UNIVERSITAS INDO GLOBAL MANDIRI

PENGENALAN POLA SERANGAN DI JARINGAN KOMPUTER MENGGUNAKAN METODE SIGNATURED BASED STUDI KASUS UNIVERSITAS INDO GLOBAL MANDIRI (UIGM) PALEMBANG

SKRIPSI

M.AGUS MUNANDAR

2015.11.0042P

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

PALEMBANG

2017

UNIVERSITAS INDO GLOBAL MANDIRI

PENGENALAN POLA SERANGAN DI JARINGAN KOMPUTER MENGGUNAKAN METODE SIGNATURED BASED STUDI KASUS UNIVERSITAS INDO GLOBAL MANDIRI (UIGM) PALEMBANG

SKRIPSI

M.AGUS MUNANDAR

2015.11.0042P

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

PALEMBANG

2017

LEMBAR PENGESAHAN

LEMBAR PERSETUJUAN DEWAN PENGUJI

SURAT KETERANGAN REVISI

LEMBAR SIAP SIDANG SKRIPSI

MOTTO DAN KATA PERSEMBAHAN

- *All beginning is difficult and no gain without pain "Semua permulaan adalah sulit dan mencapai sesuatu harus kerja keras".*
- *Never put off till tomorrow what you can do today "Jangan pernah menunggu hari esok apa yang bisa kamu lakukan hari ini".*

Kupersembahkan kepada:

- *Allah SWT yang telah memberi kemudahan, kebaikan serta kesuksesan dalam setiap langkah hidupku dan Rosulullah Muhammad SAW sebagai teladan.*
- *Kedua Orangtuaku, yang telah memberikan dukungan moril mapun materi serta do'a yang tiada henti untuk kesuksesan saya, karena tiada kata seindah do'a dan tiada do'a yang paling khusyuk selain do'a yang terucap dari orang tua. Ucapan terima kasih saja takkan pernah cukup untuk membalas kebaikan orang tua, karena itu terimalah persembahan bakti dan cinta ku untuk kalian kedua orang tua ku.*
- *Saudaraku yang telah memberi dukungan, semangat dan bantuan dalam penyusunan skripsi ini. Terima kasih dan telah menjadi kakak yang terbaik.*
- *Spesial buat mutiara hatiku, engkau yang memounyai kebeningan hati dengan belaian kasih sayang sesejuk embun yang kau berikan padaku sehingga aku bisa bersemangat dan berpacu untuk maju. Aku ingin melihatmu dengan tenang setenang mentari dan sinar pagi, aku mencintaimu dengan lembut selembut sutra dan tetesan air mata, aku menyayangimu seperti sayangnya engkau kepadaku.*
- *Dosen Pembimbingku Dr. Herri Setiawan, M.Kom dan Lastri Widya Astuti, S.Kom., M.Kom yang selalu meluangkan waktu untuk membimbingku dalam menyelesaikan Skripsi.*

PENGENALAN POLA SERANGAN DI JARINGAN KOMPUTER MENGGUNAKAN METODE SIGNATURED BASED STUDI KASUS UNIVERSITAS INDO GLOBAL MANDIRI (UIGM) PALEMBANG

ABSTRAK

Masalah keamanan jaringan semakin menjadi perhatian dikarenakan semakin banyaknya *tools* maupun teknik yang dapat digunakan untuk masuk kedalam sistem secara ilegal dan membuat lumpuh sistem yang ada. Hal tersebut dapat terjadi karena adanya celah dan tidak ada sistem keamanan yang melindunginya sehingga sistem menjadi rentan terhadap serangan. Pengenalan pola serangan di jaringan merupakan salah satu upaya agar serangan tersebut dapat dikenali, sehingga mempermudah administrator jaringan dalam menangani apabila terjadi serangan,*Intrusion Detection System* (IDS) adalah salah satu teknik yang dapat digunakan dalam keamanan jaringan, IDS dapat mendeteksi serangan secara *real time*. IDS dapat membantu administrator dalam mendeteksi serangan yang terjadi.Metode yang diusulkan menggunakan*signatured based*dan simulasi, dimana paket data yang masuk akan dinilai apakah paket data berbahaya atau tidak dan menggunakan beberapa *rule* untuk mencari nilai akurasi terbaik. Beberapa *rule* yang digunakan berdasarkan hasil *training* dan uji menghasilakan 60% hasil *training* dan 50% untuk hasil uji *rule* 1, 50% hasil *training* dan 75% hasil uji *rule* 2, 75% hasil *training* dan hasil uji rule 3, 25% hasil *training* dan hasil uji *rule* 4, 50% hasil *training* dan hasil uji untuk *rule* 5. Hasil pengujian dengan metode *signatured based* ini mampu mengenali pola data serangan dalam mendeteksi protokol TCP dan UDP dan dapat dimonitoring dengan tampilan dalam bentuk *web base.*

Kata Kunci : Pengenalan Pola, IDS, *Snort*, *Snort Rule*, Serangan *Synflood.*

PATTERN RECOGNITION USING COMPUTER NETWORK ATTACK METHODSIGNATURED BASED CASE STUDYINDO GLOBAL MANDIRI UNIVERSITY(UIGM)PALEMBANG

ABSTRACT

The issue of network security is increasingly becoming a concern due to the growing number of tools or techniques you can use to log into the system illegally and paralyzing the existing system. It can occur due to loopholes and no security system that protects him so that the system becomes vulnerable to attack. Pattern recognition of the attack on the network is one of the efforts so that such attacks can be recognized, so that makes it easy to network administrators in handling in the event of an attack, Intrusion Detection System (IDS) is one technique that can be used in network security, IDS can detect attacks in real time. IDS can help administrators in detecting an attack happens. The proposed method using signatured based and simulation, where the incoming data packets will be judged whether or not a malicious data packet and use some rule to figure the value of the best accuracy. Some of the rules that are used based on the results of the training and test result 60% results training and 50% for test results rule 1, 50% and 75% of the training results test results rule 2, 75% of the training and test results rule 3, 25% the results of the training and test results rule 4, 50% yield training and test results for rule 5. The results of testing with this method signatured based able to recognize data patterns of attack in detecting TCP and UDP protocols and can be monitored with the display in the form of a web base.

Keywords : *Pattern Recognition, IDS, Snort, Snort Rule, Synflood attack.*

KATA PENGANTAR

Puji dan syukur penulis persembahkan kehadirat Allah SWT berkat rahmat dan hidayahnya lah penulis akhirnya dapat menyelesaikan penulisan skripsi ini dengan baik dan tepat pada waktu nya, tidak lupa juga shalawat dan salam tercurahkan kepada junjungan kita nabi besar nabi muhammad SAW beserta keluarga, sahabat serta para pengikutnya dan insya Allah kita semua hingga akhir zaman.

Skripsi yang berjudul "Pengenalan Pola Serangan di Jaringan Komputer Menggunaknan Metode *Signatured Based* Studi Kasus Universitas Indo Global Mandiri Palembang" disusun guna memenuhi syarat kelulusan dalam memperoleh gelar sarjana (S1) pada program studi Teknik Informatika Fakultas Ilmu Komputer Universitas Indo Global Mandiri (UIGM) Palembang.

Tidak lupa penulis mengucapkan terima kasih atas bantuan yang diberikan selama penyusunan skripsi ini kepada :

1. Bapak Dr. H. Marzuki Alie, SE., MM, selaku rektor Universitas Indo Global Mandiri (UIGM) Palembang.
2. Ibu Lastri Widya Astuti, M.Kom, selaku Dekan Fakultas Ilmu Komputer UIGM dan Dosen Pembimbing II.
3. Ibu Shinta Puspasari, S.Si., M.Kom, selaku Ketua Program Studi Teknik Informatika Fakultas Ilmu Komputer UIGM.
4. Bapak Dr. Herri Setiawan, M.Kom, selaku Dosen Pembimbing I.
5. Ibu Maya Amelia, M.Kom, selaku Dosen Pembimbing Akademik.
6. Kedua Orang Tua, Kakak tercinta yang telah memberikan dukungan moril maupun materil, kasih sayang serta Do'a sehingga penulis menyelesaikan skripsi ini.
7. Bapak/Ibu Dosen Fakultas Ilmu Komputer UIGM, yang telah mengajarkan dan memberikan ilmu serta berbagai pengalaman.
8. Karyawan/karyawati Universitas Indo Global Mandiri (UIGM) Palembang.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak terdapat kekurangan, karenanya penulis mengharapakan kritik dan juga saran yang membangun agar dapat digunakan demi perbaikan praskripsi ini nantinya. Penulis juga berharap semoga skripsi ini akan memberikan banyak manfaat bagi yang membacanya.

Palembang, 28 Februari 2018
Penulis

M.Agus Munandar
2015.11.0042P

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Pemanfaatan jaringan internet di dunia pendidikan salah satu bentuknya adalah di lingkungan kampus atau perguruan tinggi, hal ini membuktikan bahwa semakin berkembangnya dunia pendidikan khususnya yang terjadi di sumatera selatan. Universitas Indo Global Mandiri (UIGM) sebagai salah satu perguruan tinggi di sumatera selatan telah mempunyai sistem jaringan dan *server* yang digunakan untuk mempermudah dosen dan mahasiswa dalam melakukan aktifitas proses belajar mengajar.

Komputer yang terhubung ke jaringan internet,seperti *server*berpotensi sangat rentan data nya diambil oleh pihak yang tidak bertanggung jawab, kendala tersebut mengakibatkan proses dalam pertukaran data akan menjadi lambat bahkan bisa berakibat kerusakan sistem, pada kasus seperti ini.*Firewall* merupakan salah satu solusi untuk membantu menjaga keamanan jaringan komputer, tetapi jika hanya mengandalkan *firewall* saja belum menjamin keamanannya, sehingga berkembanglah teknologi yang dinamakan*Intrusion Detection System* (IDS).

Intrusion Detection (ID) adalah usaha mengidentifikasi adanya penyusup yang memasuki sistem tanpa otorisasi (misalnya *cracker*) atau seorang user yang sah tetapi menyalahgunakan (*abuse*) *privelege* sumber daya *system* (misal *insider threat*). *Intrusion Detection System* (IDS) atau *system* deteksi penyusupan adalah *systemcomputer* (bisa merupakan kombinasi *software* dan *hardware*) yang berusaha melakukan deteksi penyusupan. IDS akan melakukan pemberitahuan saat terdeteksi sesuatu yang dianggap mencurigakan atau tindakan ilegal. IDS tidak melakukan pencegahan terjadinya serangan(Raharjo, 2015),namun pemanfaatan IDS dapat meminimalisir gangguan/serangan terhadap sistem yang ada dengan cara memberikan peringatan atau *alert* kepada admin jaringan.

Metode yang digunakan dalam pendeteksian intrusi dapat digolongkan menjadi 2 (dua) bagian, yaitu *anomaly detection* dan *misuse detection*,metode *anomaly detection* mendeteksi adanya penyusupan/penyerangan dengan mengamati adanya kejanggalan-kejanggalan pada sistem atau adanya keanehan dan kejanggalan dari kondisi pada saaat sistem normal, apabila menemukan adanya anomali pada paket yang diterima atau dikirimkan, maka akan memberikan peringatan pada pengelola/admin jaringan, dan juga pada metode ini dapat melakukan deteksi intrusi untuk jenis yang baru, untuk metode ini, admin jaringan harus terus memberi tahu IDS bagaimana lalu lintas data yang normal pada sistem jaringan komputer tersebut agar terhindar dari kesalahan penilaian oleh IDS.

Metode *signatured based* telah memiliki daftar *signatured* yang dapat digunakan untuk menilai apakah paket yang dikirimkan berbahaya atau tidak. Sebuah paket data akan dibandingkan dengan daftar yang sudah ada. Metode ini akan melindungi sistem dari jenis-jenis serangan yang sudah diketahui sebelumnya. Oleh karena itu, untuk tetap menjaga keamanan sistem jaringan komputer, data *signatured* yang ada harus tetap ter-*update*(Alamsyah, 2011).

Berdasarkan latar belakang tersebut, maka dalam penelitian ini, penulis mengambil judul ***"Pengenalan Pola Serangan di Jaringan Komputer dengan Metode Signatured Based Studi Kasus Universitas Indo Global Mandiri (UIGM) Palembang",*** yang diharapkan dapat menjaga dan memonitoring keamanan dalam jaringan komputer.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan sebelumnya maka rumusan masalah pada penelitian ini yaitu mengenai pola serangan di jaringan, rumusan masalah ini bisa diperinci sebagai berikut :

1. Bagaimana bentuk peringatan yang diberikan kepada administrator ?
2. Bagaimana mengenali pola paket data serangan yang ada di jaringan ?

1.3 Batasan Masalah

Batasan masalah yang dibahas dalam penelitian ini, yaitu :

1. Sistem Operasi yang digunakan yaitu sistem operasi *linux*.
2. Serangan menggunakan DDOS.
3. Menggunakan *snort* IDS sebagai pendeteksi penyusupan.
4. Tidak membahas cara pencegahan.
5. Menggunakan 5 (lima) *rule* IDS sebagai pendeteksi serangan.

