain of Custody*. *Chain of Custody* merupakan proses untuk merekam kronologi pengamanan, penahanan, pengendalian, dan pemindahan barang bukti fisik atau elektronik. *Chain of Custody* dituliskandalan dalam sebuah dokumen yang berfungsi untuk menjelaskan kronologi penanganan barang bukti tersebut, sehingga diharapkan tidak menimbulkan keraguan pada saat proses pengadilan. Ketika barang bukti akan digunakan dalam proses pengadilan, makan diperlukan penanganan yang sangat hati-hati untuk mencegah terjadinya kontaminasi atau perubahan dari barang bukti tersebut. Ide dibalik *Chain of Custody* ini adalah untuk menegaskan bahwa barang bukti tersebut memang benar-benar terkait dengan tindak kejahatan, bukan semata barang bukti yang ditanamamkan di tempat kejahatan, hanya untuk membuat seseorang tampak bersalah.

Pihak yang berwenang harus selalu memiliki akses terhadap barang bukti, mendokumentasikannya, dan meyerahkannya kepada pihak yang bertanggung jawab terhadap

evidence room (tempat pengamanan di mana barang bukti disimpan). Dokumen *Chain of Custody* tidak memiliki format yang standart atau baku, namun harus berisi informasi mengenai:

a. Barang bukti yang dikumpulkan.

b. Identitas semua penanggung jawab barang bukti.

c. Durasi penyimpanan barang bukti.

d. Pemindahan barang bukti (termasuk di dalamnya adalah tanda tangan pihak yang terlibat dalam proses pemindahan barang bukti.

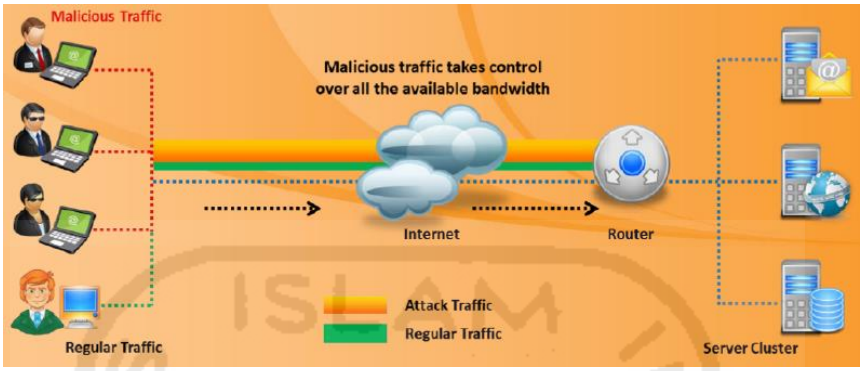
2.3 Komponen Jaringan

Beberapa jenis peralatan memurut Volonino dan Anzaldua (2008) untuk memahami bagaimana system forensic bekerja pada jaringan ang telah dilakukan untuk tingkat besar, adalah:

1. Router: sebuah computer husus yang bertujuan memindahkan data yang melintasi dua jaringan IP address ang berbeda. Router bekerja pada lapisan tiga dalam model OSI.
2. *Switch*: computer jaringan yang menggunakan *Media Access Control* (MAC) identifikasi dari sebuah host pada jaringan untuk memindahkan data dalam jaringan. *Switch* bekerja pada lapisan tiga dalam model OSI yang merupakan penghubung jaringan multiport untuk menembatani segmen jaringan.
3. Hub: merupakan bagian utama dari jaringan yang berfungsi untuk mengirimkan data yang diterima pada semua port. Perangkat ini bekerja pada layer sua karena tidak ada skema pengalamatan pada lapisan kedua. Sekarang hub jarang digunakan karena cenderung meningkatkan volume traffic dan memeperlambat jaringan sedangkan switch jauh lebih efisien dalam memindahkan data.
4. *Network Interface Card* (NIC): sebuah peangkat yang terdapat MAC (*Media Access Control*) yaitu aamat computer yang unik untuk mnegidentifikasi host atau computer. NIC adalah penghubung antara jaringan dan host.
5. Host; setipa perangkat komputasi yang terpasang ke jaringan memiliki alamat IP dan alamat MAC. Computer adalah sebuah host yang memiliki alamat IP dan alamat MAC, juga laptop, PDA, WAP, router, switch, maupun perangkat mobile seperti smartphone, ipod juga telah memiliki alamat IP dan MAC.
6. Media: sebuah bagian dari jaringan yang dapat berbentuk kabel tembaga, kabel serat optic atau gelombang radio. Memungkinkan untuk menghubungkan perangkat ke jaringan dan media yang berbeda juga proptokol yang berbeda untuk membantu menciptakan rentang waktu dan data ang dapat mengaitkan tersangka.

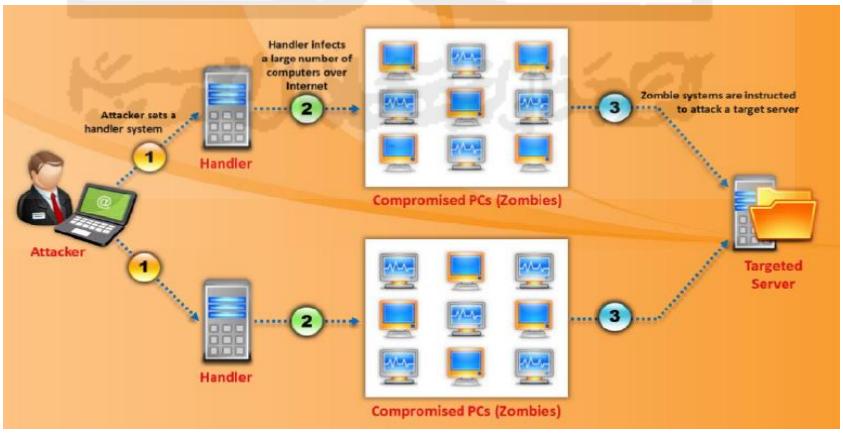
2.4 Serangan *Flooding*

Serangan *Flooding* pada modul *Certified Ethical Hacker* (CEH) terdiri dari DoS (*Denial of Service)* dan Ddos *(Distributed Denial of Service),* DoS merupakan serangan yang ditunjukkan untuk mengacaukan atau menhentikan sebuah layanan. Skema serangan DoS dapat dilihat pada Gambar 2.3.



Gambar 2. 3 Serangan DoS

Sedangkan Ddos *(Distributed Denial of Service)* adalah serangan yang dilakukan dengan banyak computer secara bersama-sama. DoS dan DdoS yang sebenarnya jenis serangan yang sama terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (resource) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut. Berikut skema serangan DDoS dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Serangan DDoS

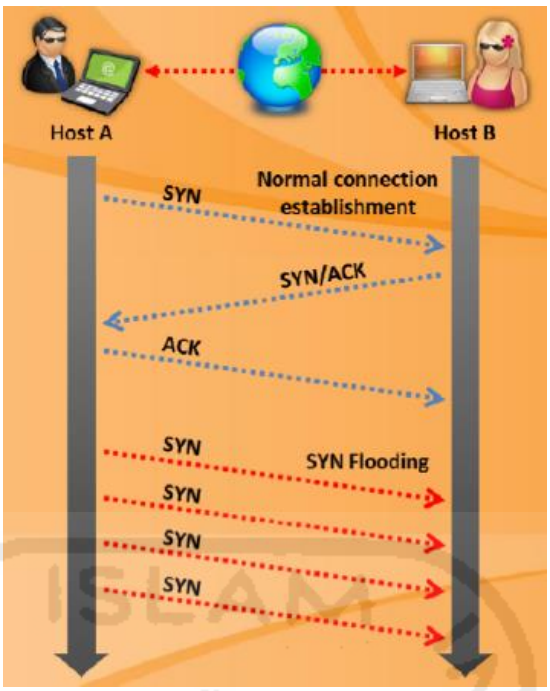
Serangan *DoS (Denial-of-Service attacks)* dan DdoS (*Distributed-Denial-of-Service*) akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut:

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

Bentuk serangan *Denial of Service* awal adalah

1. *SYN Flooding Attack*, yang pertama kali muncul pada tahun 1996 dan mengeksploitasi terhadap kelemahan yang terdapat di dalam protokol *Transmission Control Protocol (TCP)*. Serangan-serangan lainnya akhirnya dikembangkan untuk mengeksploitasi kelemahan yang terdapat di dalam sistem operasi, layanan jaringan atau aplikasi untuk menjadikan sistem, layanan jaringan, atau aplikasi tersebut tidak dapat melayani pengguna, atau bahkan mengalami crash. Beberapa tool yang digunakan untuk melakukan serangan DoS pun banyak dikembangkan setelah itu (bahkan beberapa tool dapat diperoleh secara bebas), termasuk di antaranya *Bonk, LAND, Smurf, Snork, WinNuke,* dan *Teardrop.*

Meskipun demikian, serangan terhadap TCP merupakan serangan DoS yang sering dilakukan. Hal ini disebabkan karena jenis serangan lainnya (seperti halnya memenuhi ruangan hard disk dalam sistem, mengunci salah seorang akun pengguna yang valid, atau memodifikasi tabel routing dalam sebuah router) membutuhkan penetrasi jaringan terlebih dahulu, yang kemungkinan penetrasinya kecil, apalagi jika sistem jaringan tersebut telah diperkuat. Contoh serangan paket SYN pada Gambar 2.5.

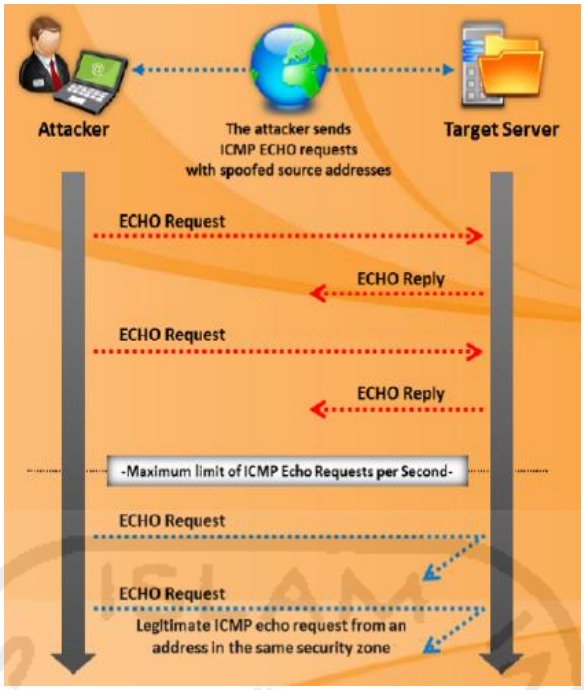


Gambar 2. 5 SYN Flooding

Dalam aturan ini, computer sumber akan mengirimkan paket SYN yang dibalas dengan paket SYN/ACK dan dibahas lagi dengan paket ACK. Sampai tiga kali berhubungan yang menjadi aturan baku, namun hacker menemukan cara untuk mengacaukan system operasi dengan megacaukankan aturan tiga respon tersebut

1. Serangan paket ICMP

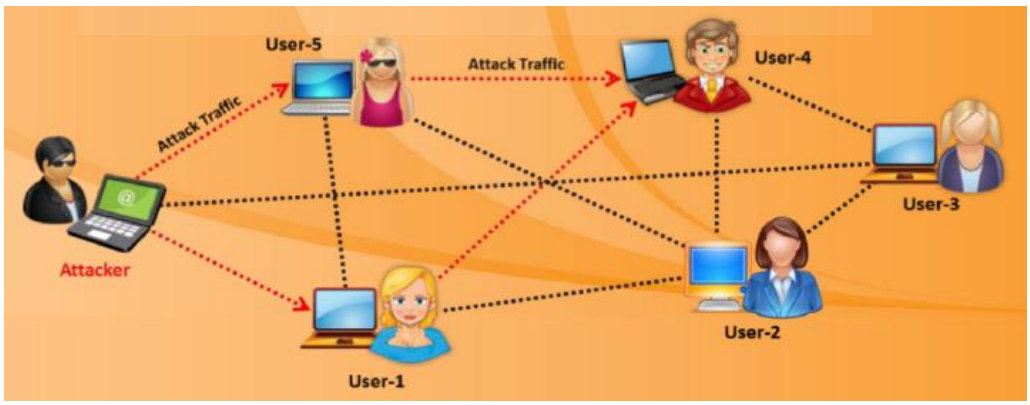
IMCP atau Internet Control Message Protokol adalah protocol sederhana yang umumnya digunakan untuk melacak keberadaan device seperti perintah PING. ICM juga digunakan oleh fungsi tracert yang digunakan untuk melacak jalur yang dilalui oleh paket data dari sumber menuju tujuan. Karena didukung oleh hampir semua operasi yang ada, paket ICMP juga digunakan oleh hacker untuk melakukan penerangan yang sesuai dengan karakteristiknya. Contoh serangan ICMP pada Gambar 2.6.



Gambar 2. 6 Serangan ICMP

2. Serangan *Peer-to-Peer*

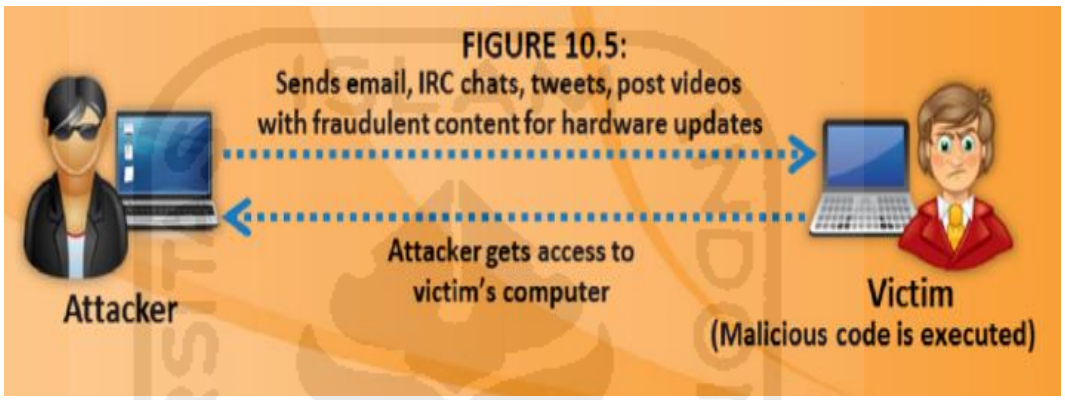
Jaringan *peer-to-peer* sangatlah popular dan digunakan oleh banyak sekali pengguna dan hacker yang kreatif pernah memanfaatkan jaringan ii untuk melakukan serangan DDoS. Serangan ini memungkinkan untuk terjadi karena adanya kelemahan pada jaringan ini pada waktu itu. *Hacker* megeksplorasi kelemahan pada direct connect (DC) yang digunakan oleh jaringan *peer-to-peer* untuk melakukan koneksi langsung. Berkat eksploitasi yang dilakukan,pengguna jaringan ini secra tidak sengaja membuat koneksi ke computer korban secara bersama-sama. Karena jumlah pengguna yang banyak, dengan koneksi yang banyak pula, secara otomatis serangan DDoS akan terjadi. Contoh serangan *peer-to-peer* pada Gambar 2.7.



Gambar 2. 7 Serangan Peer-to-peer

3. Permanen DoS

Serangan permanen DoS tidak membutuhkan banyak computer dan menimbulkan dampak yang sangat besar dan bahkan biasanya akan menyebabkan *downtime* yang cukup lama. Tujuan *hacker* adalah merusak secara permanen alat yang digunakan oleh korban. Untuk melakukan perusakan terhadap alat yang digunakan, salah satu cara yang digunakan oleh *hacker* adalah menipu korban agar agar mengupdate *firmware* alatnya dengan *firmware* palsu yang telah disiapkan oleh *hacker*. Contoh Gambar 2.8 yang menggambarkan serangan permanen DoS.



Gambar 2. 8 Permanen DoS

2.5 Intrusion Detection System (IDS)

Intrusion Detection System (IDS) adalah sebuah aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS dapat melakukan inspeksi terhadap lalu lintas inbound dan outbound dalam sebuah sistem atau jaringan, melakukan analisis dan mencari bukti dari percobaan intrusi (penyusupan). OSI model dan sensor jaringan pasif yang secara khusus diposisikan pada *choke point* pada jaringan metode dari lapisan OSI.

Kebanyakan produk IDS merupakan sistem yang bersifat pasif, mengingat tugasnya hanyalah mendeteksi intrusi yang terjadi dan memberikan peringatan kepada administrator jaringan bahwa mungkin ada serangan atau gangguan terhadap jaringan. Akhir-akhir ini, beberapa vendor juga mengembangkan IDS yang bersifat aktif yang dapat melakukan beberapa tugas untuk melindungi host atau jaringan dari serangan ketika terdeteksi, seperti halnya menutup beberapa port atau memblokir beberapa alamat IP Address.

Ada dua jenis IDS, yakni:

1. *Network-based Intrusion Detection System* (NIDS): Semua lalu lintas yang mengalir ke sebuah jaringan akan dianalisis untuk mencari apakah ada percobaan serangan atau

penyusupan ke dalam sistem jaringan. NIDS umumnya terletak di dalam segmen jaringan penting di mana server berada atau terdapat pada "pintu masuk" jaringan. Kelemahan NIDS adalah bahwa NIDS agak rumit diimplementasikan dalam sebuah jaringan yang menggunakan switch Ethernet, meskipsun beberapa vendor switch Ethernet sekarang telah menerapkan fungsi IDS di dalam switch buatannya untuk memonitor port atau koneksi.

2. *Host-based Intrusion Detection System* (HIDS): Aktivitas sebuah host jaringan individual akan dIpantau apakah terjadi sebuah percobaan serangan atau penyusupan ke dalamnya atau tidak. HIDS seringnya diletakkan pada server-server kritis di jaringan, seperti halnya firewall, web server, atau server yang terkoneksi ke Internet.

Ada beberapa cara bagaimana IDS bekerja. Cara yang paling populer adalah dengan menggunakan pendeteksian berbasis *signature* (seperti halnya yang dilakukan oleh beberapa antivirus), yang melibatkan pencocokan lalu lintas jaringan dengan basis data yang berisi cara-cara serangan dan penyusupan yang sering dilakukan oleh penyerang. Sama seperti halnya antivirus, jenis ini membutuhkan pembaruan terhadap basis data signature IDS yang bersangkutan.