1.4 Tujuan dan Manfaat

Tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut :

1. Melakukan penggunaan *tools* untuk mengenali pola serangan yang ada di jaringan dengan aplikasi *snort*.
2. Menerapkan metode *signatured based* untuk mengenali pola data normal dan pola data serangan.
3. Melakukan pembuatan monitoring dan penanganan serangan di jaringan.

Manfaat yang ingin dicapai dalam penelitian ini adalah :

1. Mempermudah admin jaringan komputer dalam melakukan monitoring.
2. Dapat mengetahui pola penyerangan di jaringan komputer.
3. Dapat mencegah serangan yang dilakukan *attacker* secara otomatis.

1.5 Sistematika Penulisan

Sistematika penulisan penelitian ini disusun untuk memberikan gambaran umum mengenai penelitian yang dijalankan (*Panduan Skripsi*, 2017). Sistematika penulisan penelitian ini adalah sebagai berikut :

BAB 1 PENDAHULUAN

Bab 1 ini akan membahas mengenai latar belakang permasalahan, perumusan masalah, batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab 2 ini berisi uraian literatur atau pustaka yang terkait dengan permasalahan penelitian, termasuk pembahasan atas teori, temuan dan bahan

penelitian sebelumnya yang diperoleh dari berbagai referensi sebagai dasar penelitian. Pada bab ini dijelaskan secara komprehensif tentang semua materi terkait penelitian.

BAB 3 METODE PENELITIAN

Bab 3 ini menjelaskan tentang analisis dimana mencakup analisis kebutuhan yang akan digunakan baik perangkat keras *(hardware)* maupun perangkat lunak *(software)*, perancangan topologi jaringan, tahapan-tahapan penelitian.

BAB 4 HASIL DAN PEMBAHASAN

Bab 4 ini menjelaskan tentang hasil analisis mencakup simulasi serangan dan membahas mekanisme dari penelitian tersebut.

BAB 5 PENUTUP

Bab 5 ini berisi kesimpulan apa saja berdasarkan hasil analisa pada bab-bab sebelumnya beserta saran yang bisa digunakan untuk perbaikan dan pengembangan sistem yang akan dibangun.

BAB 2
LANDASAN TEORI

2.1 Jaringan Komputer

Jaringan komputer *(computer networks)* adalah suatu himpunan interkoneksi sejumlah komputer *autonomous*, dalam bahasa yang populer dapat dijelaskan bahwa jaringan komputer adalah kumpulan beberapa komputer (dan perangkat lain seperti *router, switch,* dan sebagainya) yang saling terhubung satu sama lain melalui media perantara. Media perantara ini bisa berupa media kabel ataupun media tanpa kabel (nirkabel) (Sofana, 2015).

Menurut Pratama (2015), jaringan komputer dapat dikelompokkan berdasarkan jangkauan geografis, secara umum jaringan komputer terbagi menjadi 4 (empat) yaitu :

1. *Local Area Network* (LAN)

Local Area Network (LAN) merupakan jaringan komputer terkecil untuk pemakaian pribadi, LAN memiliki skala jangkauan mencakup 1 Km hingga 10 Km, dalam bentuk koneksi *wired* (kabel), *wireless* (nirkabel), maupun kombinasi keduanya. Umumnya LAN lebih banyak diimplementasikan di dalam sebuah ruangan maupun sebuah gedung.

2. *Metropolitan Area Network* (MAN)

Metropolitan Area Network (MAN) merupakan jaringan komputer yang memiliki cakupan area dan luas yang lebih besar dibandingkan LAN. MAN memiliki jarak jangkauan antara 10 Km hingga 50 Km, wilayah jangkauan MAN dapat mencakup sebuah wilayah kota, yang didalamnya terdapat banyak gedung dan pemukiman, ini berarti di dalam sebuah MAN telah terintegrasi banyak LAN yang berasal dari berbagai gedung dan pemukiman yang ada.

3. *Wide Area Network* (WAN)

Wide Area Network (WAN) merupakan jaringan komputer yang lebih luas dari MAN, dengan cakupan area seluas sebuah negara atau benua. WAN terdiri

atas dua atau lebih MAN di dalamnya. Setiap MAN tersebut terdiri atas dua atau lebih LAN didalamnya, sehingga dapat dikatakan bahwa WAN merupakan gabungan dari sejumlah jaringan komputer yang berada dalam satu kawasan seluas sebuah negara ataupun benua.

4. Internet

Internet atau *Interconnection Networking* merupakan jaringan komputer yang terluas, dengan cakupan seluruh planet bumi ini. Internet menghubungkan semua WAN, MAN, dan LAN di dalamnya, sehingga dapat dikatakan bahwa internet terdiri atas semua komputer dan perangkat lainnya kedalam satu jaringan komputer terbesar di dunia, yang menghubungkan setiap gedung, setiap tempat, setiap pengguna komputer, dari berbagai daerah, kota, negara, pulau, benua didalam kesatuan alam bumi ini.

2.2 Topologi Jaringan Komputer

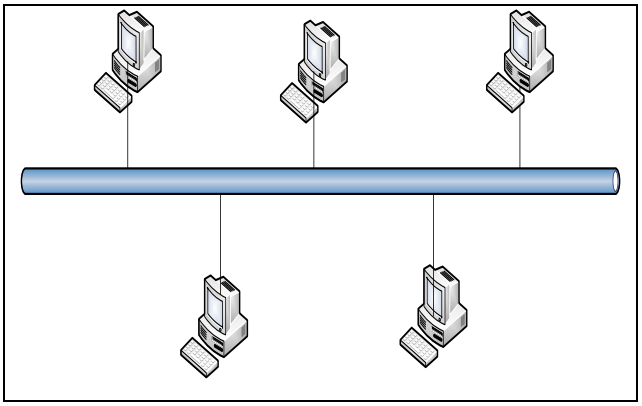
Topologi dapat diartikan sebagai *layout* atau arsitektur atau diagram jaringan komputer. Topologi merupakan suatu aturan/*rules* bagaimana menghubungkan komputer *(node)* secara fisik. Topologi berkaitan dengan cara komponen-komponen jaringan (seperti: *server, workstation, router, switch*) saling berkomunikasi melalui media transmisi data (Sofana, 2015), sedangkan menurut Ginta dkk (2013), topologi jaringan komputer adalah suatu cara menghubungkan komputer yang satu dengan komputer lainnya sehingga membentuk pola hubungan antar terminal dalam suatu sistem jaringan yang dapat mempengaruhi tingkat efektifitas kinerja jaringan.

Ada beberapa jenis topologi jaringan yang dapat di implementasikan dalam jaringan komputer yaitu topologi Bus, topologi Ring, topologi Star, topologi Mesh

1. Topologi Bus

Topologi bus merupakan topologi yang menghubungkan semua terminal kesatu jalur komunikasi yang kedua ujungnya ditutup dengan terminator. Terminator adalah perangkat yang menyediakan resistansi listrik untuk menyerap

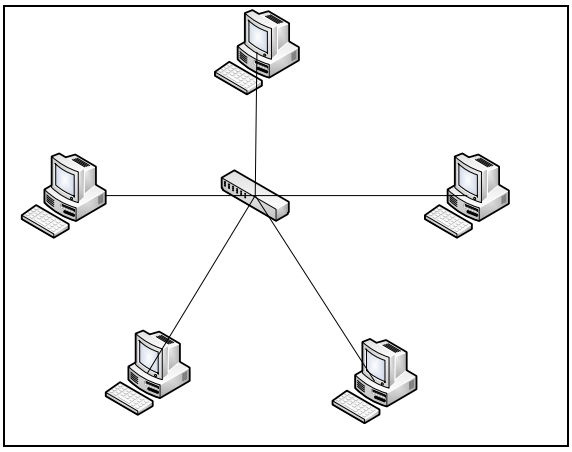
sinyal pada akhir transmisi sambungan agar sinyal tidak terlontar kembali dan diterima oleh stasiun jaringan.



Gambar 2.1 Topologi BUS

2. Topologi Star

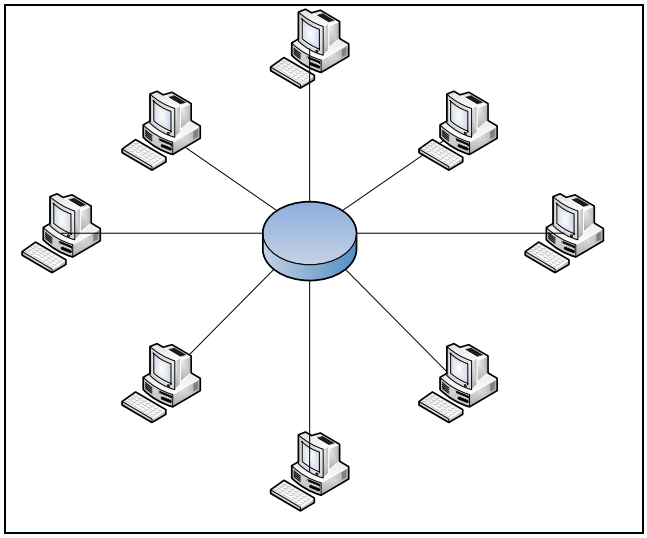
Topologi star didesain dimana setiap *node* (*file server, workstation,* dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *Hub* atau *concentrator*.



Gambar 2.2 Topologi Star

3. Topologi Ring

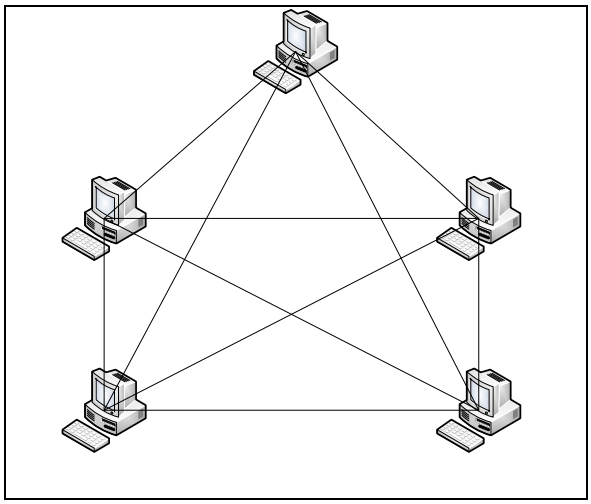
Topologi ring yaitu semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin. Tiap workstation ataupun *server* akan menerima dan melewatkan informasi dari satu komputer ke komputer lain, bila alamat-alamat yang dimaksud sesuai maka informasi diterima dan bila tidak informasi akan dilewatkan.



Gambar 2.3 Topologi Ring

4. Topologi mesh

Topologi mesh memiliki hubungan yang berlebihan antara peralatan-peralatan yang ada, susunan dalam suatu jaringan saling berhubungan dengan perlatan yang lainnya.



Gambar 2.4 Topologi Mesh

Tabel 2.1 Topologi Jaringan Komputer

No	Topologi	Kelebihan	Kekurangan
1	**Bus**	a. Topologi yang sederhana b. Kabel yang digunakan	a. *Traffic* (lalu lintas) yang padat akan

		sedikit untuk menghubungkan komputer-komputer atau peralatan yang lain. c. Biaya lebih murah d. Cukup mudah apabila ingin memperluas jaringan.	memperlambat jalur bus. b. Seluruh jaringan mati jika terjadi kerusakan pada kabel utama c. Membutuhkan *terminator* pada kedua sisi kabel utamanya d. Sangat sulit mengidentifikasi permasalahan jika jaringan mati e. Paling lambat
2	**Star**	a. Mudah dipasang dan pengkabelan b. Tidak mengakibatkan gangguan bila terjadi perbaikan c. Mudah untuk mendeteksi kesalahan dan memindahkan perangkat lain	a. Memiliki satu titik kesalahan, terletak pada *hub*, jika *hub* pusat mengalami kegagalan, maka seluruh jaringan akan gagal untuk beroperasi b. Membutuhkan lebih banyak kabel karena semua kabel jaringan harus ditarik se satu *central point*. c. Jumlah terminal terbatas, tergantung dari port yang ada pada *hub*.
3	**Ring**	a. Data mengalir dalam satu arah sehingga terjadinya *collision* dapat dihindarkan	a. Apabila ada satu komputer dalam ring yang gagal berfungsi,

		b. Aliran data mengalir lebih cepat karena dapat melayani data dari kiri atau kanan dari *server* c. Dapat melayani aliran lalulintas data yang padat, karena data dapat bergerak kekiri atau kekanan.	maka akan mempengaruhi keseluruhan jaringan b. Menambah atau mengurangi komputer akan mengacaukan jaringan c. Sulit untuk melakukan konfigurasi ulang
4	**Mesh**	a. Keuntungan utama adalah *fault tolerance* b. Terjaminnya kapasitas *channel* komunikasi, karena memiliki hubungan yang berlebih. c. Relatif lebih mudah untuk dilakukan *troubleshoot*	a. Sulitnya pada saat melakukan instalasi dan melakukan konfigurasi ulang saat jumlah komputer dan peralatan-peralatan yang terhubung semakin meningkat jumlahnya b. Biaya yang besar untuk memelihara hubungan yang berlebih

2.3 IP Address

Menurut Pratama (2015), IP Address merupakan bentuk alamat secara numerik (desimal dan bineri) yang diberlakukan kepada semua komputer dan perangkat terhubung lainnya didalam jaringan komputer, yang memanfaatkan Internet Protocol (IP) sebagai protokol untuk proses komunikasi dan transmisi paket data. IP Address pada umumnya dikelompokkan ke dalam tiga kelas, yaitu kelas A, kelas B, dan kelas C, perbedaaan dari masing-masing kelas yaitu :

1. Kelas A

   Kelas A pada IP Address dimulai dari pengalamatan 10.0.0.0 hingga 10.255.255.255. range IP address kelas A ini akan mampu memuat

total jumlah alamat bagi komputer dan perangkat terhubung lainnya sebanyak 16.777.216 buah.

2. Kelas B

   Kelas B pada IP Address dimulai dari pengalamatan 172.16.0.0 hingga 172.31.255.255. range IP Address kelas B ini akan mampi memuat total jumlah alamat bagi komputer dan perangkat terhubung lainnya sebanyak 1.048.576 buah.

3. Kelas C

   Kelas C pada IP Address dimulai dari pengalamatan 192.168.0.0 hingga 192.168.255.255. range IP address kelas C ini akan mampu memuat total jumlah alamat bagi komputer dan perangkat terhubung lainnya sebanyak 65.536 buah.

2.4 OSI Layer

Menurut Sofana (2015) OSI *Reference Modelfor Open Networking* atau model referensi jaringan terbuka OSI adalah sebuah model arsitektural jaringan yang dikembangkan oleh badan *international organization for standardization* (ISO) di eropa pada tahun 1977. OSI sendiri merupakan singkatan dari *open system interconnection*. Model ini disebut juga dengan "model tujuh lapis OSI" (OSI *seven layer* model).

Model OSI dibuat untuk mengatasi berbagai kendala *internetworking* akibat perbedaan arsitektur dan protokol jaringan. Berikut ini adalah penjelasan masing-masing layer OSI :

1. *Layer 7 Application*

Application layer berfungsi sebagai antarmuka (penghubung) aplikasi dengan fungsionalitas jaringan, mengatur bagaimana aplikasi dapat mengakses jaringan, dan kemudian membuat pesan-pesan kesalahan. Pada layer inilah sesungguhnya user "berinteraksi dengan jaringan". Contoh protokol yang berada pada lapisan ini adalah, FTP, telnet, SMTP, HTTP, POP3, dan NFS.

2. *Layer 6 Presentation*

Presentation layer berfungsi untuk mentranslasikan data yang hendak di transmisikan oleh aplikasi kedalam format yang dapat ditransmisikan melalui jaringan. Protokol yang berada pada level ini adalah sejenis *redirector software*, seperti *network shell* (semacam *virtual network computing* (VNC) atau *remote desktop protocol* (RDP). Kompresi data dan enkripsi juga ditangani oleh layer ini,

3. *Layer 5 Session*

Session layer berfungsi untuk mendefinisikan bagaimana koneksi dimulai, dipelihara, dan diakhiri. Selain itu, di level ini juga dilakukan resolusi nama. *Layer Session*, sering disalah artikan sebagai prosedur *logon* pada *network* dan berkaitan dengan keamanan. Beberapa protokol dalam layer ini adalah :

a. Netbios, protokol yang dikembangkan IBM, menyediakan layanan ke layer presentation dan layer application.
b. NETBEUI, (*Netbios Extended User Interface*), protokol pengembangan dari NETBIOS, digunakan pada *Microsoft networking*.
c. ADSP (*AppleTalk Data Stream Protocol*).
d. PAP (*Printer Access Protocol*), protokol untuk printer *Postscript* pada jaringan *AppleTalk*.

4. *Layer 4 Transport*

Transport Layer berfungsi untuk memecah data menjadi paket-paket data serta memberikan nomor urut setiap paket sehingga dapat disusun kembali setelah diterima. Paket yang diterima dengan sukses akan diberi tanda (*acknowledgement*). Sedangkan paket yang rusak atau hilang di tengah jalan akan dikirim ulang. Contoh protokol yang digunakan pada layer ini adalah UDP, TCP dan SPX.

5. *Layer 3 Network*

Network Layer berfungsi untuk mendefinisikan alamat-alamat IP, membuat *header* untuk paket-paket, dan melakukan *routing* melalui *internetworking* dengan menggunakan *router* dan *switch layer* 3. Pada *layer* ini juga dilakukan proses deteksi *error* dan transmisi ulang paket-paket yang *error*. Contoh protokol yang digunakan adalah IP dan IPX.

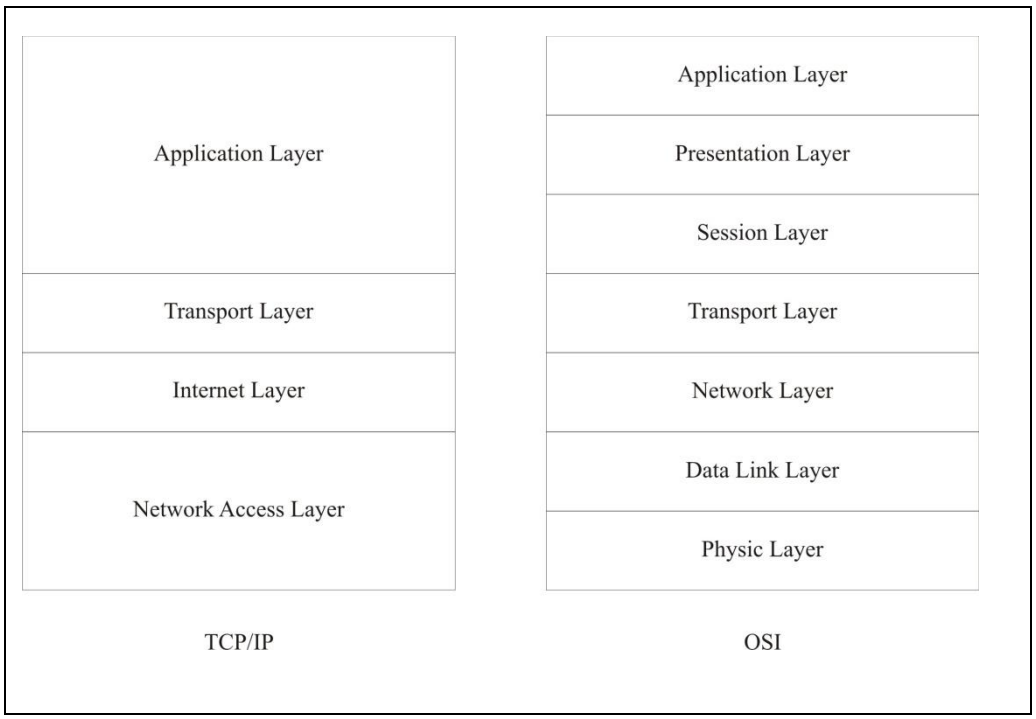
6. Layer 2 Data Link

Data Link Layer berfungsi untuk menentukan bagaimana bit-bit data dikelomopokkan menjadi format yang disebut *frame*. Pada level ini terjadi *error correction, flow control*, pengalamatan perangkat keras (*MAC Address*), dan menentukan bagaimana perangkat-perangkat jaringan seperti *bridge* dan *switch layer* 2 beroperasi.

Menurut spesifikasi IEEE 802, *layer* ini dikelompokkan menjadi dua yaitu *logical link control* (LLC) dan *media access control* (MAC). Contoh protokol pada layer ini adalah ethernet (802.2 & 802.3), tokenbus (802.4), tokenring (802.5), demang priority (802.12).

7. Layer 1 Physical

Physical layer berfungsi untuk mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti halnya Ethernet atau Token Ring), topologi jaringan, dan pengabelan. Selain itu, level ini juga mendefinisikan bagaimana *network interface card* (NIC) berinteraksi dengan media *wire* atau *wireless*.



Gambar 2.5 OSI Layer

Menurut Knowledge (2010), sebuah port dalam jaringan komputer dapat dikatakan sebagai pintu keluar masuk paket data ke atau dari sebuah host. Port dapat mengidentifikasikan aplikasi dan layanan yang menggunakan koneksi didalam jaringan TCP/IP, port UDP dan TCP dibagi menjadi tiga, yaitu :

1. *Well-know Port* : pada awalnya berkisar antara 0 hingga 255 kemudian diperlebar menjadi 0 hingga 1023, port number yang termasuk ke dalam *well-know port*, selalu mempresentasikan layanan jaringan yang sama, dan ditetapkan oleh *internet assigned number authority* (IANA)
2. *Registered Port* : port-port yang digunakan oleh vendor-vendor komputer atau jaringan yang berbeda untuk mendukung aplikasi dan sistem operasi yang mereka buat. *Registered port* juga diketahui dan didaftarkan oleh IANA tetapi tidak dialokasikan secara permanen, sehingga vendor lainnya dapat menggunakan port number yang sama, *range registered port* dari 1024 hingga 49151
3. *Dinamically Assigned Port* : merupakan port-port yang ditetapkan oleh sistem operasi atau aplikasi yang digunakan untuk melayani *request* dari pengguna seusai dengan kebutuhan, *range dinamically assigned port* dari 1024 hingga 65536

Tabel 2.2 Port yang sering digunakan pada jaringan komputer

Port	Jenis Port	Fungsi
21	TCP, UDP	*File Transfer Protocol* (FTP)
22	TCP, UDP	*Remote Server* dan *Transfer Data* (SSH)
23	TCP, UDP	*Remote Server* (Telnet)
25	TCP, UDP	*Mail Server* (SMTP)
53	TCP, UDP	*Domain Name Server* (DNS)
80, 81	TCP, UDP	*Web Server* (HTTP)
110	TCP, UDP	*Mail Server* (POP3)

Sumber : Knowledge (2010)

2.5 Keamanan Komputer

Jaringan komputer memainkan peran nyata di dalam kehidupan manusia saat ini, hampir semua layanan saat ini dijalankan secara *online* dan *mobile*, yang memerlukan koneksi ke dalam jaringan komputer. Setiap layanan memerlukan adanya keamanan sistem di dalamnya, bertujuan untuk menciptakan layanan yang memberikan rasa nyaman dan percaya (*trust*) bagi para pengguna layanan tersebut.

Menurut Pratama (2015) inti dari keamanan komputer adalah melindungi komputer dan jaringannya dengan tujuan mengamankan informasi yang ada di dalam nya, keamanan komputer meliputi beberapa aspek, yaitu :

1. *Confidetiality*

   Confideality atau disebut juga dengan kerahasiaan merupakan tujuan utama dari keamanan komputer dan keamanan jaringan komputer. Kerahasiaan ini harus dijaga baik oleh pengguna maupun oleh sistem di dalam jaringan komputer.

2. *Integrity*

   Integrity merupakan upaya untuk menjaga agar data dan informasi tidak diubah oleh pihak yang tidak berhak, sehingga keabsahan data dan informasi tetap terjaga.

3. *Availability*

   Availability atau ketersediaan merupakan upaya untuk menyediakan akses dan otoritas kepada pihak-pihak yang berhak terhadap data dan informasi tersebut, dengan adanya *availability* ini maka secara jelas ditampilkan siapa saja yang memiliki hak terhadap akses ke dalam sistem, sesuai dengan hak akses yang diberikan.

Tujuan utama keamanan jaringan dapat dicapai dengan metode keamanan yang dapat melindungi sistem yang ada di jaringan baik dari dalam maupun dari luar jaringan, namun bukan hanya melindungi tetapi juga harus dapat bertindak apabila terjadinya serangan yang terdapat di dalam jaringan. Salah satu metodenya adalah IDS, namun selain metode tersebut dibutuhkan juga suatu pemahaman

dalam menentukan kebijakan keamanan (*security policy*) dalam keamanan jaringan.

1. Jenis-jenis Serangan

Menurut Manuaba (2013), jenis-jenis serangan pada jaringan komputer yang biasa dilakukan oleh *attacker* adalah sebagai berikut :

1. Reveal SSID

Merupakan serangan yang dilakukan dengan menyingkap SSID dari *access point* yang sengaja disembunyikan oleh administrator jaringan komputer.

2. MAC Address Spoofing

Merupakan usaha yang dilakukan oleh seorang *hacker* untuk menembus keamanan *MAC address filtering* dengan cara melakukan *spoofing MAC address* pada jaringan komputer, dengan menggunakan *MAC address user* yang sah untuk mendapatkan layanan jaringan komputer.