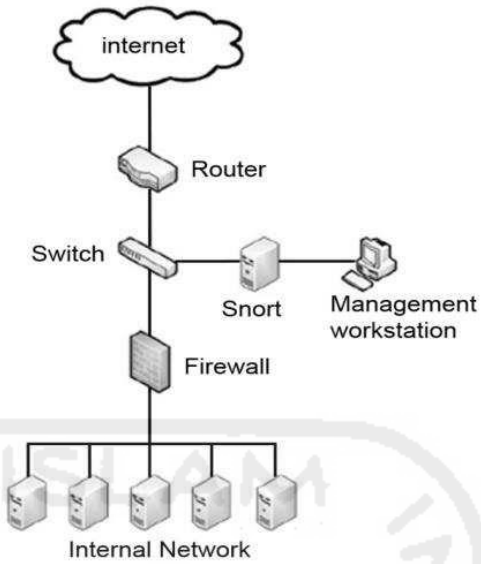
Metode selanjutnya adalah dengan mendeteksi adanya anomali, yang disebut sebagai Anomaly-based IDS. Jenis ini melibatkan pola lalu lintas yang mungkin merupakan sebuah serangan yang sedang dilakukan oleh penyerang. Umumnya, dilakukan dengan menggunakan teknik statistik untuk membandingkan lalu lintas yang sedang dIpantau dengan lalu lintas normal yang biasa terjadi. Metode ini menawarkan kelebihan dibandingkan signature-based IDS, yakni ia dapat mendeteksi bentuk serangan yang baru dan belum terdapat di dalam basis data signature IDS. Kelemahannya, adalah jenis ini sering mengeluarkan pesan false positive. Sehingga tugas administrator menjadi lebih rumit, dengan harus memilah-milah mana yang merupakan serangan yang sebenarnya dari banyaknya laporan false positive yang muncul.

Teknik lainnya yang digunakan adalah dengan memantau berkas-berkas sistem operasi, yakni dengan cara melihat apakah ada percobaan untuk mengubah beberapa berkas sistem operasi, utamanya berkas log. Teknik ini seringnya diimplementasikan di dalam HIDS, selain tentunya melakukan pemindaian terhadap log sistem untuk memantau apakah terjadi kejadian yang tidak biasa.

2.6 Snort

Snort salah satu produk open source yang secara defacto menjadi standar IDS (*Intrusion Detection System*) di industri. Snort merupakan salah satu software untuk mendeteksi instrupsi pada sistem, mampu menganalisis secara real-time traffic dan logging IP Address, mampu menganalisis port dan mendeteksi segala macam intrusion atau serangan dari luar seperti *buffter*

overflows, stealth scan, CGI attacks, SMP probes, OS fingerprinting. Skema topologi Snort dapat di lihat pada Gambar 2.9.



Gambar 2. 9 Simple Snort Network Topology

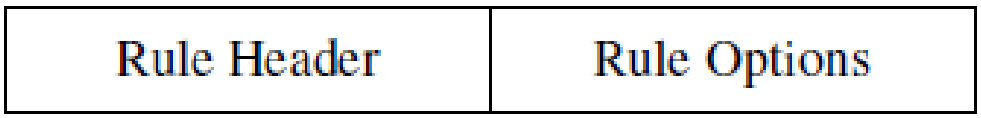
2.6.1 Komponen Snort

Menurut penelitian komponen IDS Snort dibagi menjadi beberapa bagian, berupa:

1. *Packet Dekoder:* packet dekoder mengambil paket yang sesuai dengan paket ang di tangkap dalam bentuk struktur data dan melakukan identifikasi protocol, decode IP dan kemudian TCP atau UDP yang dapat disesuaikan sesuai yang dibutuhkan.
2. *Prepocessor:* komponen atau plug-in yang dapat digunakan dengan snort untuk mengatur atau memodifikasi paket data dalam melakukan beberapa operasi untuk mengetahui apakah paket sedang digunakan oleh penyusup atau tidak.
3. *Rules Files:* merupakan suatu file teks yang berisidaftar aturan yang sintakna sudah diketahui.
4. *Detection Engine:* merupakan detection plug-in untuk mengenali paket serangan atau bukan.
5. *Output Plugins:* merupakan suatu modul yang mengatur format dari keluaran untuk alert dan file logs yang bisa diakses dan menyimpan output yang dihasilkan oleh logging dari system alert dari log.

2.6.2 Aturan Snort

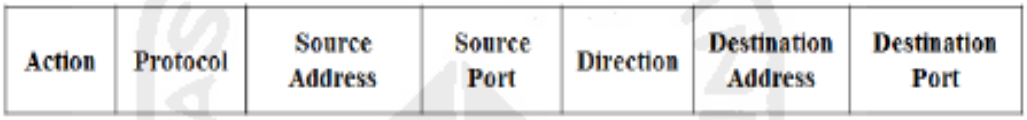
Aturan snort ada 2 struktur yang terlihat dalam Gambar 2.10 dalam penelitian (Khamphakdee, 2014) yaitu:



Gambar 2. 10 Struktur Rule

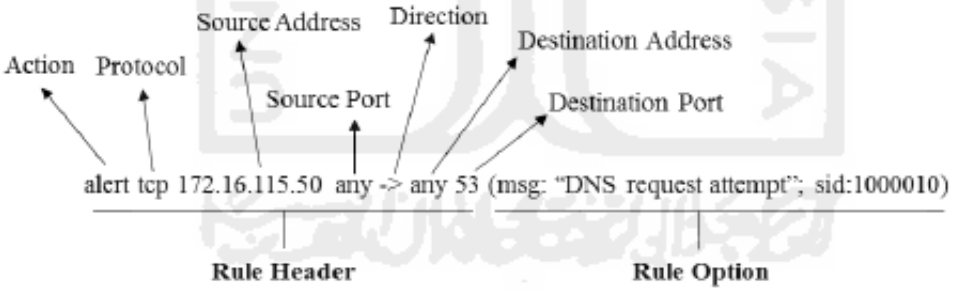
Keterangan:

a. *Rule header* : merupakan bagian rule dimana aksi-aksi *rule* diidentifikasikan *alert, log, active, dynamic,* dan lain-lainya yang termsuk diantara aksi-aksi penting yang digunakan dalam dalam desain rule Snort pada Gambar 2.11.



Gambar 2. 11 Snort IDS Rule Header Structure

b. *Rule options*: merupakan bagian rule dimana pesan-pesan peringtana (alert) di identifikasi. Contohnya:



Gambar 2. 12 Snort IDS Example

Gambar 2.12 menunjukkan contoh aturan IDS snort yang mendengus aturan yang menunjukkan kewaspadaan. Jika protokol tcp, dengan sumber alamat IP nomor 172.16.115.50 terdeteksi dari port dikirim ke alamat IP tujuan dan destination port number 53 (DNS). Selain itu juga menampilkan pesan permintaan DNS dengan jumlah aturan 1000010.

2.6.3 Kepala Aturan (rule header)

Kepala aturan mengandung informasi yang menetapkan siapa, dimana, paket beserta apa yang harus dilakukan pada saat sebuah kejadian cocok dengan atribut pada aturan. Ada Lima tindakan yang tersedia pada snort berupa:

a. *Alert*: menbuat sebuah pesan peringatan (alert) menggunakan metode alert terpilih kemudian me-log paket yang dimaksut.

b. *Log*: intruksi untuk me-log paket

c. *Pass*: mengabaikan paket.

d. *Activate*: memberi pesan peringatan lalu mengaktifkan aturan dinamis lainnya.

2.6.4 *Class Type*

Digunakan untuk mengkategorikan sebuah rule sebagai pendeteksi sebuah serangan yang menjadi bagiandari jenis serangan yang lebih umum. Snort menediakan pembagian *rules* serangan yang digunakan oleh satu set *rule* yang yang diberikan. Penetapan klasifikasi ini terbagi dalam 3 prioritas tinggi, sedang, rendah. Lihat tambel 2.1 Prioritas klasifikasi serangan.

Tabel 2. 1 Prioritas klasifikasi serangan

Classtype	Deskripsi	Tingkat
Attempted-admin	Mencoba mendapatkan hak administrasi	Tinggi
Attempted-user	Mencoba mendapatkan hak user	Tinggi
Kickass-porn	Pornografi	Tinggi
Policy-violation	Serangan privasi perusahaan	Tinggi
Shellcode-detect	Kode executable terdeteksi	Tinggi
Successful-admin	Sukses untuk mendapatkan hak administrator	Tinggi
Successful-user	Sukses mendapatkan hak user	Tinggi
Trojan-activity	Trojan jaringan terdeteksi	Tinggi
Unsuccessful-user	Tidak sukses mendapatkan hak user	Tinggi
web-aplication-attack	Serangan aplikasi web	Tinggi
attempte-dos	Percobaan Denial 0f Service (DoS)	Sedang
attempted-recon	Percobaan penyadapan informasi	Sedang
bad-unknown	Trafik yang jelek atau rusak	Sedang
default-login-attemp	Mencoba login dengan default username dan password	Sedang

Tabel 2. 2 Con't Prioritas klasifikasi serangan

Classtype	**Deskripsi**	**Tingkat**
denial-of-service	Deteksi atas sebush serangan Denial of Service (DoS)	Sedang
misc-attack	Serangan lain-lain	Sedang
non-standard-protocol	Deteksi atas protocol atau event non-standar	Sedang
rpc-portmap-decode	Decode pada RPC query	Sedang
succesfull-dos	Serangan Denial of Service (DoS)	Sedang
succesfull-recon-langescale	Sabotase informasi besar-besaran	Sedang
succesfulli-recon-limited	Penadapan informasi	Sedang
suspicious--filename-detect	Nama file yang mencurigakan terdeteksi	Sedang
suspicious-login	Sebuah usaha login menggunakan username yang mencurigakan	Sedang
system-call-detect	Sebuah system call terdeteksi	Sedang
unusual-client-port-connection	Klien yang menggunakan port yan tidak biasa	Sedang
web-aplication-activity	Akses ke sebuah aplikasi web yang rentan	Sedang
icmp-event	Event umum ICMP	Rendah
misc-activity	Aktivitas mencurigakan	Rendah
Network scan	Terdeteksi scan jaringan	Rendah
Not-suspicious	Trafik yang mencurigakan	Rendah
Protokol-command-decode	Decode pada perintah protocol terdeteksi	Rendah
String-detect	Sebuah string mencurigakan terdeteksi	Rendah
Unknown	Trafik yang tidak diketahui	Rendah
Tcp-connection	Sebuah koneksi TCP terdeteksi	Rendah

2.6.5 Jenis Opsi *Rule*

Opsi rule yang terdapat didalam rule option dapat dipisahkan dengan karakter semi colom (;) yang menggambarkan kemudahan penggunaan kekuatan dan fleksibilitas yang berupa kategori:

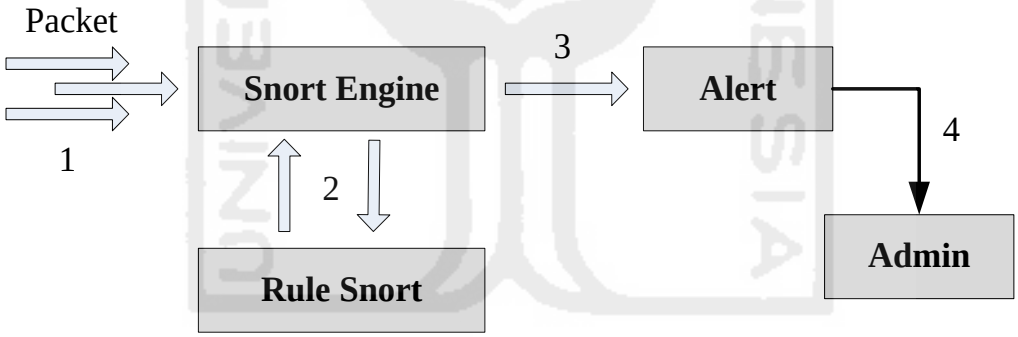
1. General: opsi yang menyediakan informasi tentang rule.
2. Payload: opsi ini mencari data dalam *payload* paket.
3. Non payload: opsi ini mencari data non-payload
4. Post detection: opsi ini merupakan pemicu aturan tertentu yang berjalan setelah aturan tersebut diaktifkan.

2.6.6 Membaca Aturan Snort

Aturan snort yaitu kumpulan aturan perilaku snort berupa tahapan aturan seperti mengidentifikasi karakteristik dari trafik yang dicurigai, menulis rule berdasarkan karakteristik, mengimplementasi aturan, mengecek trafik yang dicurigai, mengubah aturan sesuai dengan pengetesan, mengetes dan mengecek hasilnya.

Sedangkan dilihat dari kemampuan mendeteksi penyusupan pada jaringan, IDS dibagi menjadi: *knowledge-based* atau *misuse detectiondan behavior based* atau *anomaly based*. *Knowledge-based* dapat mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkan dengan database aturan IDS Snort (berisi catatan serangan). Sedangkan *behavior based* (anomaly) dapat mendeteksi adanya penusupan dengan mengamati adanya kejanggalan-kejanggalan pada sistem, atau adanya penyimpangan-penyimpangan dari kondisi normal.

Sedangkan dilihat dari kemampuan mendeteksi penyusupan pada jaringan, IDS dibagi menjadi dua yakni: *host based* dan *network based*. *Host based* mampu mendeteksi hanya pada host tempat implementasi IDS, sedangakn *network based* IDS mampu mendeteksi seluruh host yang berada satu jaringan dengan host implementasi IDS tersebut. Berikut Gambar 2.13 yang menjelaskan *block* diagram sistem pencegahan penyusupan.



Gambar 2. 13 Block Diagram

Keterangan:

1. Packet
2. Snort engine berfungsi untuk membaca paket data dan membandingkannya dengan aturan basis data, jika paket data diibaratkan sebagai penyusup/serangan, maka snort engine akan menghasilkan alert (berbentuk file log).
3. Rule snort menyediakan aturan berupa jenis pola serangan. Rule ini berupa file text yang disusun dengan aturan tertentu. Setelah paket data melintasi jaringan maka akan di deteksi oleh aturan snort.
4. Alert bagian ini merupakan pencatatan serangan pada sebuah file log. Apabila paket yang melintasi jaringan sesuai dengan pola ang ada maka akan muncul tanda peringatan.

Bab 3 Metodologi Penelitian

Bab ini menjelaskan bagaimana cara penelitian ini dilakukan, sehingga dapat memberikan rincian tentang alur atau langkah-langkah yang dibuat secara sistematis serta dapat digunakan dijadikan pedoman dengan jelas dalam menyelesaikan masalah, membuat analisa terhadap hasil penelitian, serta kesulitan yang digadapi. Adapun tahapan-tahapan atau langkah-langkah pada penelitian ini dapat dilihat pada Gambar 3.1.

Gambar 3. 1 Alur Metodologi Penelitian

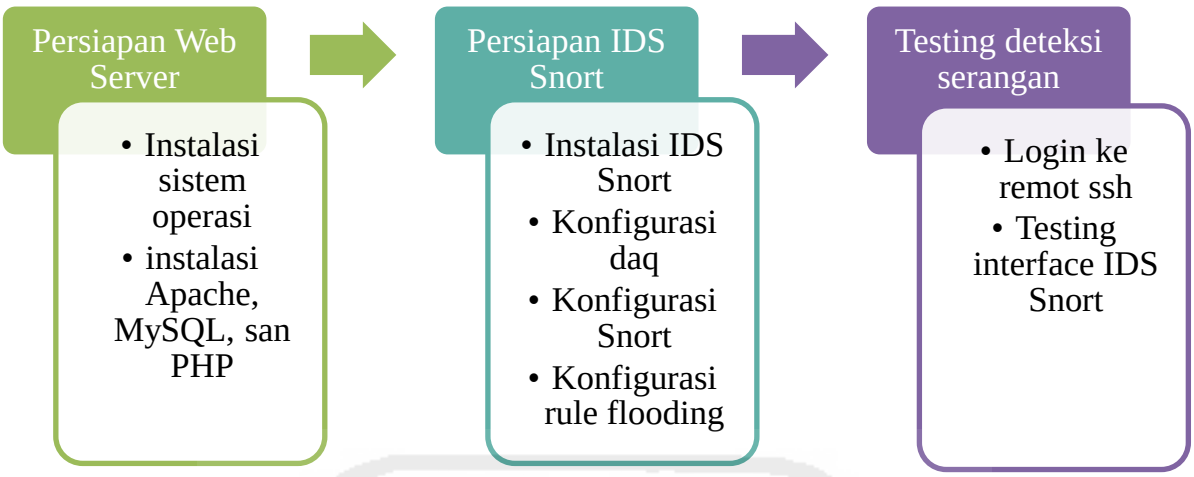
3.1 Literatur Review

Literatur review dilakukan untuk mendapatkan informasi mengenai topik-topik yang akan diteliti yang dapat diperoleh dari buku,dokumen, artikel, atau bahan tertulis lainnya yang berupa buku laporan, teori, maupun penemuan lainya yang bersifat *online* maupun *offline* yang bertujuan memberikan informasi.

Review atau kajian pustaka dilakukan untuk tujuan terhadap dilkukannya penelitian yang terkait dengan masalah-masalah yang terkait untuk mendeteksi serangan berbasis *Intrusion Detection System* (IDS), berikut juga metode yang digunakan untuk melakukan proses deteksi agar dapat menunjang tujuan ahir dalam penelitian ini.

3.2 Identifikasi Sistem

Merupakan tahap perancangan dan implementasi sistem jaringan *Intrusion Detection System* (IDS) yang akan digunakan sebagai objek penelitian pada Gambar 3.2. Terdiri dari beberapa komponen berupa, persiapan system, persiapan IDS dan testing deteksi serangan.

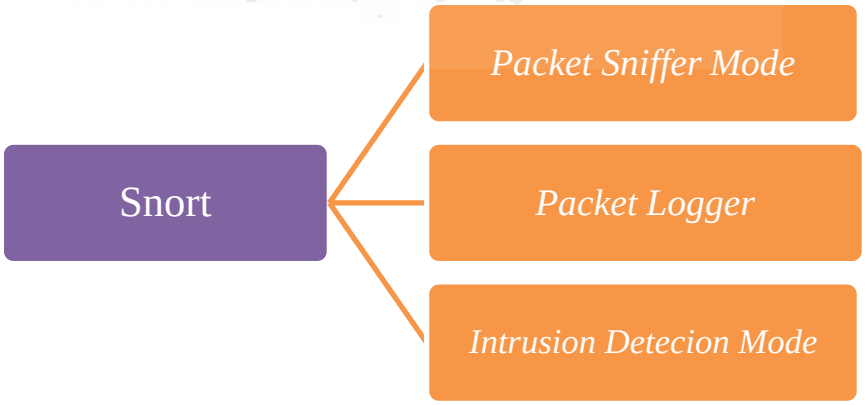


Gambar 3. 2 Tahapan Implementasi Intrusion Detection System (IDS) Snort

3.3 Konfigurasi Snort

Mempersiapkan *Intrsion Detection System* (IDS) Snort yang merupakan paket utama yang dibutuhkan dalam sistem, paket yang digunakan adalah paket *default snort* dari ubuntu yang dapat diinstal langsung dari *terminal console linux*, melakukan Konfigurasi *snort* merupa konfigurasi daq, konfigurasi *rules* yang tujuannya adalah menganalisis *packet* berdasarkan *rule* yang ada untuk mengenali adanya upaya serangan *hacker*. Sedangkan konfigurasi ini dilakukan agar *log* pada *Snort* dapat terbaca oleh *databse*.

Untuk memudahkan dalam memahami sistem pendeteksi penyusup oleh snort dapat di jelaskan pada Gambar 3.3 sebagai berikut:



Gambar 3. 3 Prinsip Kerja Sistem Snort

Dari Gambar 3.3 Dapat dijelaskan sebagai berikut:

1. *Snort*

   Snort merupakan aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS dapat melakukan inspeksi terhadap lalu lintas dalam sebuah jaringan, melakukan analisis dan mencari bukti dari percobaan *intrusi* (penyusupan).

2. *Packet sniffer mode*

   Dalam packet sniffer mode, *snort* bekerja sebagai sniffer sama seperti Wireshark. yang dapat melihat lalu lintas data pada jaringan komputer. Dikarenakan data mengalir secara bolak-balik pada jaringan, menangkap tiap-tiap paket dan kadang-kadang menguraikan isi dari RFC (Request for Comments) atau spesifikasi yang lain.

3. *Packet Logger*

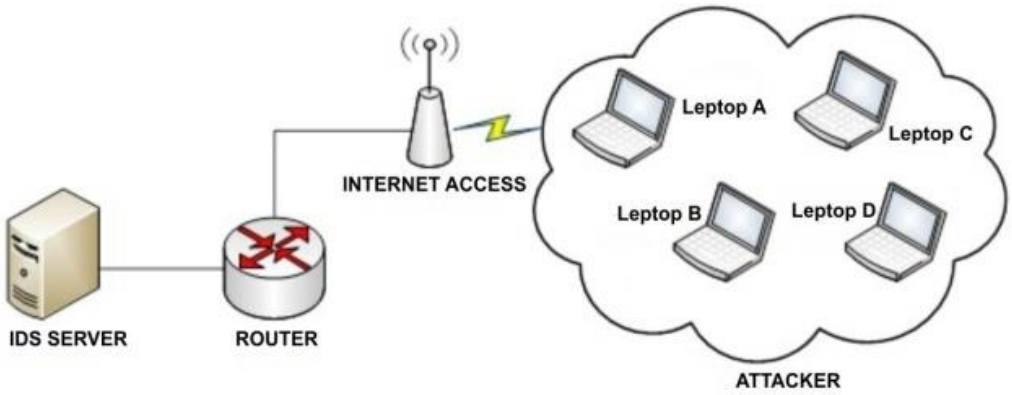
   Mencatat semua paket yang lewat pada jaringan untuk dapat di analisis.

4. *Intrusion Detection Mode*

   Pada mode ini Snort akan berfungsi untuk mendeteksi serangan yang dilakukan melalui jaringan computer. Untuk menggunakan mode IDS ini diperlukan setup dari berbagai jenis rule atau aturan yang akan membedakan sebuah paket normal dengan paket yang membawa serangan. Contoh Rule untuk membuat file baru # nano /etc/snort/rules/local.rules dan isi dengan code alert tcp any any → 192.168.1.0/24 111 (content:"|00 01 86 a5|";msg:"mountd access";).

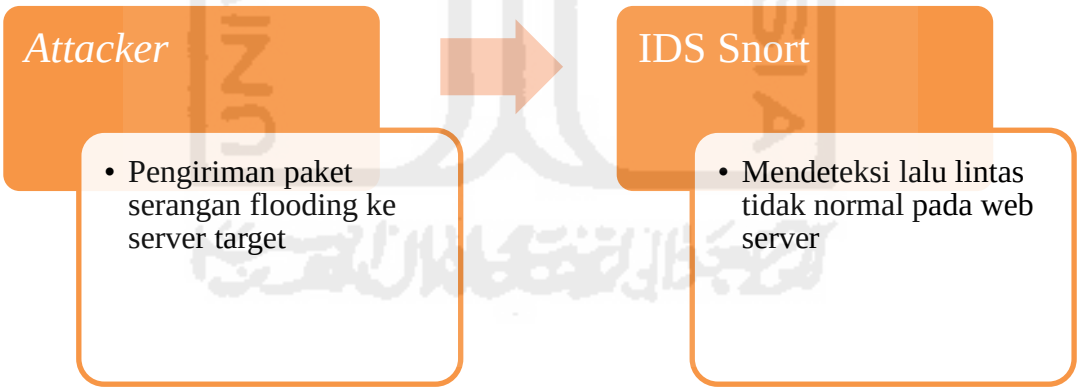
3.4 Simulasi Kasus

Merupakan tahap dilakukannya simulasi kasus untuk mencoba mengimplementasikan snort dalam mendeteksi penyusupan atau serangan. Simulasi kasus bertujuan untuk melakukan pengujian terhadap snort dalam mendeteksi penyusup atau serangan yang melakukan tindak kejahatan pada web server target yang digunakan untuk melindungi jaringan dengan kemampuan untuk merespon sesuai dengan kebijakan keamanan dari IDS Snort. Simulasi kasus *Intrusion Detection System* (IDS) Snort yang akan dijalankan mengguankan skenario pengiriman paket serangan menggunakan beberapa tool untuk menyerang web server target sekaligus menunjukkan bahaya yang dapat ditimbulkan oleh serangan. Gambar 3.4 menunjukkan gambaran umum dari skenario kasus deteksi serangan *flooding* pada web server.



Gambar 3. 4 Simulasi Kasus

Skema serangan adalah ketika *attacker* mengirimkan paket *flood* kepada target web server maka *Intrution Detection System* (IDS) akan mendeteksi adanya serangan lalu lintas trafik yang meningkat sesuai dengan rule flood yang telah ditentukan pada *Intrusion Detection System* (IDS). Pengiriman paket tersebut akan menyebabkan akses web server lambat, bahkan server akan mati ketika pengiriman serangan paket tersebut melebihi beban yang dimiliki server. Gambar 3.5 menunjuukan tahapan simulasi kasus deteksi serangan *flooding* pada web server yang menunjukkan *attacker* dalam mengirimkan serangan *flooding* ke target, sehingga *Intrusion Detection System* (IDS) mendeteksi lalu lintas pada web.



Gambar 3. 5 Tahap Simulasi Serangan Flooding Pada Web Server

3.5 Analisis

Menurut Muh Al Azhar forensik adalah suatu proses ilmiah atau suatu usaha ilmiah yang didasari ilmu pengetahuan dalam memngumpulkan, menganalisa dan menghadirkan bukti dalam suatupersidangan di pengadilan untuk membantu pengungkapan suatu kejahatan melalui pengungkapa bukti-bukti yang sah menurut undang-undang dan peraturan yang berlaku. Digital

forensik merupakan aplikasi bidang ilmu pengetahuan dan teknologi komputer untuk kepentingan pembuktian hukum (pro justice), yang dalam hal ini adalah untuk membuktikan kejahatan berteknologi tinggi atau computer crime secara ilmiah (scientific) sehingga bisa mendapatkan bukti-bukti digital yang dapat digunakan untuk menjerat pelaku kejahatan tersebut. Disinilah tugas untuk para investigator dalam menangani kasus penyelidikan untuk dapat merecover ulang ejadian peristiwa tindak kriminal.

Sebelum dilakukannya proses model forensic maka terlebih dahulu melakukan uji tes deteksi serangan *flooding* menggunakan simulasi serangan yang nantinya akan dicocokkan dengan *rule* yang telah ditentukan ke dalam *Intrution Detection System* (IDS) Snort. Tahap deteksi serangan *flooding* ini terdiri dari beberapa proses, yaitu:

1. Tahap Pengoleksian (collection)

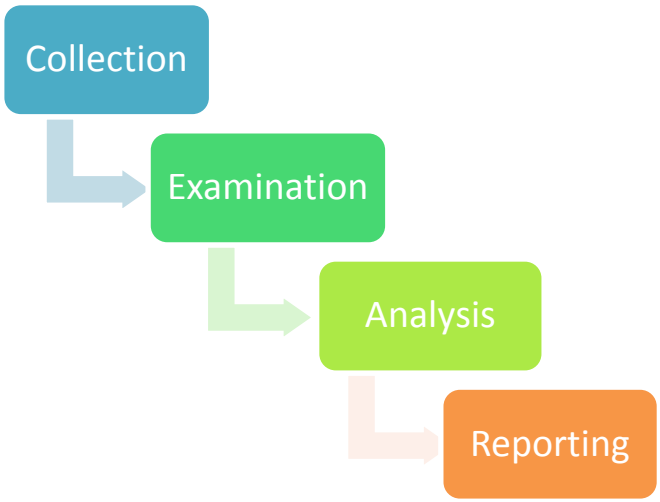
Proses pengoleksian merupakan proses pertama dalam model proses forensic untuk meneliti dan mencari barang bukti, pengenalan terhadap bukti-bukti penusupan, dan pengumpulan bukti dari *Intrution Detection System* (IDS) Snort yang melewati jaringan. Sehingga jika ada paket yang mencurigakan dan sesuai dengan aturan lalu lintas mengirimkan pesan *alert* dan menimpan sebagai log snort.

2. Tahap Pemeriksaan (Examination)

Pada tahap permeiksaan ini digunakan untuk mencari informasi yang tersembunyi dan mengungkapkan dokumen file log snort yang telah tersimpan sebagai alert dan hasil capture trafik web server untuk diperiksa.

3. Tahap Analisis

Pada tahap proses analisis dilakukan terhadap file log snort untuk mengetahui serangan apa yang terjadi, IP siapa yang melakukan serangan, kapan serangan terjadi, dimana serangan itu terjadi, bagaimana serangan tersebut bisa terjadi, dan mengapa itu terjadi. analisis dapat dilakukan dengan *tool* Wireshark. Model proses forensik ini dapat dilihat pada Gamabr 3.6 berikut ini.



Gambar 3. 6 Model Proses Forensik

Hasil dari model proses forensik akan dipresentasikan seperti tampak pada table dibawah ini kolom *timesstamp, source address, destination address, protocol, source port, destination port* dan *payload* atau isi pesan pada tabel 3.1 dibawah ini:

Tabel 3. 1 Pengelompokan Data

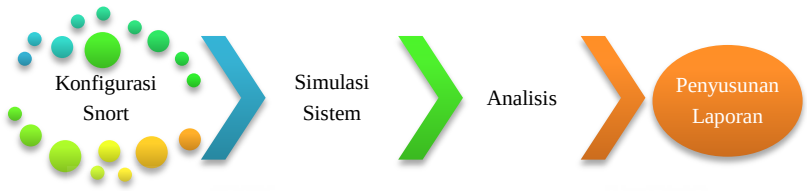
No	Timestamp	Source Address	Dest. Address	Protocol	Source Port	Dest. Port	Payload/Pesan

Setelah dikumpulkan barang bukti forensic log snort maka di dapat jumlah IP addres yang mencoba melakukan serangan *flooding* pada web server di lingkungan Univeritas Muhammadiyah Magelang.

3.6 Laporan

Merupakan tahap pembuatan laporan dan hasil pembuktian identifikasi serangan yang masuk ke dalam snort untuk pengujian mendeteksi penyusupan serta dapat mendapatkan informasi mengenai penyerang berdasarkan Tabel 3.1 dan 3.2 agar dapat mengurangi tingkat kerentanan terhadap serangan. Laporan berisi mengenai pendahuluan, litelatur review, metodologi penelitian, hasil dan pembahasan, serta penutup.

Kesimpulan yang diperoleh dari peelitian ini akan dimasukkan ke dalam bagian penutup dari laporan, berikut juga saran untuk penelitian-penelitian selanjutnya, khususnya yang mengambil penelitian tentang network *Intrusion Detection System* (IDS). Gambar 3.7 menunjukkan tahapan penusunan laporan dalam penelitian.



Gambar 3. 7 Tahapan Penyusunan Laporan

Laporan yang disusun pada ahirnya diharapkan dapat memberikan gambaran secara meyeluruh mengenai topik penelitian ini, serta dapat memberikan rekomendasi yang bermanfaat untuk penelitian selanjutnya.

Bab 4 Analisis dan Hasil

Bab ini membahas tentang langkah-langkah penelitian, analisis dan hasil yang didapatkan dari penelitian ini. Pembahasan dalam bab ini meliputi tahap studi identifikasi system yang digunakan untuk objek penelitian target web server, tahap konfigurasi digunakan untuk mengkonfigurasi *Intrutionn Detection System* (IDS) yang digunakan untuk menguji dalam mendeteksi serangan flooding pada web server. Tahap analisis digunakan untuk mencari barang bukti dari hasil file log *Intrutionn Detection System* (IDS) Snort menggunakan model proses forensic.

4.1 Literatur Review

Pada tahapan ini dilakukan kajian literature terhadap penelitian terkait serangan *flooding*, penerapan *Intrusion Detection System* (IDS), dan Snort yang akan dijadikan landasan teknis dalam penelitian ini. Penelitian yang dilakukan oleh (Lanke & Jacob, 2014) menjelaskan bahwa teknik *flooding* merupakan serangan yang ditunjukkan untuk mengacaukan atau menghentikan sebuah layanan secara bersama-sama. Aktifitas *flooding* merupakan serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (*resource*) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut(Lanke & Jacob, 2014)(Lanke & Jacob, 2014)(Lanke & Jacob, 2014)(Lanke & Jacob, 2014)(Lanke & Jacob, 2014). Salah satu teknik yang digunakan untuk mendeteksi serangan flooding adalah penerapan *Intrusion Detection System* (IDS), seperti pada penelitian (Stiawan et al., 2012) yang menggunakan *Intrusion Detection System* (IDS) untuk mengidentifikasi adanya serangan dan dapat memberikan peringatan serangan yang terjadi pada *web server*.

Selanjutnya pada penelitian (Indra, 2010) menyebutkan bahwa untuk mengurangi resiko pada celah gangguan keamanan pada jaringan web server menggunakan tool Snort yang merupakan *Intrusion Detection System* (IDS) paling unggul dalam menganalisis lalu lintas dan *packet logging* IP *network.* Snort digunakan untuk mendeteksi ancaman seperti *buffer overflows, port*

scanning, nmap maupun *port scanner* lainya. Snort memberikan informasi serangan berupa alert log. Pada penelitian ini akan diterapkan sistem *Intrusion Detection System* (IDS) Snort untuk mendeteksi serangan *flooding* pada *web server* Universitas Muhammadiyah Magelang.

4.2. Identifikasi Sistem

Tahap identifikasi sistem jaringan *Intrusion Detection System* (IDS) Snort yang akan digunakan sebagai objek penelitian. Terdiri dari beberapa komponen berupa:

a. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam penelitian ini menggunakan beberapa komponen jaringan, berupa:

- PC dengan merk Samsung adalah sebagai berikut:

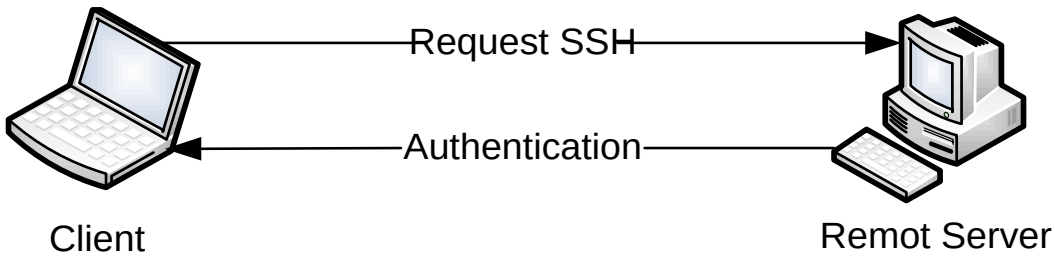
  Prosesor :Intel (R) Core (TM)2 /duo
  RAM : 4 GB
  HDD : 500 GB
  Graphic Card : Intel HD 3000 dan AMD Radeon
- *Router*
- *Switch*
- *Internet Interface Card* (NIC)

b. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak ang digunakan untuk web server dan kebutuhan forensik dalam penelitian ini adalah:

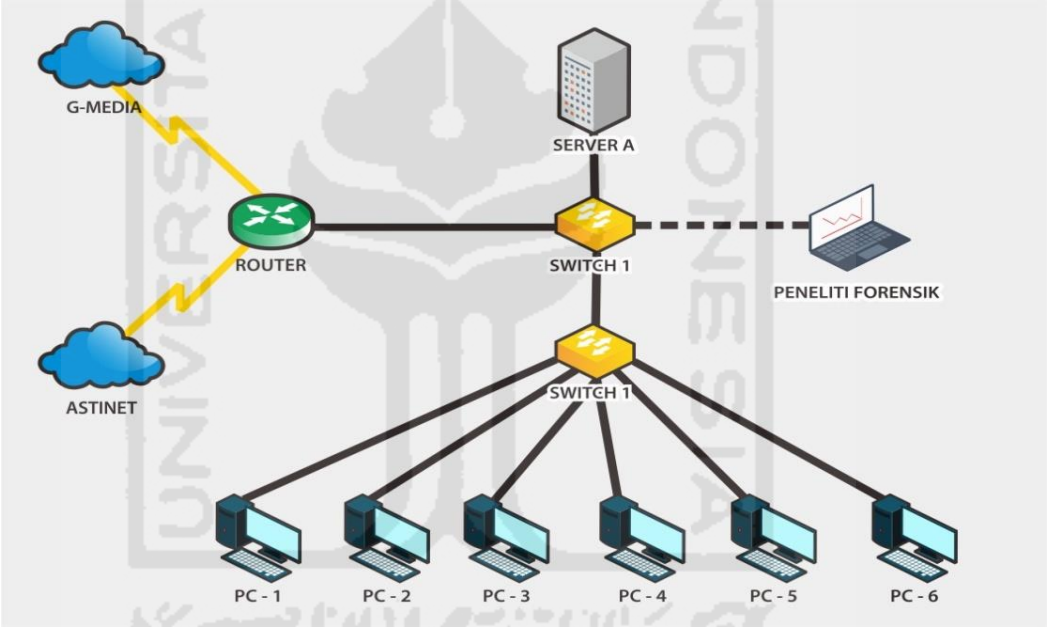
- OS Debian 8 (Apache Web Server, MySQL, Database Server)
- Snort

Penelitian forensic jaringan ini menggunakan server kampus yang bertindak sebagai target saerangan pada saat implementasi. Server ini menggunakan IP static 203.x.x.x yang diakses melalui jaringan internet. Remote server dapat dihubungi dengan menggunakan protocol ssh pada port 22 sehingga proses komunikasi menjadi lebih aman. Lihat Gambar 4.1 yang menunjukkan proses authentifikasi server.