3. Authentication Attack

Merupakan serangan terhadap *authentication user* yang sah, sehingga menyebabkan kelumpuhan atau *disconnectnya user* yang sah. *Attacker* memanfaatkan serangan ini agar mendapatkan *resource* yang lebih dalam menggunakan layanan jaringan.

4. Eavesdropping

Merupakan serangan yang dilakukan dengan cara mendengarkan semua paket-paket yang ditransmisikan oleh *user* yang berada dalam jaringan komputer yang tidak ter*enkripsi*.

5. *Session Hijacking*

Merupakan suatu serangan yang menyerang suatu sesi seorang pengguna untuk dimanfaatkan sebagai ajang untuk mendapatkan suatu hak akses ke layanan yang sedang diakses oleh *user* yang sah.

6. Man In The Middle Attack

Merupakan serangan yang dilakukan dengan melakukan *spoofing* terhadap *user* sah sehingga transmisi yang dilakukan target adalah

menuju *attacker*, sehingga *attacker* mendapatkan semua informasi yang di transmisikan oleh target.

7. *Denial of Service*

   Merupakan serangan yang menyerang ketersediaan sumber daya sehingga menyebabkan *user* sah mengalami *disconnect* dari jaringan komputer.

8. *Rogue Access Point*

   Merupakan serangan yang menggunakan suatu perangkat access point yang dibuat sama dengan access point yang berada pada suatu institusi.

2.6 IDS

Menurut Wu (2009), *Intrusion Detection System* (IDS) adalah suatu tindakan untuk mendeteksi adanya trafik paket yang tidak diinginkan dalam sebuah jaringan atau *device*. IDS dapat diimplementasikan melalui *software* atau aplikasi yang terinstall dalam sebuah *device*, dan aplikasi tersebut dapat memantau paket jaringan untuk mendeteksi adanya paket-paket ilegal seperti paket yang merusak kebijakan *rules* keamanan, dan paket yang ditujukan untuk mengambil hak akses pengguna.

2.6.1 Jenis Intrusion Detection System (IDS)

Pendeteksian dalam implementasi IDS ada beberapa jenis yaitu *signature based detection*, *anomaly based detection*, dan *stateful protocol inspection*, berikut penjelasan mengenai macam-macam jenis pendeteksian :

1. *Signature based detection*

   Sebuah IDS dapat menggunakan pendeteksian *signature based detection* dari sebuah paket, tergantung data paket yang diketahui untuk menganalisa potensi terjadinya paket ilegal. Tipe pendeteksian ini sangat cepat dan mudah dikonfigurasi. Bagaimanapun juga seorang penyerang dapat dengan mudah memodifikasi sebuah serangan untuk menyiasati agar tidak terkenali oleh *signature based* IDS, meskipun

kemampuan tipe ini terbatas dalam mendeteksi banyaknya serangan, tipe ini mempunyai kelebihan dalam hal keakuratan.

2. *Anomaly based detection*

   IDS yang dapat memantau paket jaringan dan mendeteksi data yang tidak valid, atau umumnya tidak normal menggunakan jenis deteksi *anomaly based*, metode ini berguna untuk mendeteksi paket-paket yang tidak diinginkan.

3. *Stateful protocol inspection*

   Stateful protocol inspection menyerupai pendeteksi berbasis *anomaly*, tetapi jenis ini dapat menganalisa paket lapisan 3 OSI yaitu lapisan *network* dan lapisan 4 yaitu lapisan *protocol.*

2.6.2 Keuntungan dan Kerugian IDS

Menurut (Wagner, 2007) keuntungan dan kekurangan dari IDS adalah :

1. Keuntungan IDS
   a. IDS dapat disesuaikan dengan mudah dalam menyediakan perlindungan untuk keseluruhan jaringan
   b. IDS dapat dikelola secara terpusat dalam menangani serangan yang tersebar dan bersama-sama.
   c. IDS menyediakan pertahanan pada bagian dalam.
   d. IDS memonitor internet untuk mendeteksi serangan.
   e. IDS melacak aktivitas pengguna dari saat masuk hingga saat keluar.

2. Kekurangan IDS
   a. IDS lebih bereaksi pada serangan daripada pencegahan.
   b. IDS menghasilkan data yang besar untuk dianalisis.
   c. IDS hanya melindungi dari karakteristik yang dikenal.
   d. IDS tidak turut bagian dalam kebijakan keamanan yang efektif, karena IDS harus di *setting* terlebih dahulu.
   e. IDS tidak mengidentifikasi asal serangan.

2.6.3 Peran IDS

IDS memiliki peran penting untuk mendapatkan arsitektur *defence in depth* (pertahanan yang mendalam) dengan cara melindungi akses jaringan internal, sebagai tambahan dari parameter *defence*(Putri, 2011). Hal-hal yang dilakukan IDS pada jaringan internal adalah sebagai berikut :

1. Memonitor akses *database* : ketika mempertimbangkan pemilihan kandidat untuk penyimpanan data, suatu perusahaan akan memilih database sebagai solusi untuk menyimpan data-data yang berharga.
2. Melindungi email *server* : IDS juga berfungsi untuk mendeteksi virus email seperti QAZ, worm, navidad worm, dan versi terbaru explorezip.
3. Memonitor *policy security* : apabila ada pelanggaran terhadap *policy security* maka IDS akan memberitahu bahwa telah terjadi sesuatu yang tidak sesuai.

2.7 Snort

Snort merupakan sebuah perangkat lunak yang berfungsi untuk memeriksa data-data yang masuk dan melaporkan ke administrator apabila terdapat aktifitas-aktifitas mencurigakan. Snort pertama kali dibuat dan dikembangkan oleh Martin Roesch, lalu menjadi *open source project* (*www.snort.org*), snort dapat dioperasikan ke dalam tiga mode, yaitu

1. *Sniffer mode*, pada mode operasi ini snort bertindak sebagai *sniffer*, yaitu snort dapat mennangkap atau melihat semua paket yang lewat dalam jaringan dimana snort diletakkan. Snort menampilkan hasil dari *sniffing* ini secara *real time*.
2. *Packet logger mode*, pada mode ini selain dapat melihat semua paket yang lewat, snort juga dapat mencatat atau *logging* menyimpannya pada *storage*.
3. *Network intrusion detection mode*, ciri khas dari mode ini adalah dengan menjalankan snort beserta file konfigurasi yang telah ditentukan (secara *default file snort.conf*). file *snort.conf* ini sudah

banyak mencakup konfigurasi-konfigurasi yang dibutuhkan dalam mode ini.

Snort mempunyai komponen utama yang bekerja saling berhubungan satu dengan yang lain, berikut komponen snort :

1. *Snort rules*

   Rule snort merupakan aturan-aturan yang bertujuan mengklasifikasikan aktifitas-aktifitas jaringan yang normal dan tidak. *Rules snort* ini merupakan *database* yang berisi pola-pola serangan berupa *signature* jenis-jenis serangan. *Rules snort* ini harus diupdate secara rutin agar ketika ada suatu teknik serangan yang baru, serangan tersebut dapat di deteksi (Rafiudin, 2010)

2. *Snort engine*

   Snort engine merupakan program yang berjalan sebagai *daemon* proses yang selalu bekerja untuk membaca paket data dan akan membandingkannya dengan rules snort

3. *Alert*

   Alert merupakan catatan serangan yang dideteksi oleh *snort engine*. Jika *snort engine* mendeteksi adanya serangan, maka *snort engine* akan mengirimkan *alerts* berupa *log file*, biasanya *alerts* disimpan dalam *database*.

Menurut (Mentang, Sinsuw and Najoan, 2015) snort mempunyai beberapa fitur, yaitu :

1. Karena snort bersifat *open source*, maka penggunaannya gratis, oleh karena itu, snort merupakan pilihan yang sangat baik sebagai NIDS ringan yang *cost-effective* dalam suatu organisasi yang kecil jika organisasi tersebut tidak mampu menggunakan NIDS *commercial* yang harganya lumayan besar.
2. Karena snort bersifap *open source*, maka penggunaannya bebas sehingga dapat diterapkan dalam lingkungan apa saja, kode sumbernya pun bisa didapatkan sehingga snort dapat secara bebas dimodifikasi sendiri sesuai keperluan.

3. Snort memliki bahasa pembuatan *rules* yang relatif mudah dipelajari dan *fleksibel*, berarti pengguna dapat dengan mudah membuat berbagai *rules* baru untuk mendeteksi tipe-tipe serangan yang baru.
4. Snort sudah memiliki sebuah *database* untuk berbagai macam *rules*, dan juga *database* ini secara aktif terus dikembangkan sehingga tipe-tipe serangan yang baru dapat dideteksi dan dicatat.
5. Snort merupakan *software* yang ringkas dan padat, sehingga tidak memakan banyak *resources* tetapi cukup canggih dan *fleksibel* untuk digunakan sebagai salah satu bagian dari NIDS.
6. Snort dapat melakukan *logging* langsung ke sistem *database* (MySQL).
7. Snort sebagai NIDS dapat menyembunyikan dirinya dalam jaringan komputer sehingga keberadaannya tidak bisa terdeteksi oleh komputer mana pun, disebut *stealth mode*.

2.8 Penelitian Terdahulu

Penelitian ini mengcau pada penelitian terdahulu (penelitian sebelumnya) yang telah dilakukan beberapa peniliti diantaranya :

Penelitian sebelumnya yang dilakukan oleh Sahid Aris Budiman, Catur Iswahyudi, dan Muhammad Sholeh yang mengambil judul "Implementasi *Intrusion Detection System* (IDS) menggunakan Jejaring Sosial Sebagai Media Notifikasi", hasil yang dicapai dalam penelitian ini adalah apabila menunjukkan adanya serangan yang datang dari luar menuju *host* atau *server* yang terdapat IDS, maka secara otomatis akan mendeteksi dan memberitahukan kepada administrator berupa notifikasi yang dikirimkan melalui jejaring sosial Facebook, Twitter, dan Whatsaap, sehingga administrator jaringan dapat menindak lanjuti terhadap jenis serangan yang dilakukan oleh *intruder*(Budiman *et al*., 2014).

Penelitian sejenis juga pernah dilakukan oleh Harjono dan Agung Purwo Wicaksono dengan judul " Sistem Deteksi Intrusi dengan Snort (*Intrusion Detetction System with Snort*), yang mana penelitian ini menghasilkan sistem yang dapat mendeteksi intrusi pada sistem yang dipantau, sistem dapat

menghasilkan log yang tersimpan didalam database, selain itu juga, Alert yang dihasilkan dapat ditampilkan dan dianalisis dalam tampilan web (Wicaksono *et al.*, 2014).

Adapula Penelitian tentang IDS yang dilakukan oleh Baskoro A. Pratomo dan Royanna M. Ijtihadie yang mengambil judul "Sistem Deteksi Intrusi Menggunakan N-Gram dan Cosine Similarity" dengan hasil mencari kemiripan dari serangkaian paket dengan signature yang ada, paket tidak dicocokkan dengan pola serangan, tetapi dengan pola pengaksesan sebuah halaman web oleh pengguna yang sesungguhnya, sehingga yang memiliki kemiripan tinggi akan dianggap sebagai paket yang sah, sedangkan yang rendah akan dianggap sebagai serangan(Pratomo *et al*., 2016).

Adapula penelitian tentang IDS yang dilakukan oleh Muh Masruri Mustofa, dan Eko Aribowo yang mengambil judul "Penerapan Sistem Keamanan *Honeypot* dan IDS pada Jaringan Nirkabel (*Hotspot*)" dengan hasil penelitian adalah sistem keamanan yang berlapis dengan cara menipu atau memberikan data palsu apabila ada penyerang yang akan masuk kesuatu sistem atau server utama, dan juga honeypot akan merekam aktifitas dari penyerang dalam bentuk *log*, sedangkan snort memberikan rekaman traffik yang jangga dalam bentuk *file log* atau *alert* (Mustofa *et al*., 2013).

Adapula penelitian tentang IDS yang dilakukan oleh Fitriyanti A.Masse, Andi Nurul Hidayat, dan Badrianto yang mengambil judul "Penerapan Network Intrusion Detection System Menggunakan Snort Berbasis *Database* MySql pada Hotspot kita" dengan hasil penelitian adalah bahwa setiap tindakan yang dilakukan oleh penyerang terhadap jaringan dapat diketahui oleh mesin sensor, sehingga dapat dilakukan pencegahan sebelum terjadi kerusakan pada jaringan yang lebih luas (A.Masse *et al*., 2015).

Berdasarkan penelitian terdahulu, maka pada penelitian ini dibuat pengenalan pola serangan di jaringan komputer yang mengambil tempat di Universitas Indo Global Mandiri Palembang dan menggunakan metode Signatured based yang diharapkan pada penelitian ini adalah agar mampu

mendeteksi serangan di jaringan komputer dan dapat memblok serangan secara otomaatis

Perbedaan penelitian ini dengan penelitian terdahulu dapat dilihat pada tabel 2.3.