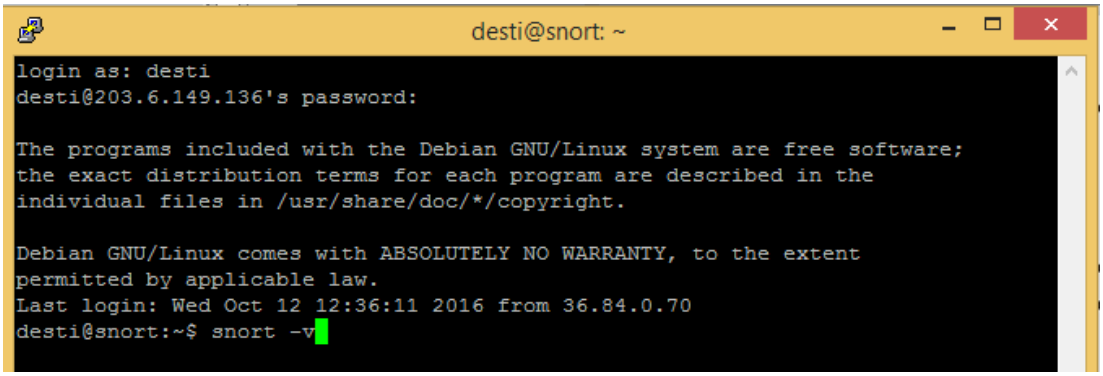
Gambar 4. 1 Proses Authentifikasi Server

Gambar 4.1 Melukiskan proses authentifikasi menggunakan *protocol* ssh yang lebih aman dikarenakan menggunakan enkripsi dalam pertukaran data. Letak dari server forensik jaringan dapat dilihat di Gambar 4.2 yaitu artsitektur dari forensik.



Gambar 4. 2 Arsitektur Forensik Jaringan

Selanjutnya, Gambar 4.3 Menggambarkan cara menggunakan *remote* pada server forensic jaringan dengan masuk ke *root* untuk konfigurasi snort.



Gambar 4. 3 Remote Server

Tahap awal konfigurasi merupakan proses instalasi IDS Snort kemudian memasukkan rule bawaan dari snort untuk mendeteksi serangan flooding. berikut contoh beberapa rule untuk deteksi serangan *flooding* sebagai berikut:

```
# alert tcp $HOME_NET 20432 -> $EXTERNAL_NET any (msg:"MALWARE-OTHER shaft
client login to handler"; flow:to_client,established; content:"login|3A|";
fast_pattern:only; metadata:ruleset community; reference:cve,2000-0138;
reference:url,security.royans.net/info/posts/bugtraq_ddos3.shtml;
classtype:attempted-dos; sid:230; rev:13;)
```

4.3 Simulasi Serangan Flooding

Proses simulasi merupakan merupakan tahap awal yang dilakukan untuk menguji konfigurasi *Intrucion Detection System* (IDS) Snort dalam mendeteksi serangan *flooding*. Simulasi serangan menggunakan alat bantu LOIC (*Low Orbit Ion Canon*). LOIC (*Low Orbit Ion Canon*) merupakan alat yang digunakan untuk menguji serangan pada target web server. Alat ini mempunyai kelebihan dapat melakukan pengiriman paket *request* berdasarkan protokol TCP, UDP maupun ICMP. Selain itu target pada *port* yang akan dikirim dapat ditentukan oleh penyerang. Dalam pengujian ini, LOIC digunakan untuk melakukan serangan ke port 80.

Alasan penggunaan port tersebut sebagai target adalah *port* tersebut merupakan *port* yang digunakan dalam mengakses web server yang digunakan oleh pengguna dalam mengakses informasi menggunakan jaringan internet. Proses pengujian serangan dengan memasukkan alamat IP target pada aplikasi LOIC dari mesin *attacker* pada menu 1 (satu) atau *select your target*, kemudian tetapkan alamat yang akan diserang menggunakan tombol *lock on* yang berada pada menu 1 (satu), selanjutkan menentukan target *port* adalah 80, target protokol adalah UDP, jumlah *threads* yang akan dikirimkan sebanak 10, dan kecepatan pengiriman paket pada tingkat faster pada menu 3 (tiga), Aplikasi LOIC ditunjukkan pada Gambar 4.4.

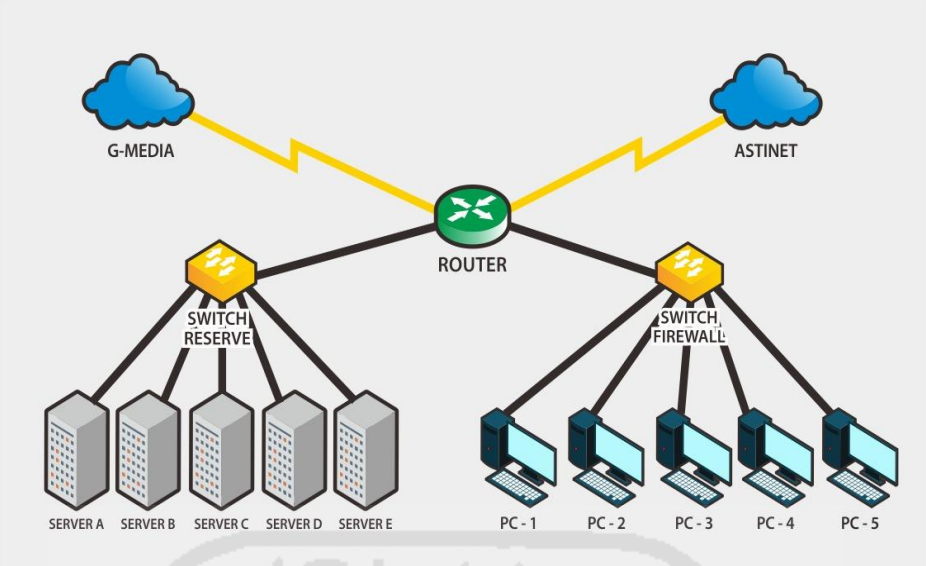
Gambar 4. 4 Serangan Flooding

Setelah semua konfigurasi dimasukkan pada aplikasi LOIC kemudian melakukan seranngan dengan menekan tombol *start flooding* untuk memulai atau *stop flooding* untuk menghentikan serangan. Serangan *flooding* dilakukan selama 5-15 menit.

4.4 Analisis dan Investigasi Forensik

(Lipeng, Xingyuan, Huilin, & Wang, 2013) teknologi IDS bertujuan untuk mengidentifikasi instruksi ilegal yang tersembunyi pada jaringan lalu lintas yang diserang, penelitian ini berkomitmen untuk menyediakan sutu metode generasi yang sistematis dan dan ilmiah untuk menghindari berbagai bentuk serangan dengan menggunakan kerangka kerja dan rekomendasi untuk pertahanan dari serangan yang masuk.

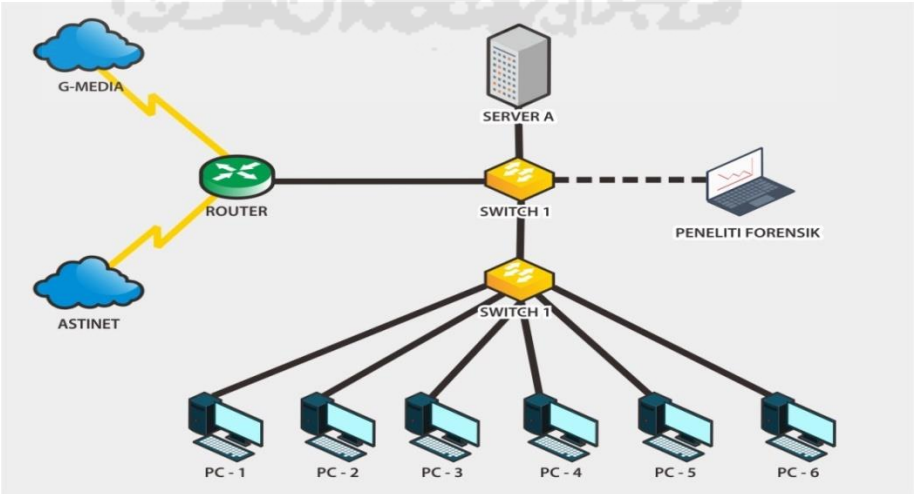
Pusat Pelayanan Teknologi Informasi dan Komunikasi UMMgl (TIK UMMgl) yang merupakan lembaga pelayanan yang berfokus pada pengolahan data, dan interkoneksi kampus. Gambar 4.5 adalah topologi jaringan Universitas Muhammadiyah Magelang adalah distributed (menyebar), pengembanangan dari topologi star, TIK UMMgl menjadi pusat jaringan sekaligus pembagi bandwidth dari tiap-tiap fakultas. Pembagian bandwith ditentukan berdasarkan fakultas.



Gambar 4. 5 Topologi Jaringan UMMgl

4.4.1 Implementasi Forensik Jaringan

Implementasi pada penelitian forensik jaringan terdapat pada rancangan arsitektur forensic jaringan seperti gambar yang ditunjukkan pada gambar 4.6 yang merupakan arsitektur forensik jaringan Universitas Muhammadiyah Magelang dalam mendeteksi serangan *flooding* menggunakan *Intrusion Detection System* (IDS) Snort. User yang ingin mengakses server yang ada di UMMgl harus melewati switch terlebih dahulu lalu masuk ke dalam proxy kemudian server. Server IDS diletakkan sejajar dengan core switch untuk mendeteksi tindakan illegal pada jaringan. Pengambilan data dilakukan oleh peneliti dengan cara paket *sniffer* yang ada pada *Intrusion Detection System* (IDS) Snort yang telah terekam.



Gambar 4. 6 Arsitektur Forensik jaringan

Setelah data tersebut terekam maka proses analisis dilakukan oleh peneliti dalam menguraikan karakteristik file log flooding yang telah terdeteksi oleh IDS Snort. Pada gambar 4.6 Peneliti bertindak sebagai investigator forensik jaringan dimulai dari mempersiapkan sistem, *Intrusion Detection System* (IDS) Snort untuk mendeteksi penyusup pada jaringan, dan menganalisis karakterinstik file log yang didapat dari IDS Snort. Pada Snort telah dimasukkan aturan sebagai pendeteksi pola pada jaringan.

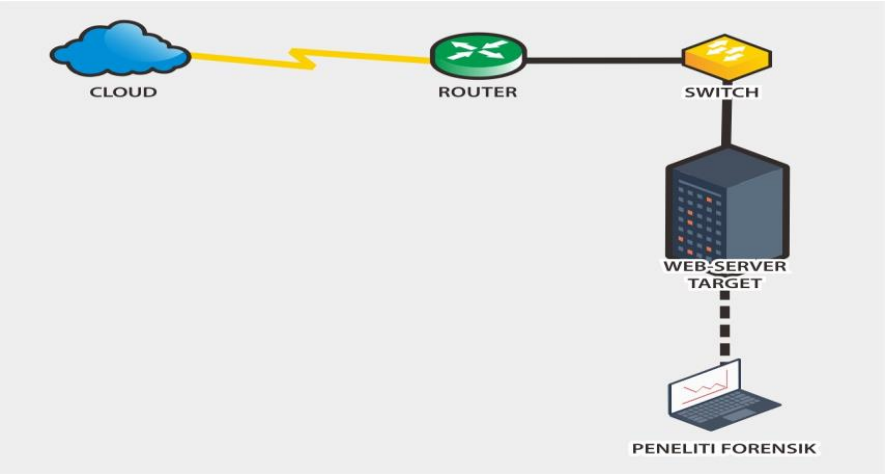
Setelah aturan ditentukan, lalu menangkap trafik jaringan yang memiliki pola yang sama dengan aturan yang telah dimasukkan ke *Intrusion Detection System* (IDS). Dan jika tidak sama pola nya maka tidak akan ditangkap oleh IDS-nya. Agar file log yang didapat bias dianalisis di wireshark maka perlu memasukkan skrip ke *Intrusion Detection System* IDS Snort dengan parameter "snort -r /var/log/snort/snort.log -l /var/log/snort/" sehingga yang dihasilkan dengan file log yang dihasilkan dengan angka biner misalnya snort.log.1498549117.

Selanjutnya, dilakukan data cleaning suatu file log terdiri dari banyak data sehingga perlu dilakukan cleaning agar data yang mau diproses sesuai dengan yang diinginkan. Pengambilan log dilakukan secara real time, kemudian dikumpulkan file log tersebut agar bias dianalisis oleh peneliti investigator. Dengan menggunaan model proses forensic, sebuah file log seharusnya bisa menjawab pertanyaan penyerangan apa yang terjadi, siapa yang menyerang dari IP addressnya, kapan terjadi penyerangan, di server manakah telah terjadi penyerangan, bagaimana itu bisa terjadi dan mengapa itu bisa terjadi.

4.4.2 Analisis Model Proses Forensik

a. Tahap Pengoleksian (*collection*)

Pengoleksian barang bukti pada penelitian ini menggunakan hasil *record* dari trafik IDS. IDS diimplementasikan kurang lebih selama 3 bulan selama penelitian berlangsung. Proses rekonstruksi dimulai setalah *Intrusion Detection System* (IDS) Snort menangkap trafik yang dianggap *rule* yang telah ditetapkan. Proses pengambilan *payload* sebagai data serangan *flooding* dalam penelitian ini sebagai berikut:



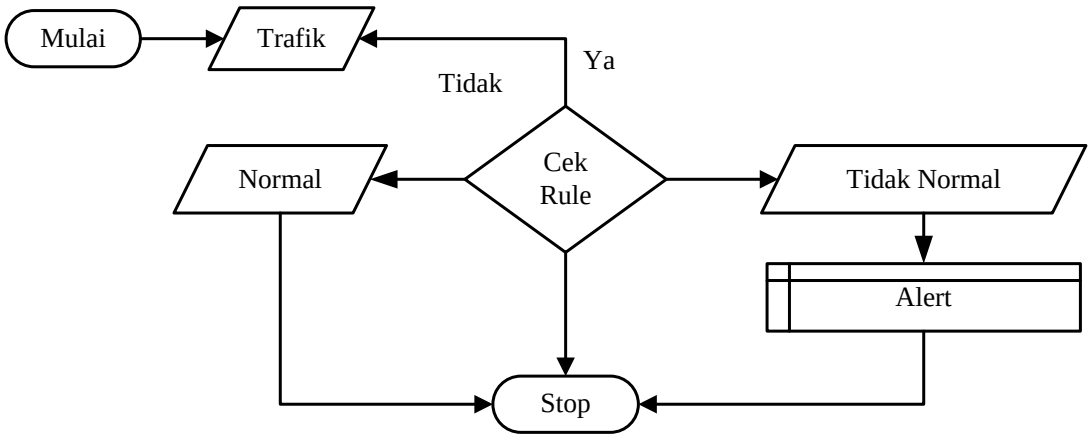
Gambar 4. 7 Proses Pengambilan Data

Gambar 4.7 melukiskan bahwa jaringan yang terhubung dan terkoneksi ke internet, kemudian switch tersebut terhubung dengan server *Intrusion Detection System* (IDS) Snort, sehingga apabila ada *traffic* yang bersifat *anomaly* maka akan langsung terdeteksi oleh Snort dan berbunyi alarm.

b. Tahap Pemeriksaan (*Examination*)

Peneliti menggunakan *Intrusion Detection System* (IDS) Snort untuk memeriksa penyusupan pada jaringan sehingga jika ingin mengambil *file log* dalam bentuk p.cap (*packet capture*) diperlukan skrip atau parameter untuk dipasang di Snort menggunakan "snort -r /var/log/snort/snort.log -l /var/log/snort/" agar *file log* yang digunakan dalam bentuk default p.cap.

Proses pemeriksaan *file log* dilakukan menggunakan hasil rekaman dari *Intrusion Detection System* (IDS) Snort, pemeriksaan rekaman data akan dikumpulkan dengan cara *packet sniffer* yang terdapat pada server *Intrusion Detection System* (IDS) Snort yang digunakan dalam mendeteksi adanya penyusupan maka akan terlihat urutan pada Gambar 4.8 berikut ini.



Gambar 4. 8 Alur Intrusion Detection System (IDS) Snort

Proses pemeriksaan pengambilan *log* yang tersimpan sebagai *alert* pada gambar diatas melukiskan jika ada *traffic* yang melewati server kampus maka *Intrusion Detection System* (IDS) Snort akan mendeteksi cek rule sebagai paket yang normal atau tidak normal dengan menggunakan aturan rule yang sudah ditentukan. Paket yang ditangkap akan diimplementasikan pada saat penelitian berlangsung sehingga akan menghasilkan rekaman trafik yang dianggap melanggar rule yang telah ditetapkan dalam bentuk data file default .pcap.

Data *file log* yang telah berhasil diperiksa selanjutnya akan diambil dalam bentuk *default* p.cap yang terdiri dari beberapa *file log* yang merupakan format paket yang ditangkap setelah memasukkan parameter tertentu pada IDS. Setelah itu, semua data file log dianalsisis menggunakan Wireshark, maka akan dapat dilihat urutan waktunya (*timestamp*).

c. **Tahap Analisis (*Analysis*)**

Pada tahap analisis file log telah diperiksa akan diselidiki lebih dalam. File log yang sudah digabungkan menjadi satu dapat dipakai untuk mengetahui perubahan pada jaringan dan untuk melihat *timestamp*. Data log tersebut diperiksa kembali untuk melihat jenis data apa saja yang berhasil ditangkap dan juga untuk mengetahui protocol apa saja yang banyak digunakan.