Tabel 2.3 Perbandingan Penelitian

No	Nama	Judul	Tahun	Metode	Ringkasan Penelitian
1.	Budiman dkk	Implementasi Intrusion Detection System (IDS) Menggunakan Jejaring Sosial Sebagai Media Notifikasi	2014	Studi Pustaka	Penelitian ini menghasilkan notifikasi ke jejaring sosial facebook, twitter, whatsapp jika terdapat serangan dalam sistem
2.	Harjono dan Wicaksono	Sistem Deteksi Intrusi dengan Snort (*Intrusion Detection System with Snort*)	2014	Studi Pustaka	Penelitian ini menghasilkan pendeteksian serangan yang dapat ditampilkan dalam bentuk *web*

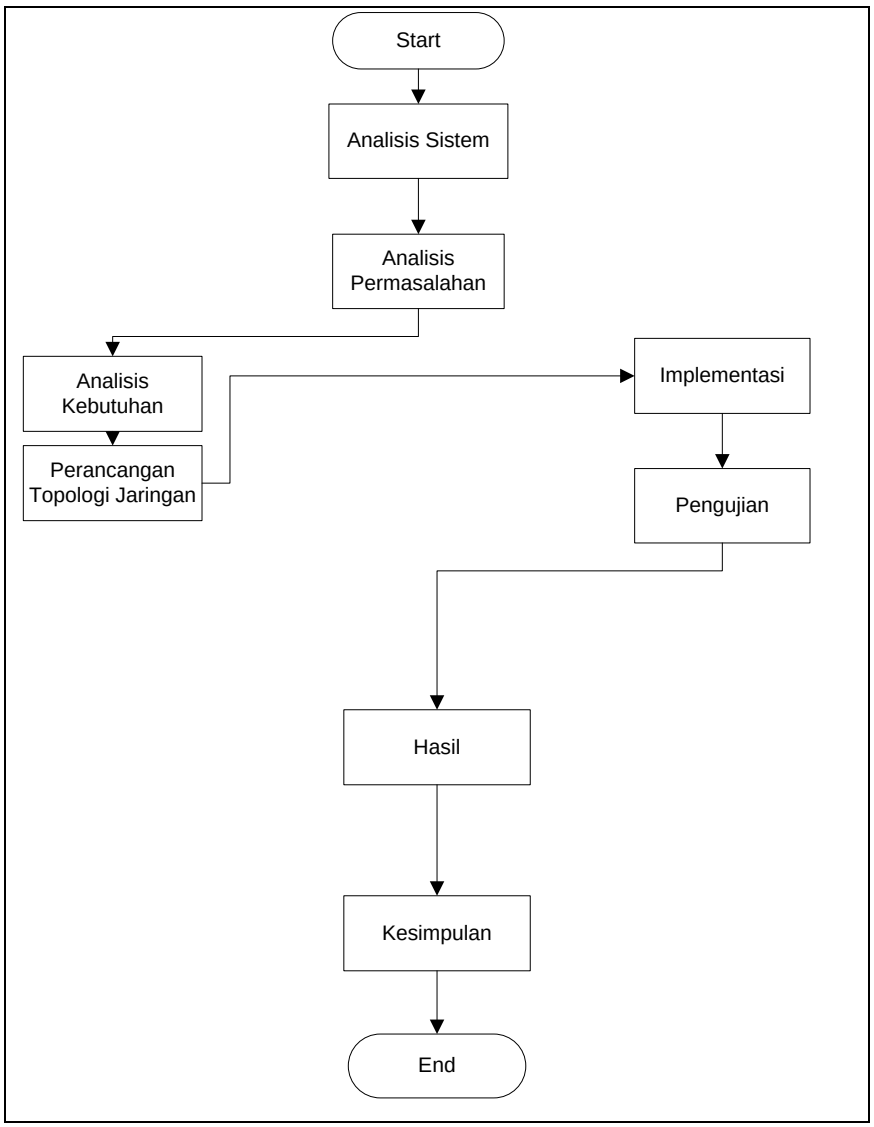
3.	Pratomo dan Ijtihadie	Sistem Deteksi Intrusi Menggunakan N-Gram dan Cosine Similarity	2016	N-Gram dan Cosine Similarity	Penelitian ini mencari kemiripan paket dengan signatured yang ada, apabila kemiripan lebih tinggi maka dianggap paket yang sha apabila tidak maka akan dianggap serangan
4	Mustofa dan Aribowo	Penerapan Sistem Keamanan Honeypot dan IDS pada Jaringan Nirkabel (Hotspot)	2013	- Studi Pustaka - Observasi	Penelitian ini menghasilkan keamanan sistem yang berlapis dengan cara memberikan data palsu apabila ada penyerang yang masuk

					ke sistem
5	Fitriyanti A.Masse, Andi Nurul Hidayat, dan Badrianto	Penerapan Network Intrusion Detection System Menggunakan Snort Berbasis Database Mysql Pada Hotspot Kita	2015	Network Development Life Cycle (NDLC)	Penelitian ini menghasilkan bahwa setiap tindakan yang dilakukan oleh penyerang terhdap jaringan dapat diketahui oleh mesin sensor, sehingga dapat dilakukan pencegahan.
6	M. Agus Munandar, Dr. Herri Setiawan, M.Kom, Lastri Widya Astuti, M.Kom.	Pengenalan Pola Serangan di Jaringan Komputer Menggunakan Metode Signatured Based Studi Kasus Universitas Indo Global Mandiri (UIGM) Palembang	2017	*Signatured Based*	Penelitian ini bertujuan agar dapat mendeteksi serangan

BAB 3
METODE PENELITIAN

3.1 Tahapan Penelitian

Penelitian ini dibagi menjadi beberapa tahapan, dimana pada tahap penelitian ini merupakan langkah kerja penelitian yang dilakukan mulai dari persiapan awal hingga kesimpulan akhir, hal ini dilakukan agar dalam penelitian ini dapat lebih terarah dan terstruktur sehingga mencapai target sesuai waktu yang ditentukan.



Gambar 3.1 Diagram Alir Penelitian

3.2 Tempat Penelitian

Penulis melakukan penelitian yang bertempat di Universitas Indo Global Mandiri (UIGM) Palembang.

3.3 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan pada penelitian ini adalah sebagai berikut :

1. Studi Kepustakaan

   Studi kepustakaan adalah teknik mengumpulkan data melalui buku, dokumen, artikel dan tulisan yang relevan untuk menyusun konsep penelitian serta mengungkap objek penelitian. Studi kepustakaan ini dilakukan dengan banyak melakukan telaah dan pengutipan berbagai teori yang relevan untuk menyusun konsep penelitian.

2. Observasi

   Observasi adalah teknik mengumpulkan data yang dilakukan dengan cara mengadakan pengamatan langsung pada objek penelitian. Observasi ini dilakukan untuk memperoleh berbagai informasi dan data faktual serta memahami situasi dan kondisi dinamis objek penelitian.

3.4 Analisis Sistem

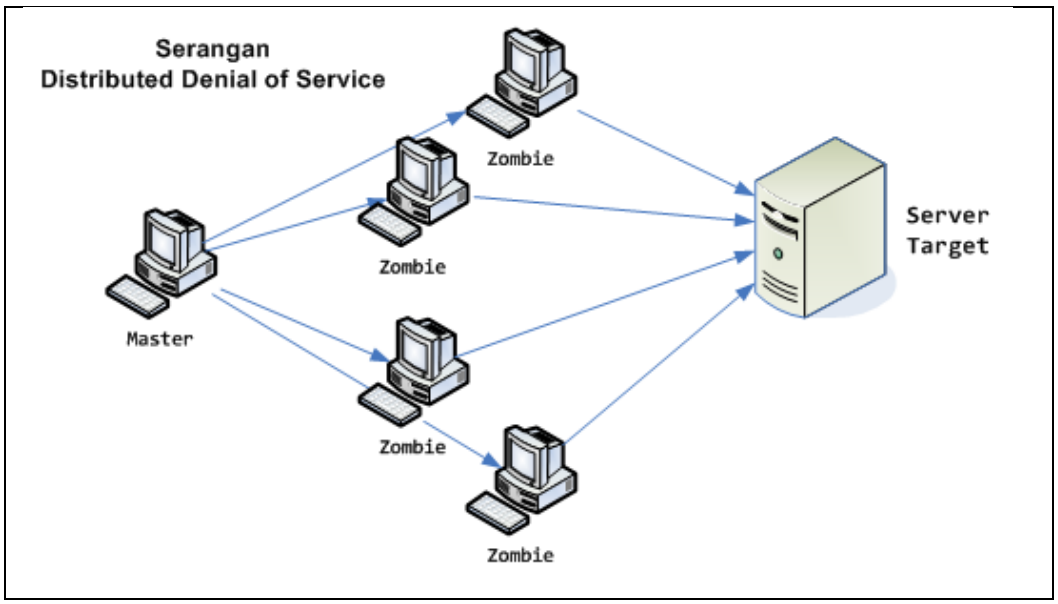
Analisis sistem (*system analysis*) dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapakan sehingga dapat diusulkan perbaikan-perbaikannya. Bab ini akan menguraikan proses analisis sistem keamanan *web server* dengan metode *intrusion detection system* (IDS) dan perancangan keamanan *web server.*

Tahapan analisis sistem dilakukan setelah tahap perencanaan sistem (*system planning*) dan sebelum tahap perancangan sistem (*system design*), tahap

analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan pada tahap ini akan menyebabkan kesalahan di tahap yang selanjutnya.

3.5 Analisis Masalah

Penelitian yang dilakukan di Universitas Indo Global Mandiri yaitu menganalisa masalah keamanan pada *web server*, kendala yang terjadi pada *web server* yang terdapat di Universitas Indo Global Mandiri (UIGM) palembang adalah masih rentannya terhadap serangan-serangan yang dilakukan oleh penyusup. Solusi dari permasalahan tersebut adalah diperlukan suatu cara untuk menjaga keamanan pada *server* sehingga mampu meminimalisirkan serangan-serangan terhadap *server* yang ada di jaringan terutama *web server*. Aplikasi pendeteksian yang diperlukan adalah aplikasi yang dapat mendeteksi adanya serangan pada sistem jaringan yaitu aplikasi *intrusion detection system* (IDS).



Sumber (*https://id.wikipedia.org/wiki/Serangan_DoS*)

Gambar 3.2 Serangan DDoS

Berdasarkan gambar 3.2 diatas dapat diambil kesimpulan bahwa serangan dengan DDoS dengan cara menggunakan banyak *host* penyerang (baik menggunakan komputer yang didedikasikan untuk melakukan penyerangan atau komputer yang dipaksa menjadi *zombie*) untuk menyerang satu buah *host* target

dalam sebuah jaringan. Serangan DDoS biasanya menyerang protokol tujuan dengan cara membanjiri lalu lintas di jaringan, bisa menggunakan protokol *transmission control protocol* (TCP) ataupun *internet control message protocol* (ICMP) yang diserang sehingga lalu lintas yang datang dari pengguna yang terdaftar menjadi tidak dapat masuk ke dalam sistem jaringan. Adapun waktu yang digunakan *attacker* untuk menyerang *server* tujuan terjadi pada saat trafik banyak digunakan, *attacker* akan membanjiri trafik tiap detiknya menjadi lebih besar.

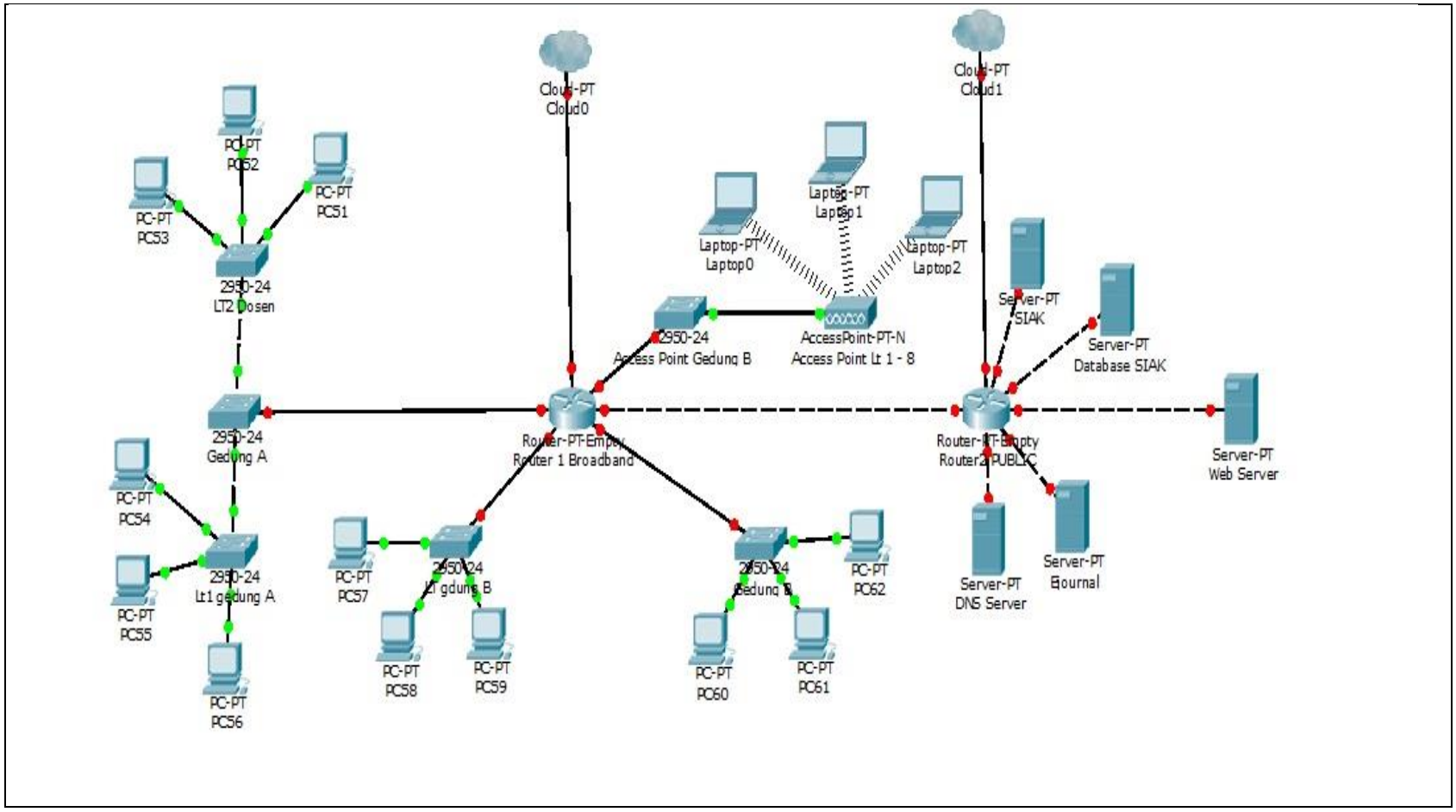
3.5.1 Tampilan Website UIGM

Gambar dibawah ini merupakan tampilan dari *website* dari Universitas Indo Global Mandiri (UIGM) Palembang, dimana *web server* menggunakan *web server apache* dan *database* nya *mysql* serta menggunakan sistem operasi Linux.

Gambar 3.3 Tampilan*Website*UIGM

3.5.2 Topologi Jaringan yang sedang Berjalan di UIGM

Lingkungan Universitas Indo Global Mandiri terdiri atas 2 Gedung yaitu Gedung A untuk Laboratorium dan ruang Dosen, sedangkan Gedung B yang dipergunakan sebagai ruang perkuliahan beserta ruangan *server* yang merupakan tempat penyedia jaringan utama dan juga tempat *server-server*, yang kemudian dialokasikan ke gedung lain agar dapat mengakses internet dan data sesuai kebutuhan dari gedung yang mengakses. Berikut merupakan topologi jaringan yang sedang berjalan di Universitas Indo Global Mandiri.



Sumber (BPT UIGM, 2017)

Gambar 3.4 Topologi yang Berjalan

3.5.3 IP Address di Universitas Indo Global Mandiri

Universitas Indo Global Mandiri menggunakan dua jenis IP, yaitu IP *Public* yang digunakan untuk mengatur *server-server* yang terdapat di UIGM, dan juga menggunakan IP Dinamis yang digunakan untuk akses internet bagi karyawan, dosen dan juga mahasiswa. Penggunaan IP untuk karyawan, dosen dan mahasiswa sudah menggunakan *Virtual* LAN (VLAN), sehingga setiap bagian akan mendapatkan *bandwidth* yang merata. Untuk IP *public* yang digunakan untuk *server* menggunakan IP kelas C yaitu 210.210.130.240/28, sedangkan untuk karyawan dan dosen menggunakan IP kelas C juga yaitu 192.168.x.x/24 dan untuk *hotspot* menggunakan IP kelas A yaitu 10.10.x.x/24

Server di Universitas Indo Global Mandiri mempunyai kegunaan masing-masing. Alamat IP yang digunakan oleh setiap *server* berbeda-beda, berikut tabel server yang digunakan beserta kegunaannya :

Tabel 3.1 IP server dan kegunaan

Nama Server	**IP**	**Keterangan**
Web Server	210.210.130.243	Digunakan untuk informasi mengenai UIGM
SIAK *Server*	210.210.130.244	Digunakan untuk informasi perkuliahan, portal dosen, portal mahasiswa, keuangan, perpustakaan
Ejournal dan *Eprints*	210.210.130.245	Digunakan untuk penyimpanan data jurnal
DNS *Server*	210.210.130.247	Digunakan untuk memetakan alamat IP menjadi nama sesuai domain

3.5.4 Firewall

Analisis *firewall* adalah mekanisme kontrol akses pada level jaringan, aplikasi penghambat yang dibangun untuk memisahkan jaringan privat dan jaringan publik, *firewall* diimplementasikan pada *software, hardware*, ataupun *server acces*, dengan menggunakan suatu *rule/policy* berupa daftar akses.

Teknologi *firewall* yang digunakan dalam permasalahan yang sering terjadi, yaitu *packet filtering gateway*, prinsip nya adalah memperbolehkan atau membatalkan terhadap suatu akses, *packet filtering gateway* didasarkan pada :

1. Paket yang dibolehkan/tidak dibolehkan berdasarkan alamat IP sumber/tujuan.
2. Paket yang dibolehkan/tidak dibolehkan berdasarkan *port* sumber/tujuan.
3. Paket yang dibolehkan/tidak dibolehkan berdasarkan protokol.
4. Paket yang dibolehkan/tidak dibolehkan berdaasarkan *flag* pada protokol tertentu.

3.5.5 Analisis kebutuhan IDS

Keamanan jaringan telah banyak dipakai teknik atau sistem yang telah umum berfungsi untuk mengamankan jaringan yaitu *firewall* dan *kriptografi.Firewall* didesain untuk melewatkan, mengehentikan atau menolak trafik, akan tetapi tidak akan pernah memberikan *alert* atau peringatan terhadap trafikyang mencurigakan. *Kriptografi* berfungsi untuk mengamankan data dari pihak yang mempunyai akses, akan tetapi juga tidak pernah memberikan peringatan terhadap akses yang mencurigakan. Sementara IDS didesain untuk memberitahu kapan aktivitas yang mencurigakan terjadi. Baik teknologi *firewall*, *kriptografi* dan IDS ketiganya perlu diimplementasikan pada jaringan untuk dapat saling melengkapi dalam menjalankan fungsi kemananan jaringan. Oleh kaerena itu, sistem keamanan yang diharapkan selain dapat mencegah penyusupan juga haruslah dapat melakukan deteksi penyusupan dari luar sistem dan sekaligus dapat melakukan deteksi penyalahgunaan *account* dari sistem luar.

IDS akan membantu administrator jaringan dengan cara meningkatkan kemampuan penemuan resiko yang membahayakan sistem secara cepat dan *real time*, dengan demikian, fungsi IDS adalah memberi peringatan kepada administrator atas serangan yang terjadi sehingga administrator dapat memperoleh keuntungan sebagai berikut :

1. Mencegah resiko timbulnya masalah.

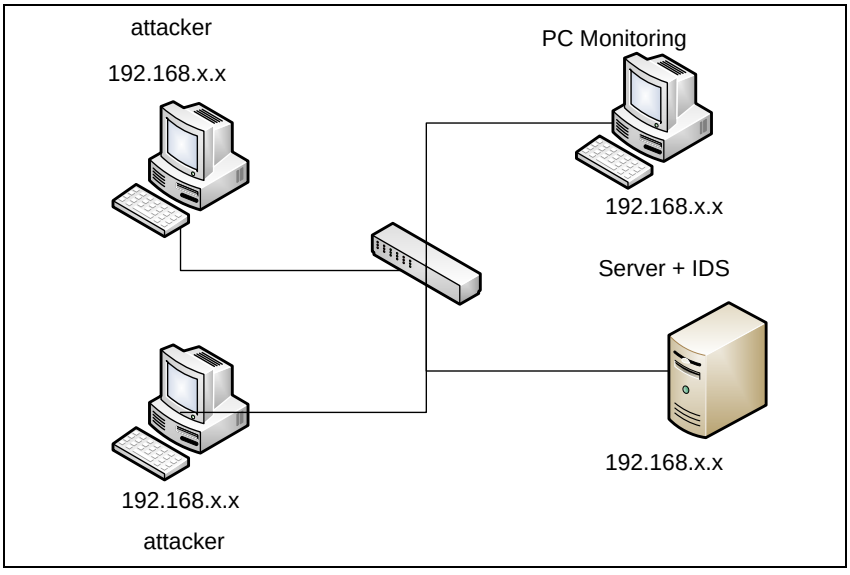
2. Mendeteksi serangan dan pelanggaran keamanan lain yang tidak dicegah oleh perangkat keamanan lainnya, biasanya penyusupan berlangsung dalam tahapan yang bisa diprediksi. Tahapan pertama adalah *probing*, atau *ekploitasi* pencarian titik masuk. Pada sistem tanpa IDS, penyusup memiliki kebebabsan melakukannya dengan resiko kepergok lebih kecil. IDS yang mendapati *probing* bisa melakukan blok akses dan memberitahukan tenaga keamanan yang selanjutnya mengambil tindakan lanjutan.
3. Mendeteksi usaha penyusupan yang berkaitan dengan serangan misalnya *probing* dan aktivitas DoS dan DDoS.
4. Mendokumentasikan ancaman yang pernah terjadi dan mungkin terjadi lagi pada jaringan. IDS akan mampu menggolongkan ancaman baik dari dalam maupun dari luar jaringan sehingga membantu membuat keputusan untuk alokasi sumber daya keamanan jaringan.
5. Bertindak sebagai pengendali kualitas pada administrasi dan perancangan keamanan pada jaringan. Saat IDS dijalankan dalam waktu tertentu, pola dari pemakaian sistem dan masalah yang ditemui bisa nampak, sehingga akan membantu pengelolaan keamanan dan memperbaiki kekurangan sebelum menyebabkan insiden.
6. Memberikan informasi yang berguna mengenai penyusupan yang terjadi, peningkatan diagnosa, *recovery*, dan perbaikan dari faktor penyebab. Meskipun pada IDS yang bertipe *passiver respone* tidak melakukan blok serangan, tetapi IDS ini masih bisa mengumpulkan informasi yang relevan mengenai serangan, sehingga membantu penanganan insiden dan *recovery*.

Semua teknologi keamanan memiliki resiko. *Firewall* memiliki resiko untuk dapat dipenetrasi, sementara *enkripsi* memiliki resiko untuk dibobol atau *didecrypt*, maka IDS juga memiliki resiko kesalahan dalam mendeteksi jaringan. Resiko kesalahan dalam mendeteksi jaringan ini dikenal dengan *false positive*. Apabila suatu IDS menghasilkan banyak *false positive* maka penggunaan IDS

pada suatu jaringan akan menyibukkan admin dalam merespon banyaknya alarm yang salah melakukan pendeteksian yang dihasilkan setiap harinya.

3.5.6 Skenario Penyerangan

Skenario penyerangan pada penelitian ini yaitu menggunakan *syn flooding attack*,dimana dalam sebuah serangan *syn flooding*, si penyerang akan mengirimkan paket-paket syn ke dalam *port-port* yang sedang berada dalam keadaan *listening* yang berada dalam *host* taget. Normalnya, paket-paket syn yang dikirimkan berisi alamat sumber yang menunjukkan sistem aktual, tetapi paket-paket syn dalam serangan ini memiliki alamat sumber yang tidak menunjukkan sistem aktual. Ketika target menerima paket syn yang telah dimodifasi tersebut, target akan merespon dengan sebuah paket syn/ack yang ditujukan ke alamat yang tercantum di dalam syn paket yang ia terima, dan kemudian akan menunggu paket ack sebagai balasan untuk melengkapi proses pembuatan koneksi. Tetapi, karena alamat sumber dalam paket syn yang dikirimkan penyerang tidak valid, paket ack tidak akan pernah datang ke target, dan *port* yang menjadi target serangan akan menunggu hingga waktu pembuatan koneksi *timed out*. Jika sebuah *port* yang *listening* menerima banyak paket-paket syn, maka *port* tersebut merespon dengan paket syn/ack sesuai dengan jumlah paket syn.



Gambar 3.5 Skenario Syn Flooding

Berdasarkan pada gambar 3.5 cara kerja serangan adalah salah 1 (satu) pc penyerang akan menyerang ke *server* IDS baik itu menyerang protokol TCP, HTTP maupun UDP, dimana saat terjadinya serangan admin akan memantau pada pc monitoring apabila terjadi serangan, ataupun cara kerja serangan dengan 2 (dua) penyerang yang langsung menyerang ke *server*, dan admin dapat memantau melalui pc monitoring yang ada, sehingga apabila terjadi nya serangan IDS akan memberikan *alert* yang dapat dilihat admin melalui pc monitoring yang sudah berbentuk *webbase*.