Dari aturan rule dibawah ini untuk mendeteksi serangan *flooding* memiliki banyak aturan diantaranya:

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"DDOS TFN
Probe"; icmp_id:678; itype:8; content:"1234";
reference:arachnids,443; classtype:attempted-recon; sid:221;
rev:4;)
```

Dengan keterangan sebagai berikut:

- `alert` adalah tanda peringatan.
- `icmp` adalah jenis protocol transport.
- `$EXTERNAL_NET any` adalah host asal yang melewati port manapun.
- `->` adalah aliran dari host asal ke host tujuan.
- `$HOME_NET any` adalah host tujuan yang melewati port manapun.
- `(msg:"DDOS TFN Probe"; icmp_id:678;`adalah pesan yang akan dikirimkan jika suatu event terjadi.
- `itype:8;` adalah jenis tipe.
- `content:"1234";` adalah konten tipe spesifik yang dicari.
- `reference:arachnids,443;` adalah referensi ke system pengidentifikasi serangan external.
- `classtype:attempted-recon;` adalah percobaan penyadapan informasi.
- `sid:221;` adalah id aturan snort.
- `rev:4;` adalah revisi aturan yang ke 4.

Aturan berikutnya:

```
alert tcp $HOME_NET any <> $EXTERNAL_NET any (msg:"DDOS shaft
synflood"; flow:stateless; flags:S,12; seq:674711609;
reference:arachnids,253; reference:cve,2000-0138;
classtype:attempted-dos; sid:241; rev:10;)
```

Dengan keterangan sebagai berikut:

- `alert` adalah tanda peringatan.
- `icmp` adalah jenis protocol transport.
- `$EXTERNAL_NET any` adalah host asal yang melewati port manapun.
- `<>` adalah aliran dari host host yang masuk.
- `$HOME_NET any` adalah host tujuan yang melewati port manapun.
- `(msg:"DDOS shaft synflood";`
  `flow:stateless;`
  `flags:S,12;`
  adalah pesan yang akan dikirimkan jika suatu event terjadi.
- `classtype:attempted-dos;` adalah jenis percobaan serangan Denial of Service (DOS).

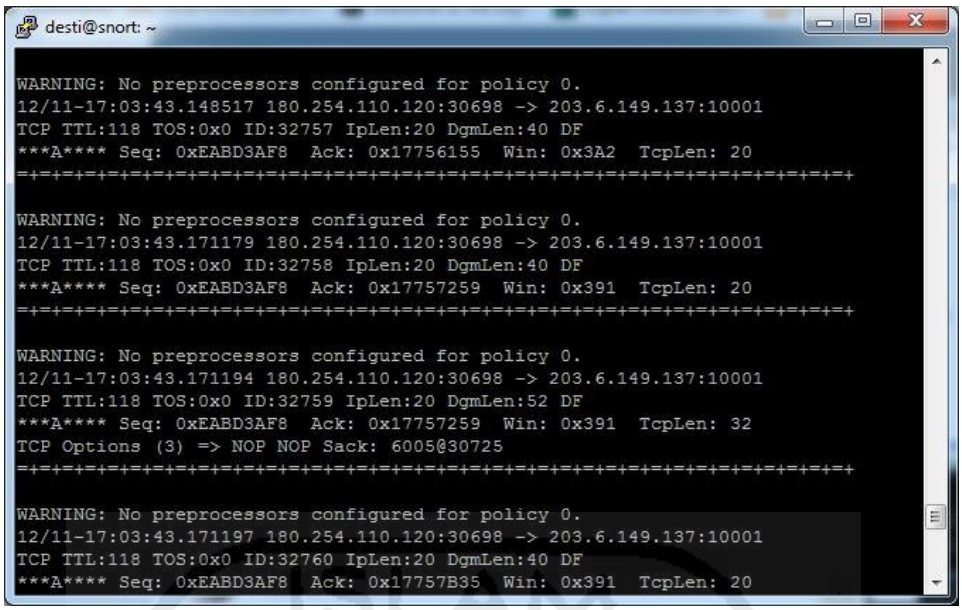
- `reference:arachnids,2533;` adalah referensi ke system pengidentifikasi serangan external.
- `sid:241;` adalah id aturan snort.
- `rev:10;` adalah revisi aturan yang ke 10.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"DDOS
Trin00 Attacker to Master default mdie password";
flow:established,to_server; content:"killme"; classtype:bad-
unknown; sid:235; rev:2;)
```

Dengan keterangan sebagai berikut:

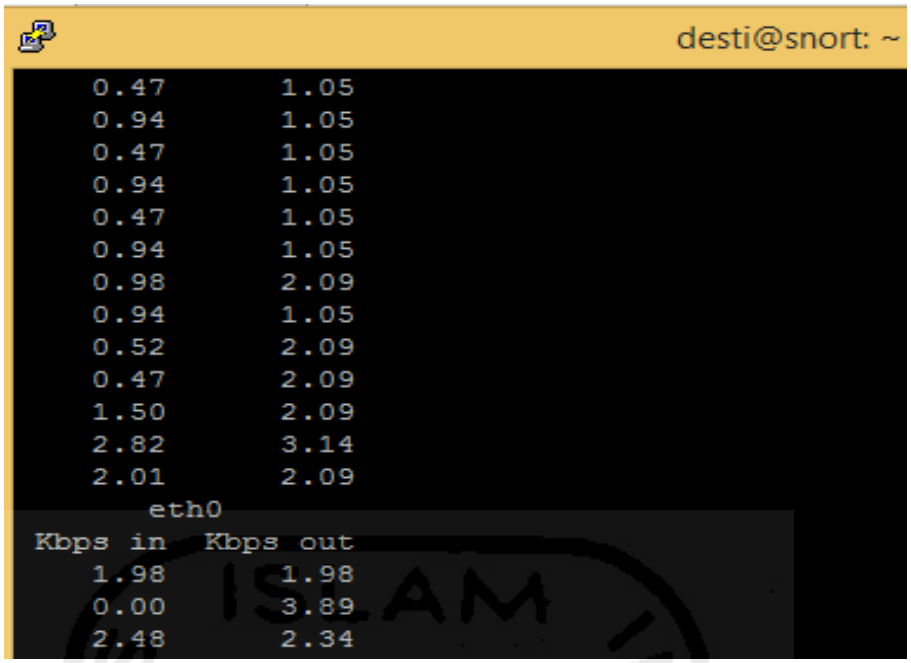
- `alert` adalah tanda peringatan.
- `tcp` adalah jenis protokol transport.
- `$EXTERNAL_NET any` adalah host asal yang melewati port manapun.
- `->` adalah aliran dari host asal ke host tujuan.
- `$HOME_NET 27665 (msg:"DDOS Trin00 Attacker to Master default mdie password";` adalah pesan yang diterima apabila terjadi sebuah event.
- `flow:established,to_server;` adalah koneksi TCP yang dibentuk dari client ke server.
- `content:"killme";` adalah konten spesifikasi yang dicari.
- `classtype:bad-unknown;` adalah trafik yang jelek atau rusak.
- `sid:235;` adalah id aturan snort.
- `rev:2;)`adalah refisi aturan yang ke 2.

Dari simulasi serangan yang telah dikirimkan dalam suatu jaringan akan terlihat *interface* trafik pada *Intrusion Detection System* (IDS) Snort menggunakan rule yang telah di pasang. Deteksi simulasi serangan *flooding* dapat dilihat pada Gambar 4.9 dibawah ini:



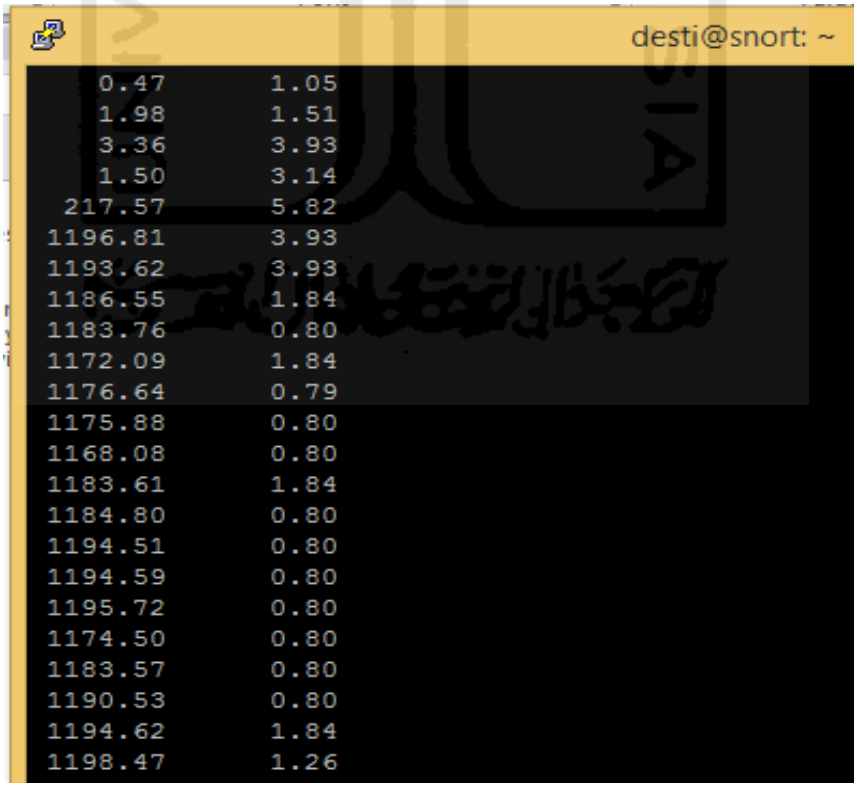
Gambar 4. 9 Interface Trafik Intrusion Detection System (IDS) Snort

Pengiriman *request* data berdampak pada turun dan naiknya aktifitas *traffic* selama penggunaanya. Sedangkan pada jam-jam sibuk, *traffic* suatu data akan sangat padat sehingga *traffic* data tersebut akan sangat mengganggu. Baik berupa data yang dikirim maupun data yang akan datang akan mengalami antrian yang mengakibatkan kelambatan dalam pengiriman maupun penerimaan data. Dengan kata lain, adanya serangan *flooding* dapat berdampak pada gangguan akses data yang digunakan oleh aktifitas internet web server dalam suatu *environment.* Di lain waktu data-data yang berada didalam *traffic* merupakan data yang tidak perlu. Data-data tersebut memang sengaja dikirim oleh seseorang untuk merusak jaringan data yang ada. Pengiriman data tersebut mengakibatkan kerugian lain. Sehingga akan terlihat *traffic* yang ada pada *Intrusion Detection System* (IDS) Snort meningkat dari *range* ukuran *kilobyte per second* (kbps) sampai rentang ukuran *megabyte per second* (mbps) tergantung dengan banyaknya paket yang di *capture*. Dibawah Gambar 4.9 merupakan *capture traffic* normal yang tidak memenuhi aturan *rule* sebagai serangan:



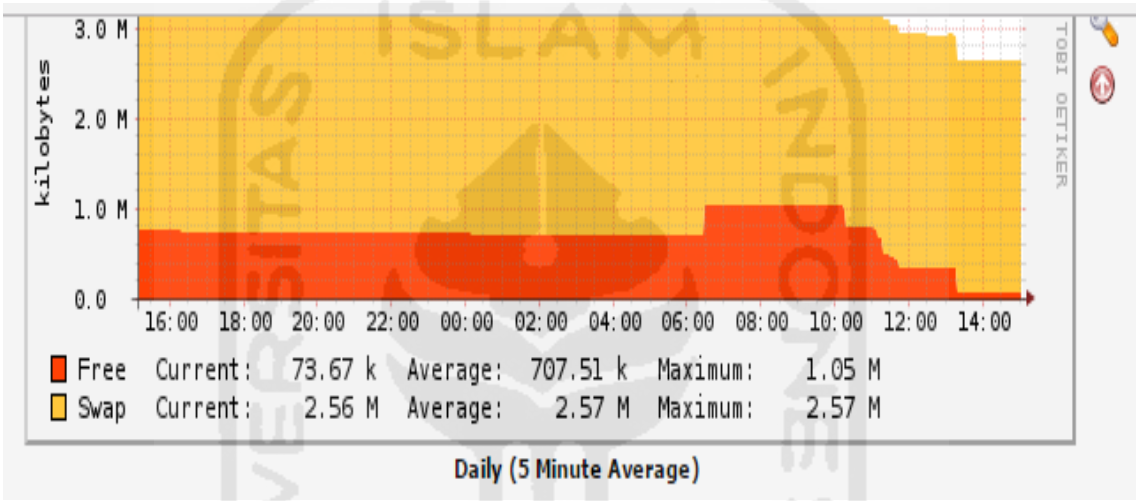
Gambar 4. 10 Traffic Normal

Gambar 4.10 diatas terlihat paket yang normal. Maka, setelah dikirim paket menggunakan tool LOIC trafik akan mengalami peningkatan sebagai serangan yang sesuai dengan rule yang telah ditetapkan.



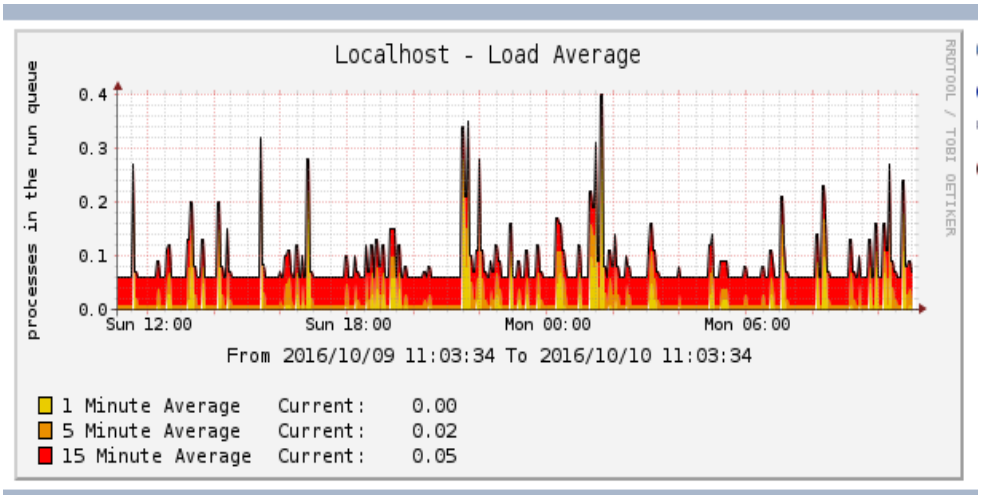
Gambar 4. 11 Trafik Serangan

Pada Gambar 4.11 menunjukan trafik yang mengalami peningkatan karena serangan paket yang dikirim oleh *attacker*. Dari *Intrusion Detection System* (IDS) Snort tersebut juga terlihat dari *graph* yang menunjukan dampak aktifitas dari masing-masing pemakaian pengiriman data yang meningkat dalam interval waktu setiap 5 menit yang ditandai dengan skema warna merah sebagai sisa memory yang tidak terpakai, kemudian warna kuning pada saat penggunaan data penggunanaan, atau biasa disebut dengan *log file* yang dapat di pantau pada Gambar 4.12.



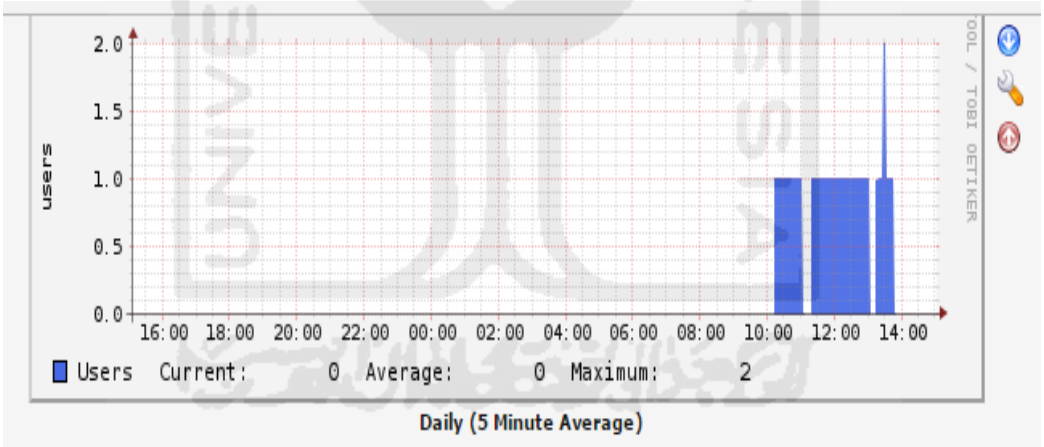
Gambar 4. 12 Memory Usage

Pada Gambar 4.12 *free current* mencapai ukuran 73,67 k dengan *average* 707,51 k pada batas *maximum* 1,05 m, selanjutnya *swap current* mencapai 2,56 m dengan *average* 2,57 m dan batas *maximum* 2,57 m. Penggunaan *memory usage* yang dapat dicek pada graph dapat di lihat pada gambar 4.13 apabila sedang terjadi peningkatan pada gambar 4.13 *load average* dalam waktu 15 menit kedepan, saat dilakukan proses simulasi serangan proses yang berjalan pada *load everage* ditandai dengan masing-masing warna, untuk skema warna kuning pemakaian data dalam waktu 1 menit, warna orange penggunaan pada interval 5 menit, selanjutnya warna merah penggunaan data pada interval 15 menit kedepan.



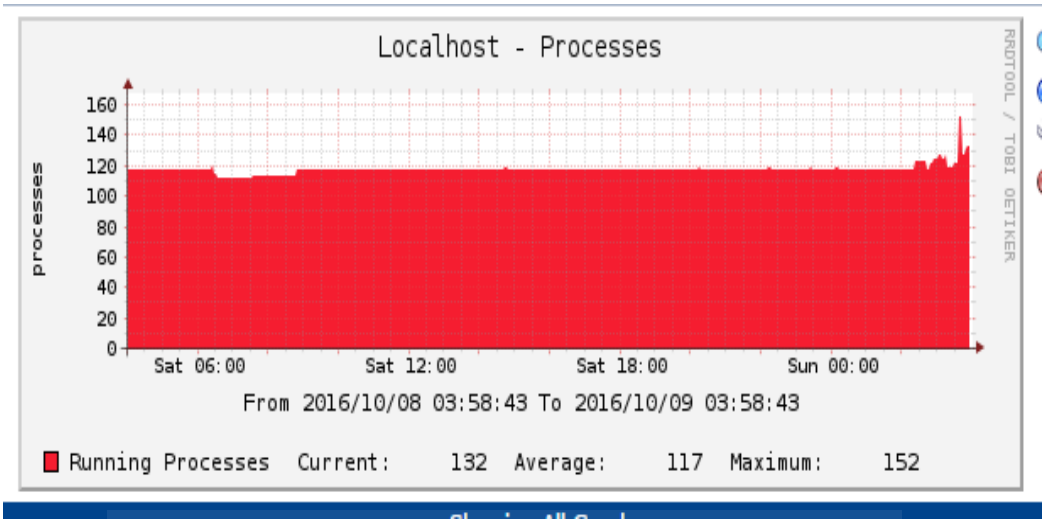
Gambar 4. 13 Memory Usage

Selanjutnya pada Gambar 4.13 *logged in users* ditunjukkan dengan skema warna biru yang meningkat tinggi karena kiriman paket yang banyak, serta proses running mencapai batas maximum 2 m pada pukul 14.00.