3.5.7 Analisa Proses Pencocokan Pola

Pencocokan pola (*pattern matching*) adalah suatu metode yang digunakan untuk mencocokkan suatu pola tertentu dengan suatu kumpulan kata atau *string*. *Regular expression* merupakan suatu konsep pencocokan pola (*pattern matching*) di dalam suatu *string*, dalam implementasinya *regular expresion* menjadi suatu pola karakter yang banyak didukung oleh banyak aplikasi dan bahasa pemrograman. Konsep *regular expression* yang akan mengenali pola dari karakter atau kata dan kemudian melakukan *substitusi*, *regular expression* juga bisa digunakan untuk menentukan apakah sebuah masukan yang diberikan sesuai dengan pola yang sudah ditentukan pada sebuah sistem.

3.6 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan proses ide dan evaluasi permasalahan-permasalahan yang ada sehingga dapat dikembangkan sistem baru yang sesuai dengan yang diharapkan.

3.6.1 Kebutuhan Perangkat Keras (*Hardware*)

Perangkat keras (*hardware*) yang digunakan pada penelitian ini adalah sebagai berikut :

1. Satu unit komputer sebagai *Web Server* dan IDS
2. Satu unit Komputer sebagai *attacker.*
3. Satu unit Komputer/laptop sebagai *monitoring*.

3.6.2 Kebutuhan Perangkat Lunak (*Software*)

Perangkat lunak (*software*) yang digunakan pada penelitian ini adalah sebagai berikut :

1. Sistem Operasi Linux.
2. Aplikasi Snort.
3. Apache *web server*.
4. *Web Base* (*Basic Analysis and security engine*).
5. *Database mysql server*.

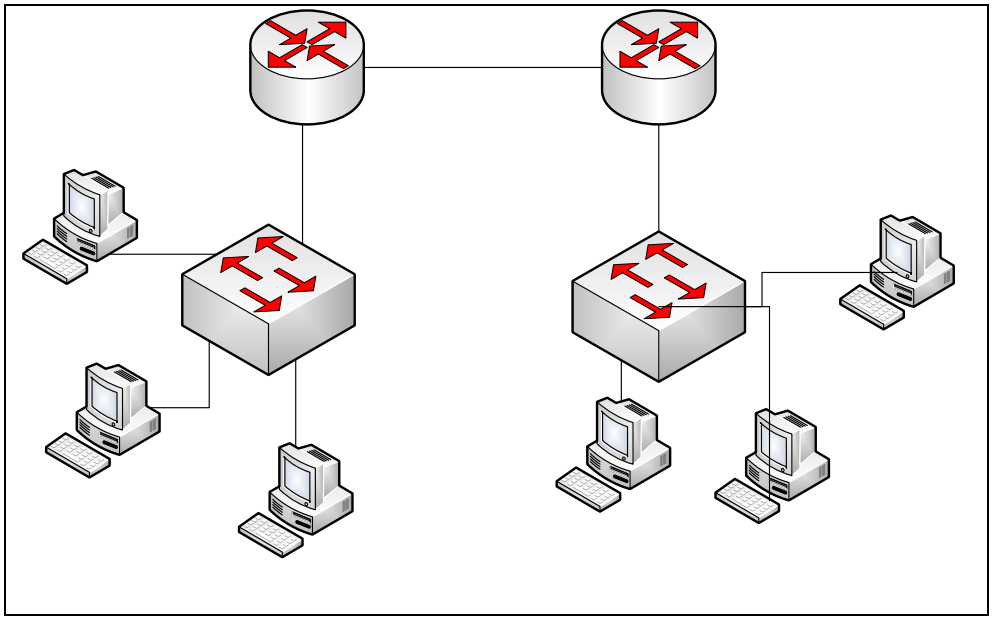
3.7 Perancangan Topologi Jaringan

Perancangan topologi jaringan merupakan proses yang akan dibutuhkan dalam menerapkan IDS, dimana meliputi topologi jaringan yang diusulkan, skema kerja dari IDS serta *flowchart* cara kerja IDS dan *rule snort*.

3.7.1 Topologi Jaringan yang diusulkan

Topologi yang diusulkan di UIGM sama dengan topologi yang sedang berjalan, akan tetapi perbedaannya adalah ditambahkan aplikasi monitoring *Intrusion Detection System* (IDS) dengan menggunakan *snort* yang berfungsi untuk mendeteksi serangan yang dilakukan pada *web server* yang terhubung pada jaringan komputer di UIGM.

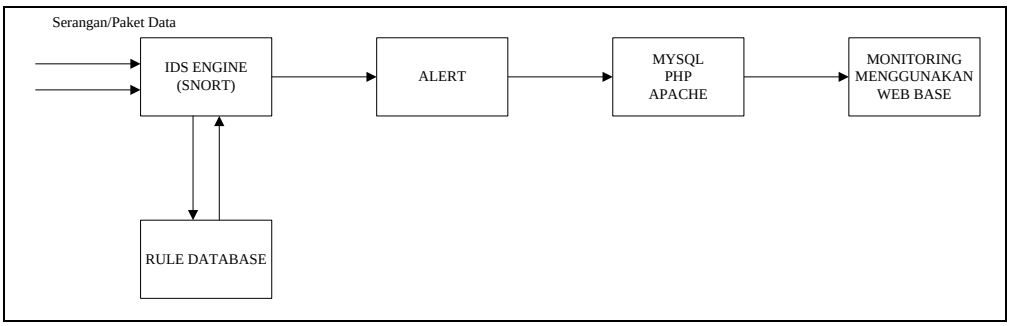
Topologi yang digunakan menggunakan topologi jaringan bertipe *hybrid*, dimana topologi ini adalah topologi yang tersusun/terhubung dari berbagai topologi lainnya. Gambar 3.5 menunjukkan topologi dari jenis *hybrid*.



Gambar 3.6 Topologi Hybrid

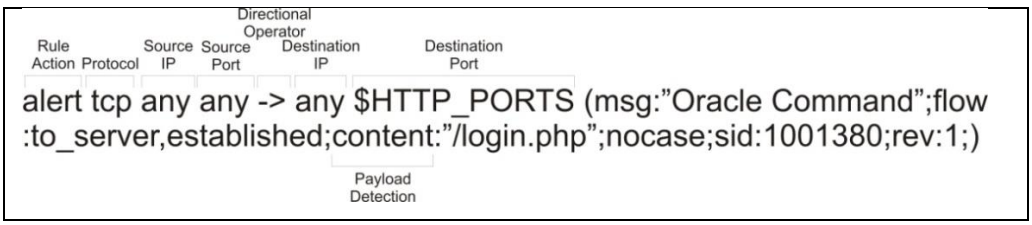
3.7.2 Skema Kerja IDS

Knowledge based/Signatured Based IDS dapat mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkannya dengan *database rule* IDS, yang berisi *signatured-signatured* paket serangan. Apabila paket data mempunyai pola yang sama dengan salah satu pola yang ada di *database rule* IDS, maka paket tersebut dianggap sebagai serangan, dan demikian juga sebaliknya, jika paket data tersebut sama sekali tidak mempunyai pola yang sama dengan pola yang terdapat di *database rule* IDS, maka paket data tersebut dianggap bukan merupakan serangan, kemudian IDS *engine* akan membaca *alert* dari IDS.



Gambar 3.7 Cara Kerja IDS

Rules Snort merupakan *database* yang berisi pola-pola serangan yang berupa *signature* jenis serangan, *rule snort* ini harus secara rutin di *update* agar ketika terjadi pola serangan baru, *snort* dapat mendeteksi pola tersebut. Penulisan *rule snort* mempunyai aturan yaitu *rule* harus ditulis dalam satu baris (*single line*). *Rule snort* dibagi menjadi dua bagian, yaitu *rule header* dan *rule options*, *rule header* berisi *rule action*, *protocol, source* dan *destination port*. Sementara *rule option* berisi pesan peringatan dan informasi dimana seharusnya paket tersebut diletakkan. Contoh *rule snort* dapat ditunjukkan pada gambar 3.8



Sumber (Affandi et al., 2013)

Gambar 3.8Rule Snort

1. *Rule Header*

   Bagian dari *rule header* menunjukkan pengertian sebagai berikut :

   a. *Rule actions*

      Rule actions akan memberitahukan kepada *snort* apa yang harus dilakukan ketika menemukan paket yang sesuai dengan *signature* yang ada pada *rule snort*. Aksi yang dapat dilakukan adalah *alert*, *log* dan *pass*.

   b. *Protocols*

      Isi dari *rule header* selanjutnya adalah jenis protokol yang digunakan, misal TCP, UDP dan ICMP.

   c. IP Addrress

      Penulisan nomor IP pada *rule snort* ditulis lengkap beserta *netmask* yang digunakan atau dapat juga menggunakan kata “any” untuk banyak alamat IP.

   d. *Port Number*

      Port number dapat dituliskan dengan angka yang menunjukkan jenis protokol ataupun dengan kata “any” untuk semua *port* yang tersedia ataupun dengan *range port* seperti 1:1023.

   e. *Directions Operator*

      Tanda -> merupakan orientasi atau *direction*. Sebelah kiri adalah *source host* dan sebelah kanan adalah *destination host*. Orientasi juga dapat bersifat *bidirectional* dengan tanda <>.

2. *Rule options*

   Rule options dapat dipisahkan dengan tanda : pada saat penulisan. Jenis *options* yang digunakan pada *snort* terdapat 15 (lima belas) kata kunci, yaitu :

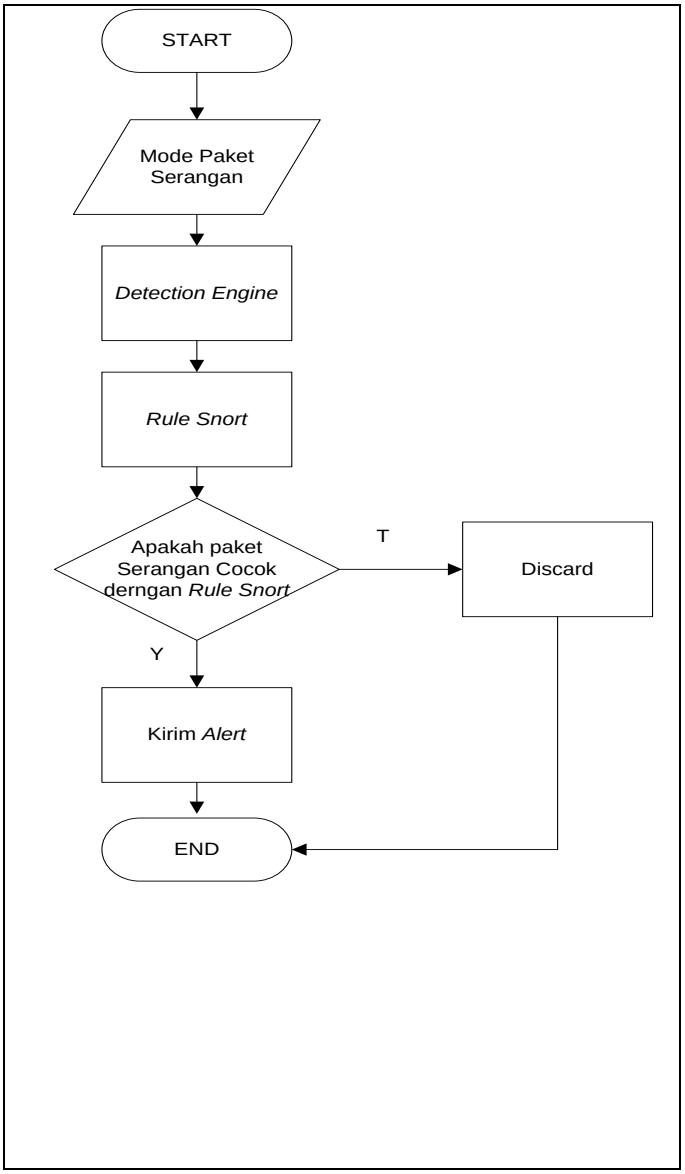
   a. Msg : menuliskan pesan dari *alert* dan paket *log*.
   b. Logto : memasukkkan *log* ke standar *output file*.
   c. Minfrag.
   d. Ttl : nilai *time to life* pada IP *header*.

e. Dsize : ukuran nilai data yang ditangkap.
f. Content : pola dari data yang dianalisa.
g. Offset.
h. Depth.
i. Flags : nilai TCP *flags*.
j. Seq : nilai TCP *sequence*.
k. Ack : nilai TCP *acknowledge*.
l. Itype : nilai ICMP *type*.
m. Icode : nilai ICMP *code*.
n. Session.