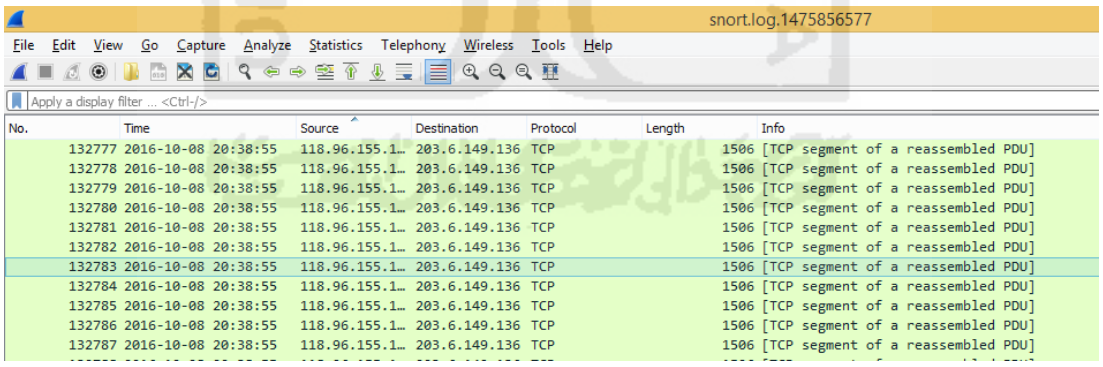
Gambar 4. 14 logged In Users

Selanjutnya Gambar 4.14 pada *running process* pada *traffic* meningkat mencapai *running process current* 132 dengan *average* 117 dan *maximum* 152 *user* per 5 menit dari tanggal 8 Oktober 2016 sampai tanggal 9 Oktober 2016.



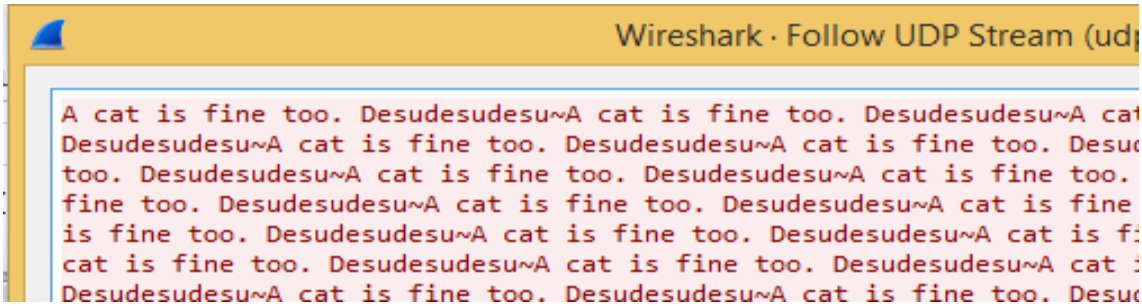
Gambar 4. 15 Runing Processes

Setelah melihat grafik yang masuk maka dapat diambil *file log* secara *real time* sehingga dapat dianalisis menggunakan Wireshark dalam mencari bukti penyerang yang mengirimkan paket *flooding* ke sever kampus. Dalam proses analisis aktivitas illegal di dalam jaringan, Wireshark mampu melihat atau menganalisis paket secara *offline* seperti pada Gambar 4.16 dengan menggunakan bantuan filter berdasarkan IP *address* (ip.scr). *Filter* yang dilakukan terhadap ip penyerang sebagai berikut:



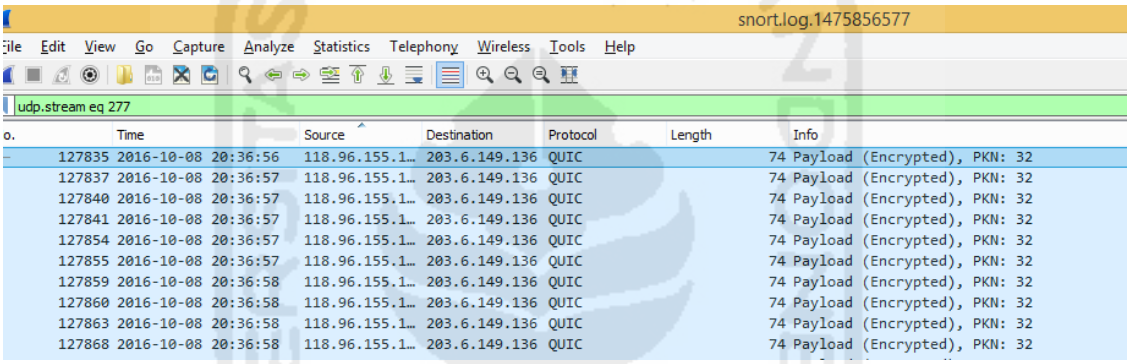
Gambar 4. 16 Filter ip.src

IP address 118.96.155.120 melakukan serangan *flooding* kemudian dapat menganalisis dengan filter ip.src= =118.96.155.120, pilih salah satu baris untuk melakukan analisis, kemudian klik kanan Follow UDP Stream pada Gambar 4.17 kemudian *clouse*.



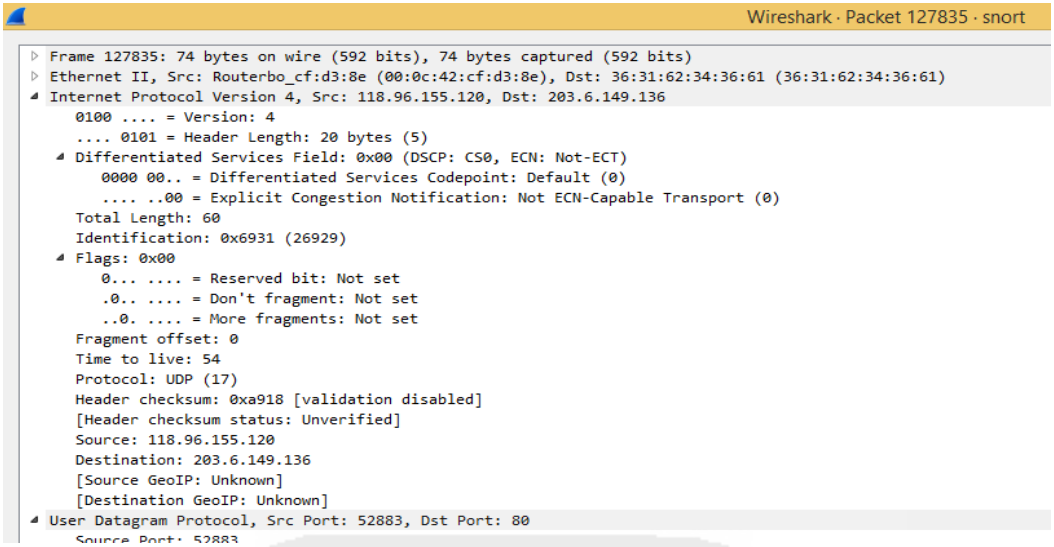
Gambar 4. 17 Follow UDP

Setelah memlilih *clouse* maka akan muncul Gambar 4.18 dibawah ini berupa *timestamp, source* IP, *destination* IP, *protocol,* dan *payload*.



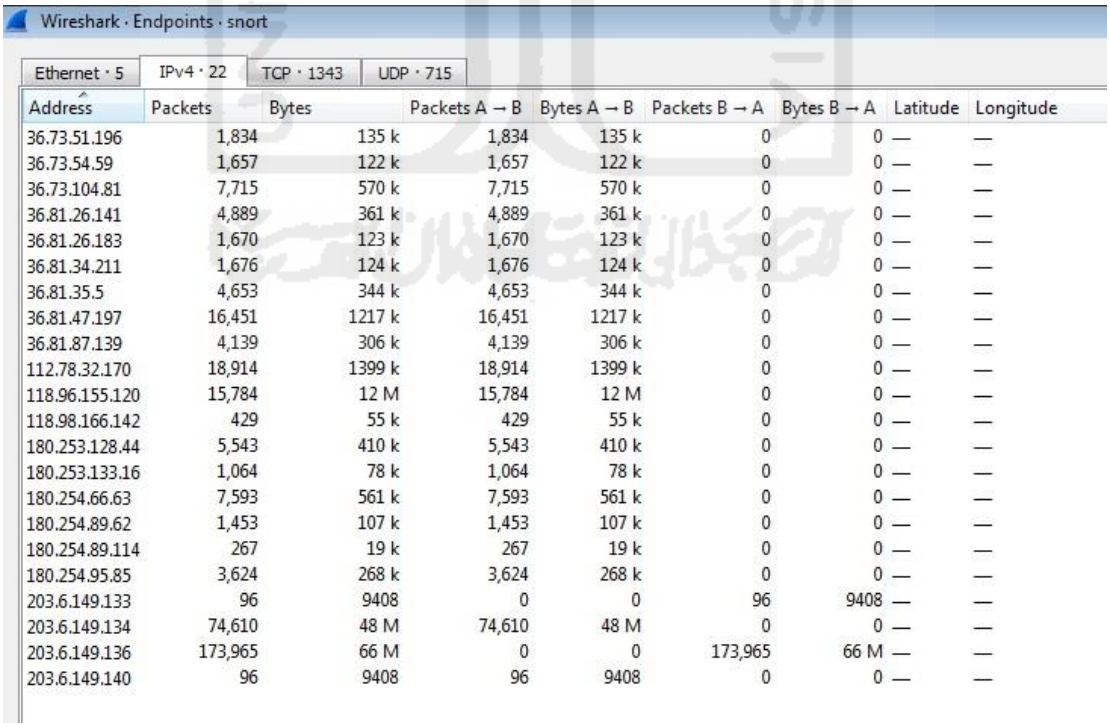
Gambar 4. 18 Hasil follow UDP

Dari kumpulan baris tersebut dapat dipilih salah satu baris untuk melakukan analisis pada setiap bagian *frame* yang mewakili sebuah *packet frame* pada serangan *flooding* dari IP address 118.96.155.120 memiliki panjang (length) berkisar pada angka 70-an Bytes (74 Bytes). Pada bagian Internet Protocol Version 4, source IP terbaca sebagai 118.96.55.120 dan destination IP address terbaca 203.6.149.136 dengan *header length* 20 *Bytes* dan total *length* 60. Pada bagian *User Datagram Protocol*, *source port* terbaca sebagai 52883 dan *destination port* terbaca sebagai 80. Apabila *filter* dikembalikan pada ip.src== 118.96.55.120 dan diteliti pada *frame* yang lain, *source port*-nya berubah-rubah, namun tetap dalam rentang yang besar (port 51000-64000). Hasil *capture frame* analisis IP *address* 118.96.55.120 pada Gambar 4.19.



Gambar 4. 19 Hasil frame

Selain itu, analisis dilanjutkan dengan modul statistik *endpoint* pada Wireshark yang digunakan untuk mengumpulkan total paket serangan yang terdapat pada *log file Intrusion Detection System* (IDS) Snort selama simulasi serangan berlangsung. Pada Gambar 4.20 dibawah ini menjelaskan bahwa IP address memiliki beban yang berbeda pada setiap paket dan kecepatan yang berbeda pada setiap bytes nya.


Wireshark · Endpoints · snort

Ethernet · 5 | IPv4 · 22 | TCP · 1343 | UDP · 715

Address	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Latitude	Longitude
36.73.51.196	1,834	135 k	1,834	135 k	0	0	—	—
36.73.54.59	1,657	122 k	1,657	122 k	0	0	—	—
36.73.104.81	7,715	570 k	7,715	570 k	0	0	—	—
36.81.26.141	4,889	361 k	4,889	361 k	0	0	—	—
36.81.26.183	1,670	123 k	1,670	123 k	0	0	—	—
36.81.34.211	1,676	124 k	1,676	124 k	0	0	—	—
36.81.35.5	4,653	344 k	4,653	344 k	0	0	—	—
36.81.47.197	16,451	1217 k	16,451	1217 k	0	0	—	—
36.81.87.139	4,139	306 k	4,139	306 k	0	0	—	—
112.78.32.170	18,914	1399 k	18,914	1399 k	0	0	—	—
118.96.155.120	15,784	12 M	15,784	12 M	0	0	—	—
118.98.166.142	429	55 k	429	55 k	0	0	—	—
180.253.128.44	5,543	410 k	5,543	410 k	0	0	—	—
180.253.133.16	1,064	78 k	1,064	78 k	0	0	—	—
180.254.66.63	7,593	561 k	7,593	561 k	0	0	—	—
180.254.89.62	1,453	107 k	1,453	107 k	0	0	—	—
180.254.89.114	267	19 k	267	19 k	0	0	—	—
180.254.95.85	3,624	268 k	3,624	268 k	0	0	—	—
203.6.149.133	96	9408	0	0	96	9408	—	—
203.6.149.134	74,610	48 M	74,610	48 M	0	0	—	—
203.6.149.136	173,965	66 M	0	0	173,965	66 M	—	—
203.6.149.140	96	9408	96	9408	0	0	—	—

Gambar 4. 20 Statistik Endpoint Snort

Kemudian, pada statistic *endpoint* Snort terdapat jumlah serangan *flooding* dalam bentuk TCP sejumlah 1343 serangan dan UDP sebanyak 715 serangan.

Selanjutnya dilakukan analisis IP *address* penyerang lainnya seperti langkah-langkah diatas maka dapat dikumpulkan data analisis untuk barang bukti pada tabel 4.1. Terdapat 15 IP *address* penyerang yang melakukan serangan terhadap web server Universitas Muhammadiyah Magelang.

d. Laporan

Berdasarkan observasi yang dikumpulkan dan diurutkan dari *literature*, dan *eksperiment* yang diimplementasikan pada penelitian ini, membuktikan bahwa serangan *flooding* pada web server merupakan serangan yang memiliki tingkat tinggi di lingkungan Universitas Muhammadiyah Magelang yang memerlukan tindakan cepat untuk deteksi serangan khususnya serangan *flooding*. Selain mendeteksi serangan diperlukan proses investigasi forensic untuk menemukan barang bukti digital tindakan illegal pada web server.

Tabel 4. 1 Prioritas klasifikasi serangan

No.	Timestamp	Source	Dest. IP	Protokol	Source Port	Dest. Port	Payload / Pesan
1	7/10/2016 17:26	203.6.149.140	203.x.x.x	ICMP	-	-	40ddf957603e0e006e69746f72696e6763616374692d6d6f…
2	7/10/2016 16:32	112.78.32.170	203.x.x.x	UDP	52658	80	69732066696e6520746f6f2e20446573756465737375646573…
3	8/10/2016 19:28	36.73.51.196	203.x.x.x	UDP	58894	80	69732066696e6520746f6f2e20446573756465737375646573…
4	8/10/2016 20:36	118.96.155.120	203.x.x.x	UDP	52882	80	69732066696e6520746f6f2e20446573756465737375646573…
5	8/10/2016 19:45	180.253.133.16	203.x.x.x	UDP	60052	80	69732066696e6520746f6f2e20446573756465737375646573…
6	8/10/2016 20:26	180.253.128.44	203.x.x.x	UDP	63749	80	69732066696e6520746f6f2e20446573756465737375646573…
7	8/10/2016 20:09	180.254.95.85	203.x.x.x	UDP	53820	80	69732066696e6520746f6f2e20446573756465737375646573…
8	8/10/2016 20:15	180.254.89.62	203.x.x.x	UDP	61246	80	69732066696e6520746f6f2e20446573756465737375646573…

Tabel 4. 2 Con't Prioritas klasifikasi serangan

No.	Timestamp	Source	Dest. IP	Protokol	Source Port	Dest. Port	Payload / Pesan
9	8/10/2016 20:51	180.254.66.63	203.x.x.x	UDP	54948	80	69732066696e6520746f6f2e20446 573756465737564657 3...
10	8/10/2016 20:07	36.73.104.81	203.x.x.x	UDP	53817	80	69732066696e6520746f6f2e20446 573756465737564657 3...
11	8/10/2016 20:08	36.73.54.59	203.x.x.x	UDP	53814	80	69732066696e6520746f6f2e20446 573756465737564657 3...
12	8/10/2016 20:20	36.81.87.139	203.x.x.x	UDP	63748	80	69732066696e6520746f6f2e20446 573756465737564657 3...
13	8/10/2016 20:25	36.81.26.141	203.x.x.x	UDP	63756	80	69732066696e6520746f6f2e20446 573756465737564657 3...
14	10/10/2016 6:34	36.81.47.197	203.x.x.x	UDP	55291	80	69732066696e6520746f6f2e204465 737564657375646573...
15	10/10/2016 6:34	36.81.35.5	203.x.x.x	UDP	56328	80	69732066696e6520746f6f2e204465 737564657375646573...

Pada tabel 4.1 terdapat sejumlah 15 IP *address* yang mencoba melakukan serangan, selain itu terdapat *timestamp* yang menjelaskan kapan terjadinya serangan tersebut berlangsung. Selanjutnya *destination* IP *address* merupakan IP yang dijadikan target *attacker*, kemudian protokol merupakan pola serangan yang dilakukan oleh *attacker*. Selain itu, terdapat *port* yang diserang dan pesan/*payload* yang dikirim oleh *attacker* ketika melakukan percobaan serangan.

Berdasarkan informasi yang dikumpulkan dari litelature-litelature dan eksperimen yang diimplementasikan pada penelitian forensik jaringan ini, membuktikan bahwa hasil dari eksperimen dan simulasi deteksi serangan *flooding* pada web server di lingkungan Universitas Muhammadiyah Magelang telah berhasil dan didapatkan barang bukti digital forensik berupa *file log* Snort *Intrusion Detection System* (IDS) sebanyak 15 IP *address* yang mencoba melakukan serangan *flooding* selama penelitian berlangsung.

Bab 5 Kesimpulan dan Saran

Pada bagian ini menjelaskan kesimpulan dari hasil penelitian yang telah dilakukan berdasarkan tujuan dan perumusan masalah penelitian, yaitu: 1) apakah pemasangan Snort mampu meberikan informasi dalam mendeteksi serangan *flooding,* dan 2) bagaimana hasil *file log* Snort dalam menemukan barang bukti digital forensik.

5.1 Kesimpulan

Kesimpulan yang telah didapatkan selama proses penelitian dalam melakukan deteksi serangan *flooding* pada web server menyimpulkan bahwa:

1. Pengimplementasian *Intrusion Detection System* (IDS) Snort pada web server dilingkungan Universitas Muhammadiyah Magelang dapat digunakan untuk membantu memberikan informasi terkait deteksi adanya serangan *flooding* dengan memanfaatkan *rule* khusus *flooding* yang diterapkan pada *Intrusion Detection System* (IDS) Snort.
2. *File log* didapatkan dari pengimplementasian Snort guna kebutuhan analisis aktivitas tindakan illegal yang terjadi pada lingkungan web server Universitas Muhammadiyah Magelang, berdasarkan analisis file log terdapat sebanyak 15 IP *address* penyerang yang melakukan tindakan illegal dengan *frekuensi* penerimaan data *timestamp, destination port,* dan pesan/*payload*. Dari hasil analisis yang dijabarkan pada BAB IV tersebut menunjukkan bahwa ada serangan *flooding*, serangan yang mengirimkan banyak data pada target, sehingga dapat mengintervensi serta merusak sumber daya khusunya IT yang ada pada Universitas Muhammadiyah Magelang.

Berdasarkan penelitian yang telah dilakukan, *Intrusion Detection System* (IDS) Snort yang diimplementasikan pada *environment web server* di lingkungan Universitas Muhammadiyah Magelang dapat memberikan Informasi terkait deteksi serangan, khususnya serangan *flooding*.

5.2 Saran

Saran yang dapat digunakan untuk penelitian berikutnya terkait dengan Intrusion Detection System (IDS) adalah:

a. Saran yang dapat diberikan untuk keperluan penelitian selanjutnya adalah pengembangan system deteksi menggunakan Intrusion Detection System (IDS) Snort dalam mendeteksi serangan lain, seperti: SQL *Injectin*, *Brute Force*, dll.
b. Melakukan pengembangan unuk melakukan implementasi Snort pada system jaringan workstation yang lebih besar seperti multi WAN.

Daftar Pustaka

Ah, M. Z. (2009). Fakultas teknik program teknik komputer depok desember 2009.

Al-Dalky, R. (2014). *Accelerating Snort NIDS Using NetFPGA-bassed Bloom Filter*. Khalifa University of Science. *International Journal of Computer Science and Security. Retrived from:*

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6906470&queryText=snort&newsearch=true

Al-Azhar, M. N. (2012). Digital Forensic : Panduan Praktis Investigasi Komputer. Salemba Infotek.

Cahyanto, T. A., & Prayudi, Y. (2014). Investigasi Forensika Pada Log Web Server untuk Menemukan Bukti Digital Terkait dengan Serangan Menggunakan Metode Hidden Markov Models. *Snati*, 15–19.

Charles, T., & Pollock, M. (2015). Digital forensic investigations at universities in South Africa. *2015 Second International Conference on Information Security and Cyber Forensics (InfoSec)*, 53–58. https://doi.org/10.1109/InfoSec.2015.7435506

Design & Deployment Of Testbed Based On ICMPv6 Flooding Attack. (2014), *64*(3), 795–801.

Franke, K. (n.d.). Computa (onal Forensics : Trends and Challenges in Applying Ar (ficial Intelligence Methodologies to Digital Forensics Impact of Computa * onal Science, 1–66.

Guide to Integrating Forensic Techniques into Incident Response. (n.d.).

Indra, A. (2010). Intrusion Detection Tools and Techniques – A Survey, *2*(6).

Introduction to Snort A . Sniffer Mode. (n.d.), 1–11.

Iswardani, A., & Riadi, I. (2016). Denial Of Service Log Analysis Using Density K-Means Method, *83*(2), 299–302.

Khamphakdee, N. (2014). Improving Intrusion Detection System Based on Snort Rules for Network Probe Attack Detection, 69–74.

Lanke, N. M., & Jacob, C. H. R. (2014). Detection of DDOS Attacks Using Snort Detection, *2*(9), 13–17.

Lipeng, D., Xingyuan, C., Huilin, T., & Wang, S. (2013). A Generation Framework of Multiple

Evasions on IDS. *2013 Third International Conference on Instrumentation, Measurement, Computer, Communication and Control*, 549–552. https://doi.org/10.1109/IMCCC.2013.124

McAfee Labs Threats Report. (2016), (March).

Nguyen, K., Tran, D., Ma, W., & Sharma, D. (2014). An Approach to Detect Network Attacks Applied for Network Forensics, 655–660.

Server, K., & Aktivitas, D. (2013). Implementasi honeypot untuk meningkatkan sistem keamanan server dari aktivitas serangan.

Shah, V., & Aggarwal, A. K. (2015). Heterogeneous fusion of IDS alerts for detecting DOS attacks. *Proceedings - 1st International Conference on Computing, Communication, Control and Automation, ICCUBEA 2015*, 153–158. https://doi.org/10.1109/ICCUBEA.2015.35

Sindhu, K. K., & Meshram, B. B. (2012). Digital Forensics and Cyber Crime Datamining, *2012*(July), 196–201.

Snort.org. 2016. Retrived form, http//:Snort.org.download.

Stiawan, D., Yaseen, A. L. A., Shakhatreh, I., Idris, M. Y., Bakar, K. A. B. U., & Abdullah, A. H. (2012). Intrusion Prevention System: A Survey, *40*(1), 44–54.

Studi, P., Informatika, T., Sains, F., Teknologi, D. A. N., & Kalijaga, U. I. N. S. (2015). Investigasi Forensik Jaringan Dari Serangan DDoS Menggunakan Metode Naive Bayes.

Suteva, N., Mileva, A., & Loleski, M. (2014). Computer Forensic Analisys of Some Web Attacks, 42–47.

Utami Putri, R., & Istiyanto, J. E. (2012). Analisis Forensik Jaringan Studi Kasus Serangan SQL Injection pada Server Universitas Gadjah Mada. *International Journal of Computer Science and Security*, *6*(2). Retrieved from http://journal.ugm.ac.id/index.php/ijccs/article/view/2157

Wireshark.org.2016. Retrived from, http//:Wireshark.org.download.

Lampiran

Rule flooding attack yang digunakan untuk deteksi serangan sebagai berikut:

```
#

#----------

# DOS RULES

#----------

#

alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"DOS Jolt attack";
dsize:408; fragbits:M; reference:cve,1999-0345; classtype:attempted-dos;
sid:268; rev:4;)

alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"DOS Teardrop attack";
fragbits:M; id:242; reference:bugtraq,124; reference:cve,1999-0015;
reference:nessus,10279; reference:url,www.cert.org/advisories/CA-1997-
28.html; classtype:attempted-dos; sid:270; rev:6;)

alert udp any 19 <> any 7 (msg:"DOS UDP echo+chargen bomb";
reference:cve,1999-0103; reference:cve,1999-0635; classtype:attempted-dos;
sid:271; rev:5;)

alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"DOS IGMP dos attack";
fragbits:M+; ip_proto:2; reference:bugtraq,514; reference:cve,1999-0918;
classtype:attempted-dos; sid:272; rev:9;)

alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"DOS IGMP dos attack";
fragbits:M+; ip_proto:2; reference:bugtraq,514; reference:cve,1999-0918;
classtype:attempted-dos; sid:273; rev:8;)

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"DOS ath"; itype:8;
content:"+++ath"; nocase; reference:arachnids,264; reference:cve,1999-1228;
classtype:attempted-dos; sid:274; rev:5;)

alert tcp $EXTERNAL_NET any <> $HOME_NET any (msg:"DOS NAPTHA";
flow:stateless; flags:S; id:413; seq:6060842; reference:bugtraq,2022;
reference:cve,2000-1039;
reference:url,razor.bindview.com/publish/advisories/adv_NAPTHA.html;
reference:url,www.cert.org/advisories/CA-2000-21.html;
reference:url,www.microsoft.com/technet/security/bulletin/MS00-091.mspx;
classtype:attempted-dos; sid:275; rev:12;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 7070 (msg:"DOS Real Audio Server";
flow:to_server,established; content:"|FF F4 FF FD 06|";
reference:arachnids,411; reference:bugtraq,1288; reference:cve,2000-0474;
classtype:attempted-dos; sid:276; rev:5;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 7070 (msg:"DOS Real Server
template.html"; flow:to_server,established;
content:"/viewsource/template.html?"; nocase; reference:bugtraq,1288;
reference:cve,2000-0474; classtype:attempted-dos; sid:277; rev:5;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 8080 (msg:"DOS Real Server
template.html"; flow:to_server,established;
content:"/viewsource/template.html?"; nocase; reference:bugtraq,1288;
reference:cve,2000-0474; classtype:attempted-dos; sid:278; rev:5;)

alert udp $EXTERNAL_NET any -> $HOME_NET 161 (msg:"DOS Bay/Nortel Nautica
Marlin"; dsize:0; reference:bugtraq,1009; reference:cve,2000-0221;
classtype:attempted-dos; sid:279; rev:4;)

alert udp $EXTERNAL_NET any -> $HOME_NET 9 (msg:"DOS Ascend Route";
content:"NAMENAME"; depth:50; offset:25; reference:arachnids,262;
reference:bugtraq,714; reference:cve,1999-0060; classtype:attempted-dos;
sid:281; rev:5;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 617 (msg:"DOS arkiea backup";
flow:to_server,established; dsize:>1445; reference:arachnids,261;
reference:bugtraq,662; reference:cve,1999-0788; classtype:attempted-dos;
sid:282; rev:8;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 135:139 (msg:"DOS Winnuke attack";
flow:stateless; flags:U+; reference:bugtraq,2010; reference:cve,1999-0153;
classtype:attempted-dos; sid:1257; rev:10;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 3372 (msg:"DOS MSDTC attempt";
flow:to_server,established; dsize:>1023; reference:bugtraq,4006;
reference:cve,2002-0224; reference:nessus,10939; classtype:attempted-dos;
sid:1408; rev:10;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 6004 (msg:"DOS iParty DOS attempt";
flow:to_server,established; content:"|FF FF FF FF FF FF|"; offset:0;
reference:bugtraq,6844; reference:cve,1999-1566; classtype:misc-attack;
sid:1605; rev:6;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 6789:6790 (msg:"DOS DB2 dos
attempt"; flow:to_server,established; dsize:1; reference:bugtraq,3010;
reference:cve,2001-1143; reference:nessus,10871; classtype:denial-of-service;
sid:1641; rev:10;)

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"DOS Cisco attempt";
flow:to_server,established; dsize:1; content:"|13|"; classtype:web-
application-attack; sid:1545; rev:8;)

alert udp $EXTERNAL_NET any -> $HOME_NET 500 (msg:"DOS ISAKMP invalid
identification payload attempt"; content:"|05|"; depth:1; offset:16;
byte_test:2,>,4,30; byte_test:2,<,8,30; reference:bugtraq,10004;
reference:cve,2004-0184; classtype:attempted-dos; sid:2486; rev:5;)
```

```
alert tcp $EXTERNAL_NET any <> $HOME_NET 179 (msg:"DOS BGP spoofed connection
reset attempt"; flow:established; flags:RSF*; threshold:type both,track
by_dst,count 10,seconds 10; reference:bugtraq,10183; reference:cve,2004-0230;
reference:url,www.uniras.gov.uk/vuls/2004/236929/index.htm;
classtype:attempted-dos; sid:2523; rev:7;)

alert udp $EXTERNAL_NET any -> $HOME_NET 2048 (msg:"DOS squid WCCP I_SEE_YOU
message overflow attempt"; content:"|00 00 00 08|"; depth:4;
byte_test:4,>,32,16; reference:cve,CAN-2005-0095; reference:bugtraq,12275;
classtype:attempted-user; sid:3089; rev:1;)
```

Lampiran tabel rule flooding attack:

No.	RULE	KETERANGAN
1	Alert	Tanda peringatan
	Ip	Alamat IP
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Jolt attack"; dsize:408;	Pesan yang akan diterima apabila terjadi sebuah event
	fragbits:M;	Ukuran data
	reference:cve,1999-0345;	Merupakan referensi ke system pengidentifikasian
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:268;	Merupakan id dari aturan snort
	rev:4;)	Refisi aturan snort ke 4
2	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	(msg:"DOS Teardrop attack";	Pesan yang akan diterima apabila terjadi sebuah event
	fragbits:M;	Ukuran data
	id:242;	Merupakan id dari aturan snort
	reference:bugtraq,124; reference:cve,1999-0015; reference:nessus,10279; reference:url,www.cert.org/advisories/CA-1997-28.html;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:270;	Merupakan id dari aturan snort
	rev:6;)	Refisi aturan snort ke 6
3	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	any 19	Host tujuan port 19
	<>	Aliran host
	any 7	Host tujuan port 7
	(msg:"DOS UDP echo+chargen bomb";	Pesan yang akan diterima apabila terjadi sebuah event
	reference:cve,1999-0103; reference:cve,1999-0635;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:271;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
4	Alert	Tanda peringatan
	Ip	Alamat IP
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS IGMP dos attack";	Pesan yang akan diterima apabila terjadi sebuah

		event
	fragbits:M+;	Ukuran data
	ip_proto:2;	IP Protokol 2
	reference:bugtraq,514; reference:cve,1999-0918;	Merupakan referensi ke system pengidentifikasian serangan
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:272;	Merupakan id dari aturan snort
	rev:9;)	Refisi aturan snort ke 9
5	Alert	Tanda peringatan
	Ip	Alamat IP
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS IGMP dos attack"; fragbits:M+;	Pesan yang akan diterima apabila terjadi sebuah event
	ip_proto:2;	IP Protokol 2
	reference:bugtraq,514; reference:cve,1999-0918;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:273;	Merupakan id dari aturan snort
	rev:8;)	Refisi aturan snort ke 8
6	Alert	Tanda peringatan
	Icmp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS ath"; itype:8; content:"+++ath"; nocase;	Pesan yang akan diterima apabila terjadi sebuah event
	reference:arachnids,264;	Merupakan referensi ke sistem pengidentifikasian serangan
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:274;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
7	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	<>	
	(msg:"DOS NAPTHA"; flow:stateless;	Pesan yang akan diterima apabila terjadi sebuah event
	flags:S;	Serangan
	id:413;	Nomor id
	seq:6060842;	Waktu
	reference:bugtraq,2022; reference:cve,2000-1039; reference:url,razor.bindview.com/publish/advisories/adv_NAPTHA.html; reference:url,www.cert.org/advisories/CA-2000-21.html; reference:url,www.microsoft.com/t	Merupakan referensi ke system pengidentifikasian serangan eksternal

	echnet/security/bulletin/MS00-091.mspx;	
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:275;	Merupakan id dari aturan snort
	rev:12;)	Refisi aturan snort ke 12
8	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Real Audio Server";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	content:"|FF F4 FF FD 06|";	Konten spesifik yang dicari
	reference:arachnids,411; reference:bugtraq,1288; reference:cve,2000-0474;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:276;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
9	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Real Server template.html";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	content:"/viewsource/template.html?";	Kontes spesifik yang dicari
	nocase;	Tidak ada aturan
	reference:bugtraq,1288; reference:cve,2000-0474;	Merupakan referensi ke system pengidentifikasian serangan
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:277;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
10	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Real Server template.html";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	content:"/viewsource/template.html?";	Konten spesifik yang dicari
	nocase;	Tidak ada aturan
	reference:bugtraq,1288;	Merupakan referensi ke system pengidentifikasian

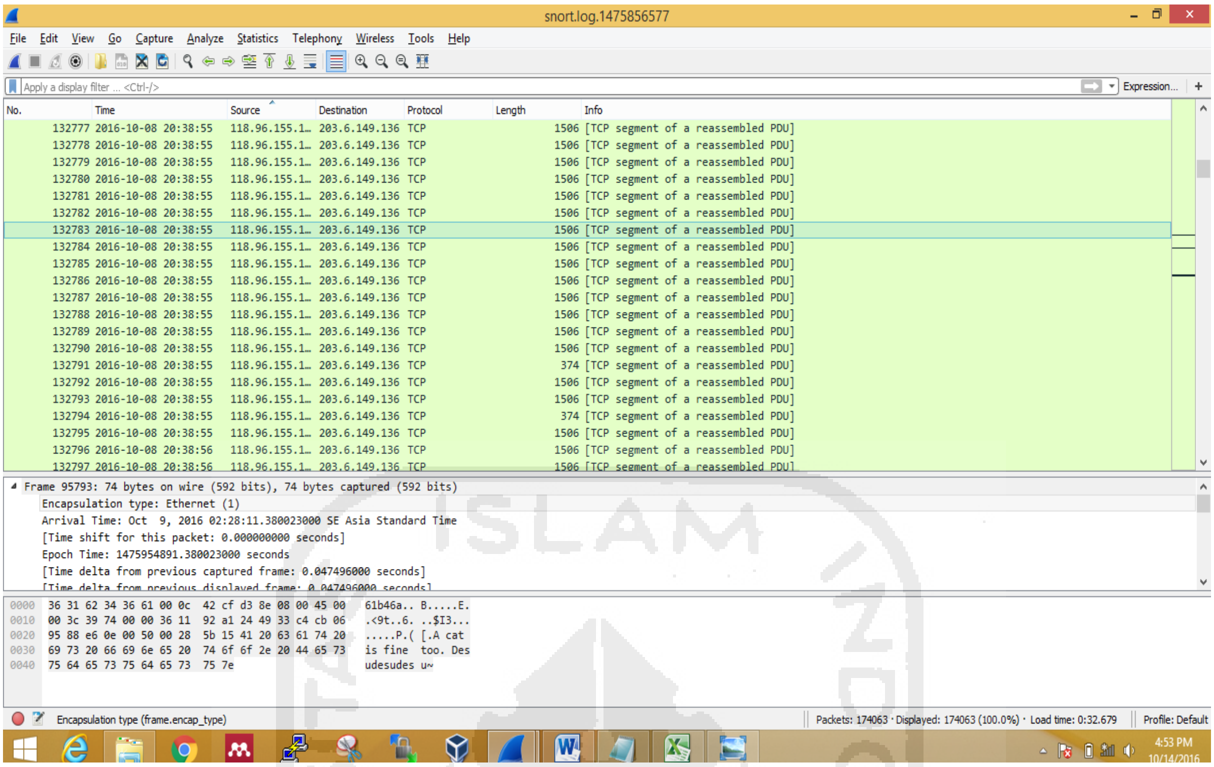
	reference:cve,2000-0474;	serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:278;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
11	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Bay/Nortel Nautica Marlin";	Pesan yang akan diterima apabila terjadi sebuah event
	dsize:0;	Ukuran data
	reference:bugtraq,1009; reference:cve,2000-0221;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:279;	Merupakan id dari aturan snort
	rev:4;)	Refisi aturan snort ke 4
12	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Ascend Route"; content:"NAMENAME";	Pesan yang akan diterima apabila terjadi sebuah event
	depth:50;	Untuk mencari pola yang sesuai dengan konten pada 50 byte pertama pada payload
	offset:25;	Akhir data 25
	reference:arachnids,262; reference:bugtraq,714; reference:cve,1999-0060;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:281;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
13	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS arkiea backup";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	dsize:>1445;	Ukuran host
	reference:arachnids,261; reference:bugtraq,662; reference:cve,1999-0788; classtype:attempted-dos;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:282;	Merupakan id dari aturan snort
	rev:8;)	Refisi aturan snort ke 8

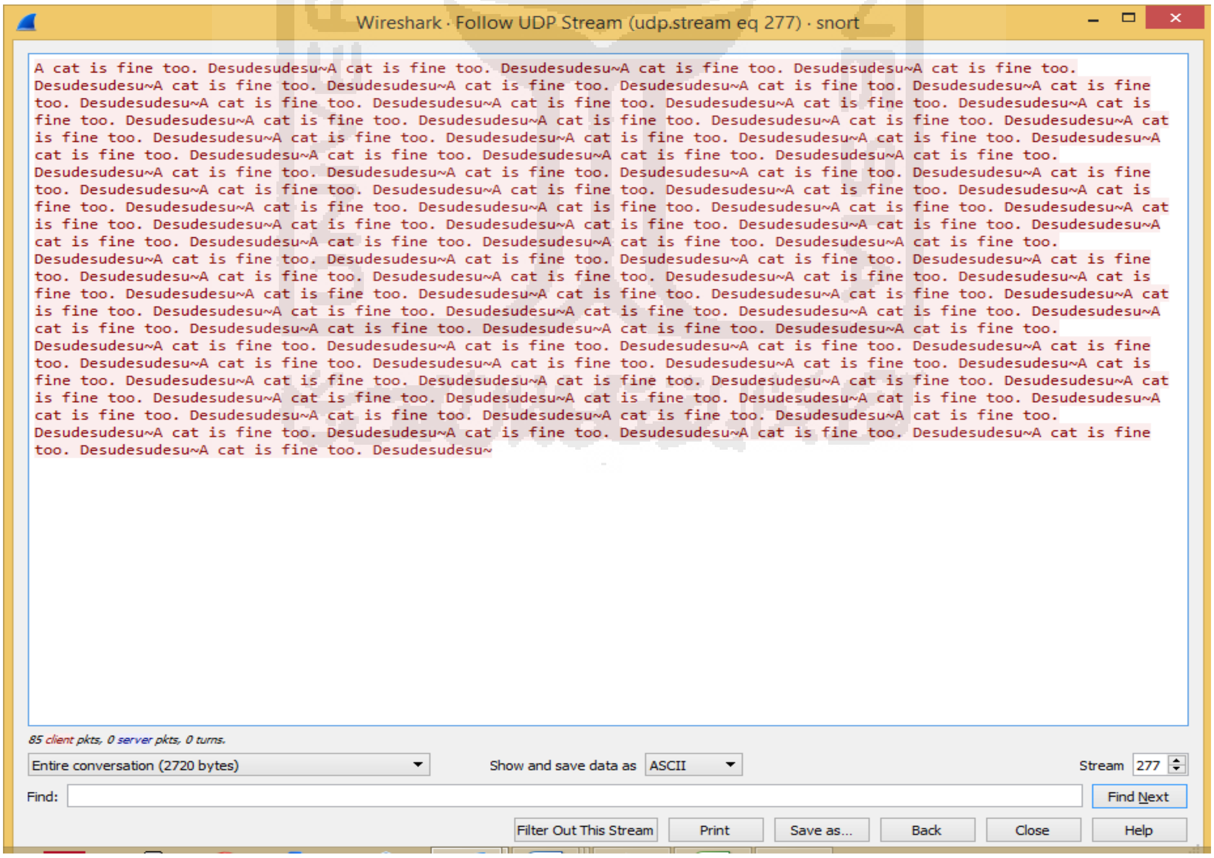
14	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Winnuke attack";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:stateless;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	flags:U+;	Nama host
	reference:bugtraq,2010; reference:cve,1999-0153;	Merupakan referensi ke sistem pengidentifikasian serangan
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:1257;	Merupakan id dari aturan snort
	rev:10;)	Refisi aturan snort ke 10
15	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS MSDTC attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	dsize:>1023;	Ukuran host
	reference:bugtraq,4006; reference:cve,2002-0224; reference:nessus,10939;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:1408;	Merupakan id dari aturan snort
	rev:10;)	Refisi aturan snort ke 10
16	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS iParty DOS attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	content:"|FF FF FF FF FF FF|";	Isi spesifik konten
	offset:0;	Akhir data
	reference:bugtraq,6844; reference:cve,1999-1566;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:misc-attack;	Percobaan Denial of Service
	sid:1605;	Merupakan id dari aturan snort
	rev:6;)	Refisi aturan snort ke 6
17	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun

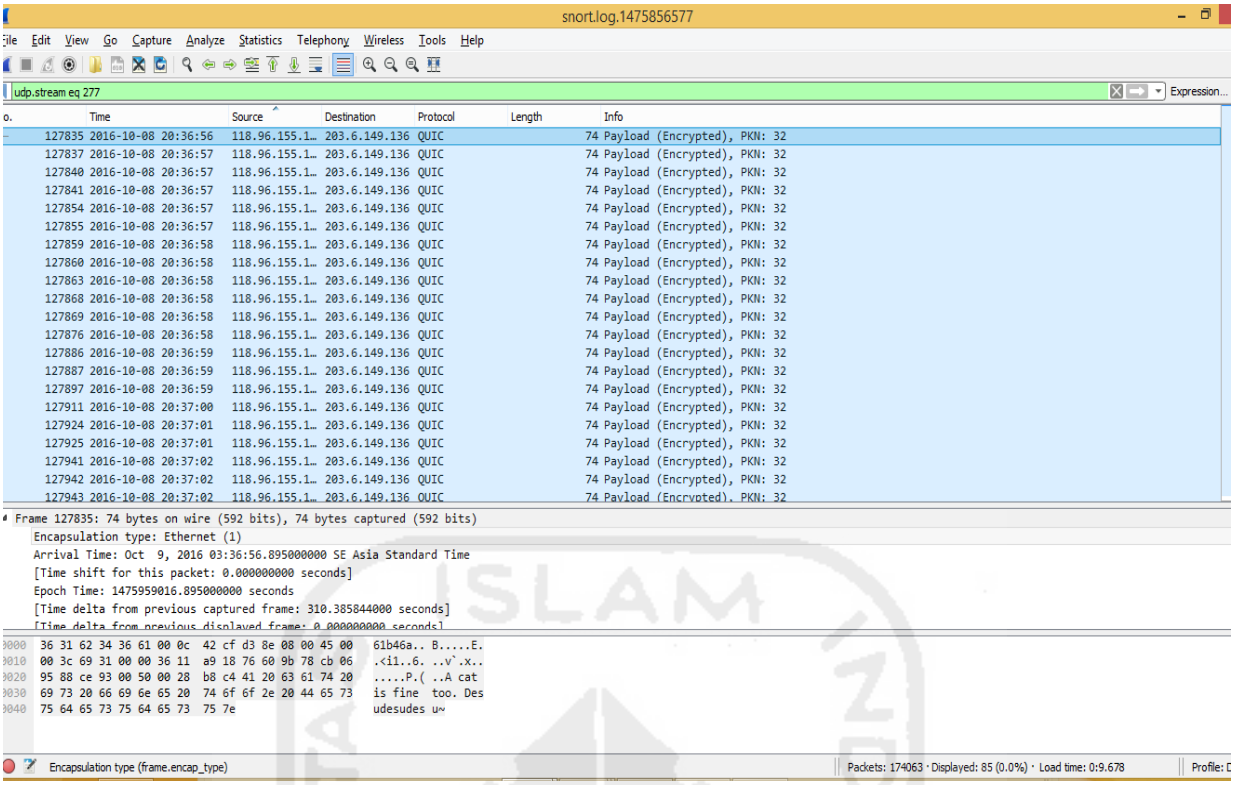
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS DB2 dos attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	dsize:1;	Ukuran konten
	reference:bugtraq,3010; reference:cve,2001-1143; reference:nessus,10871;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:denial-of-service;	Percobaan Denial of Service
	sid:1641;	Merupakan id dari aturan snort
	rev:10;)	Refisi aturan snort ke 10
18	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS Cisco attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	flow:to_server,established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	dsize:1;	Ukuran size
	content:"|13|";	Isi spesifikasi konten
	classtype:web-application-attack;	Serangan aplikasi web
	sid:1545;	Merupakan id dari aturan snort
	rev:8;)	Refisi aturan snort ke 8
19	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS ISAKMP invalid identification payload attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	content:"|05|";	Isi spesifikasi konten yang dicari
	depth:1;	Terbuka konten
	offset:16;	Akhir host 16
	byte_test:2,>,4,30;	Ukuran konten
	byte_test:2,<,8,30;	
	reference:bugtraq,10004; reference:cve,2004-0184;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:2486;	Merupakan id dari aturan snort
	rev:5;)	Refisi aturan snort ke 5
20	Alert	Tanda peringatan
	tcp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS BGP spoofed connection reset attempt";	Pesan yang akan diterima apabila terjadi sebuah event

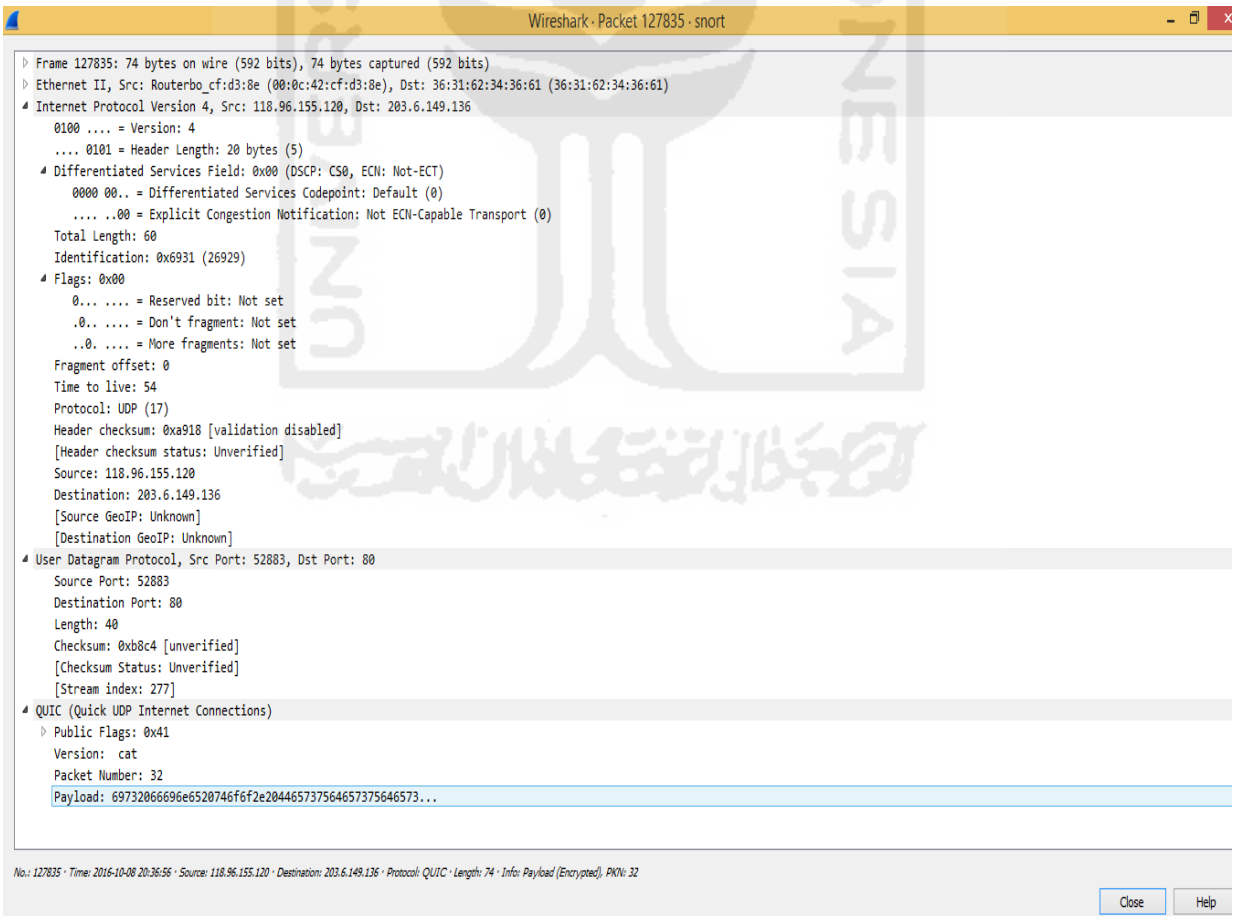
	flow:established;	Koneksi TCP yang terbentuk dalam host sumber ke host tujuan
	flags:RSF*;	Tanda yang dituju
	threshold:type both,track by_dst,count 10,seconds 10;	Tipe yang dipakaidalam waktu detik
	reference:bugtraq,10183; reference:cve,2004-0230; reference:url,www.uniras.gov.uk/vuls/2004/236929/index.htm;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-dos;	Percobaan Denial of Service
	sid:2523;	Merupakan id dari aturan snort
	rev:7;)	Refisi aturan snort ke 7
21	Alert	Tanda peringatan
	Udp	Jenis protokol transport
	$EXTERNAL_NET any	Host asal yang melewati port manapun
	->	Aliran dari host asal ke host tujuan
	(msg:"DOS squid WCCP I_SEE_YOU message overflow attempt";	Pesan yang akan diterima apabila terjadi sebuah event
	content:"|00 00 00 08|";	Isi konten spesifik yang dicari
	depth:4;	Ukuran
	byte_test:4,>,32,16;	Test kecepatan
	reference:cve,CAN-2005-0095; reference:bugtraq,12275;	Merupakan referensi ke system pengidentifikasian serangan eksternal
	classtype:attempted-user;	Mencoba mendapatkan hak user
	sid:3089;	Merupakan id dari aturan snort
	rev:1;)	Refisi aturan snort ke 1

Lampiran gambar analisis bab iv






snort.log.1475856577
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
udp.stream eq 277
No. Time Source Destination Protocol Length Info
127835 2016-10-08 20:36:56 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127837 2016-10-08 20:36:57 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127840 2016-10-08 20:36:57 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127841 2016-10-08 20:36:57 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127854 2016-10-08 20:36:57 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127855 2016-10-08 20:36:57 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127859 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127860 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127863 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127868 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127869 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127876 2016-10-08 20:36:58 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127886 2016-10-08 20:36:59 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127887 2016-10-08 20:36:59 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127897 2016-10-08 20:36:59 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127911 2016-10-08 20:37:00 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127924 2016-10-08 20:37:01 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127925 2016-10-08 20:37:01 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127941 2016-10-08 20:37:02 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127942 2016-10-08 20:37:02 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
127943 2016-10-08 20:37:02 118.96.155.1… 203.6.149.136 QUIC 74 Payload (Encrypted), PKN: 32
Frame 127835: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
Encapsulation type: Ethernet (1)
Arrival Time: Oct 9, 2016 03:36:56.895000000 SE Asia Standard Time
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1475959016.895000000 seconds
[Time delta from previous captured frame: 310.385844000 seconds]
0000 36 31 62 34 36 61 00 0c 42 cf d3 8e 08 00 45 00 61b46a.. B.....E.
0010 00 3c 69 31 00 00 36 11 a9 18 76 60 9b 78 cb 06 .<i1..6. ..v`.x..
0020 95 88 ce 93 00 50 00 28 b8 c4 41 20 63 61 74 20 .....P.(..A cat
0030 69 73 20 66 69 6e 65 20 74 6f 6f 2e 20 44 65 73 is fine too. Des
0040 75 64 65 73 75 73 64 65 73 75 7e udesudes u~
Encapsulation type (frame.encap_type)
Packets: 174063 · Displayed: 85 (0.0%) · Load time: 0:9.678
Profile:


Wireshark · Packet 127835 · snort
Frame 127835: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
Ethernet II, Src: Routerbo_cf:d3:8e (00:0c:42:cf:d3:8e), Dst: 36:31:62:34:36:61 (36:31:62:34:36:61)
Internet Protocol Version 4, Src: 118.96.155.120, Dst: 203.6.149.136
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
0000 00.. = Differentiated Services Codepoint: Default (0)
.... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
Total Length: 60
Identification: 0x6931 (26929)
Flags: 0x00
0... .... = Reserved bit: Not set
.0.. .... = Don't fragment: Not set
..0. .... = More fragments: Not set
Fragment offset: 0
Time to live: 54
Protocol: UDP (17)
Header checksum: 0xa918 [validation disabled]
[Header checksum status: Unverified]
Source: 118.96.155.120
Destination: 203.6.149.136
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
User Datagram Protocol, Src Port: 52883, Dst Port: 80
Source Port: 52883
Destination Port: 80
Length: 40
Checksum: 0xb8c4 [unverified]
[Checksum Status: Unverified]
[Stream index: 277]
QUIC (Quick UDP Internet Connections)
Public Flags: 0x41
Version: cat
Packet Number: 32
Payload: 69732066696e6520746f6f2e20446573756573756573646573...
No.: 127835 · Time: 2016-10-08 20:36:56 · Source: 118.96.155.120 · Destination: 203.6.149.136 · Protocol: QUIC · Length: 74 · Info: Payload (Encrypted), PKN: 32
Close
Help