Berdasarkan *rule* diatas IDS *snort* menghukumi apakah sebuah paket data dianggap sebagai penyusupan/serangan atau bukan, paket data dibandingkan dengan *rule* IDS, jika terdapat dalam *rule*, maka paket data tersebut dianggap sebagai penyusupan/serangan begitupula sebaliknya jika tidak terdapat dalam *rule* maka dianggap bukan penyusupan/serangan.

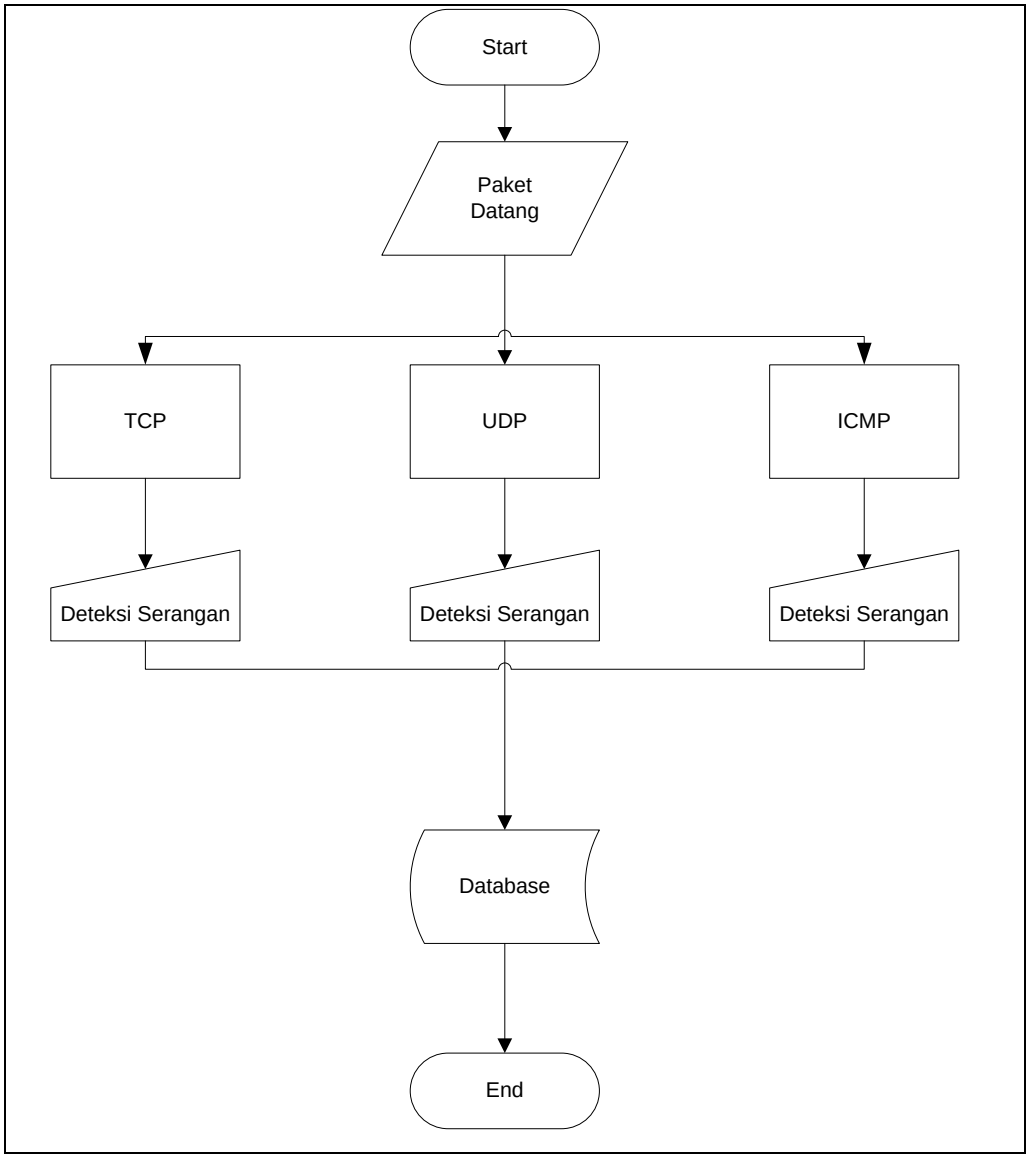
3.7.3 Diagram Alir IDS

Gambar 3.9merupakan proses IDS, dimulai dari paket data yang memasuki *interface* jaringan yang sudah dikonfigurasi dalam *snort*. Paket data tersebut akan dibaca oleh *snort engine* untuk kemudian dicocokkan dengan *signature* yang ada di dalam *rules snort*, jika paket data tersebut sesuai dengan *signature* yang ada pada *rules snort*, maka *snort* akan mengganggap itu sebagai sebuah intrusi/serangan dan *snort* akan menyimpan *alert* tersebut kemudian *firewall* akan memblok IP tersebut, namun jika paket data tersebut bukan merupakan intrusi, maka paket data tersebut akan diteruskan.



Gambar 3.9 Flowchart Proses IDS dan Bloking Otomatis

Pada gambar 3.10 menunjukkan diagram alir dari proses pembuatan *rule snort*, dimana pada saat ada paket datang, baik itu dalam bentuk protokol *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP), maupun *Internet Control Message Protocol* (ICMP), apabila protokol yang masuk tersebut dianggap mencurigakan maka selanjutnya akan dibuat *rules* yang kemudian dimasukkan kedalam *database snort*.



Gambar 3.10*Flowchart* Proses *Rule Snort*

Pseudocode proses rule snort

Input : pola {p1,p2,....,pn}
Paket data
Output : serangan ddos
Paket normal
If paket = pola then
Serangan ddos

Else

Paket normal

End IF

3.7.4 Variabel Penelitian

Variabel diperlukan untuk menentukan konsep, indikator dan variabel-variabel yang terkait dengan penelitian, sehingga pengujian dapat dilakukan dengan benar. Variabel yang diukur pada penelitian ini yaitu variabel *accuracy*. Rumus yang digunakan dalam menghitung *accuracy* adalah :

$$\text{accuracy} = \frac{Jumlah\ data\ pengujian\ yang\ benar}{Jumlah\ data\ yang\ di\ uji} \text{ x } 100\%$$

BAB 4
HASIL DAN PEMBAHASAN

4.1 Kasus Uji

Pada bagian ini dijelaskan kasus uji yang akan dilakukan dalam penerapan pengenalan pola serangan di jaringan, dimana kasus uji yang dilakukan bersifat simulasi dengan menggunakan 2 (dua) buah pc yang bertindak sebagai penyerang, 1 (satu) PC sebagai IDS dan 1 PC monitoring, adapun pengujiannya menggunakan 5 (lima) buah rule yang dapat dilihat pada tabel dibawah ini.

Tabel 4.1 Kasus Uji Jenis Serangan

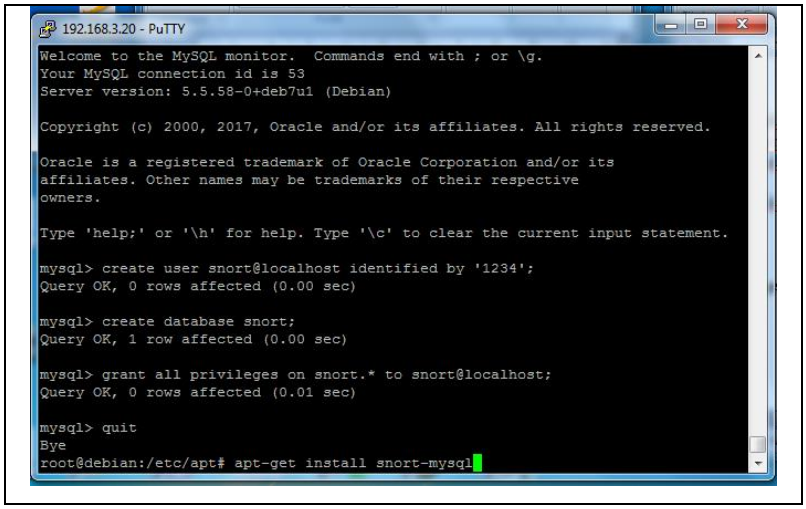
No.	**Rule**	**Serangan**		
1.	1	HTTP	TCP	UDP
2.	2	HTTP	TCP	UDP
3.	3	HTTP	TCP	UDP
4.	4	HTTP	TCP	UDP
5.	5	HTTP	TCP	UDP

Pada tabel 4.1 percobaan simulasi akan menggunakan 5 buah rule dimana setiap rule akan diuji untuk mendeteksi serangan-serangan baik menggunakan protokol HTTP, TCP, maupun UDP.

4.2 Implementasi

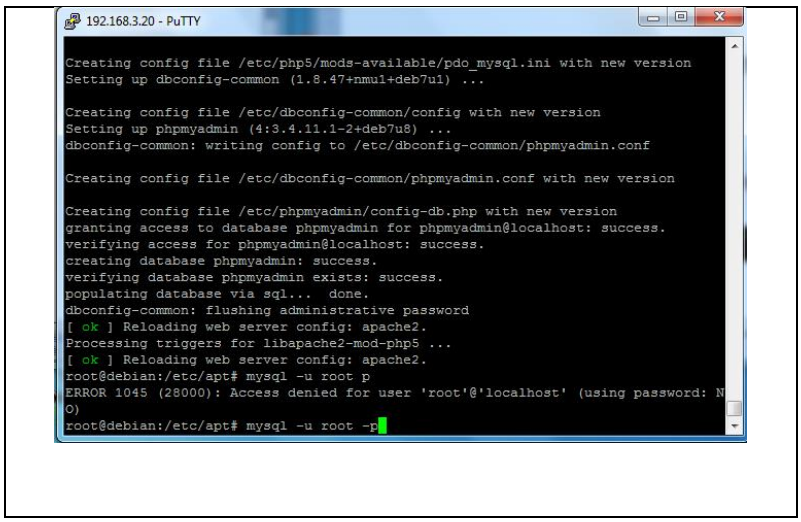
Proses instalasi linux dilakukan sebagai tahap awal dalam proses implementasi, setelah selesai proses instalasi sistem operasi dilanjutkan dengan instalasi komponen-komponen tambahan atau pendukung pendeteksi *snort*. Adapun paket pendukung IDS atau aplikasi yang diinstall adalah *mysql-server, apache2, php5, phpmyadmin* dengan menggunakan perintah *apt-get install*.

Gambar 4.1 memperlihatkan proses instalasi sensor *snort-mysql* dengan menggunakan perintah *apt-get install snort-mysql*.



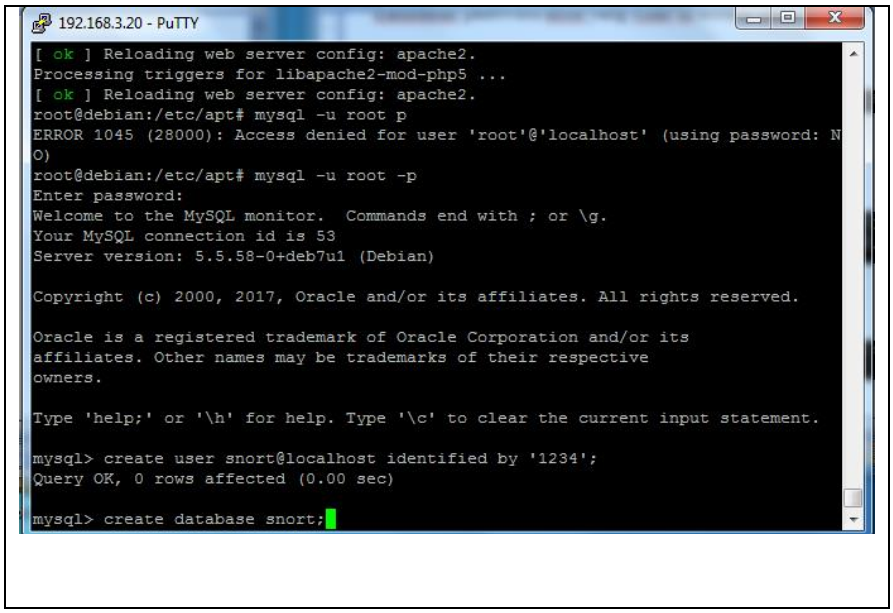
Gambar 4.1 instalasi snort mysql

Langkah selanjutnya setelah melakukan instalasi snort-mysql dilanjutkan dengan membuat *databasemysql server snort*. Pembuatan *database* ini berfungsi sebagai tempat penyimpanan informasi apabila terjadi serangan terhadap *server*.Pembuatan *database* ini yaitu dengan menggunakan perintah mysql –u root –p seperti yang terlihat pada gambar 4.2 dimana perintah tersebut untuk masuk kedalam *database mysql* dengan menggunakan *user* dan *password* yang telah dibuat, dimana pada percobaan ini menggunakan *username roo*t dan *password* “12345”.



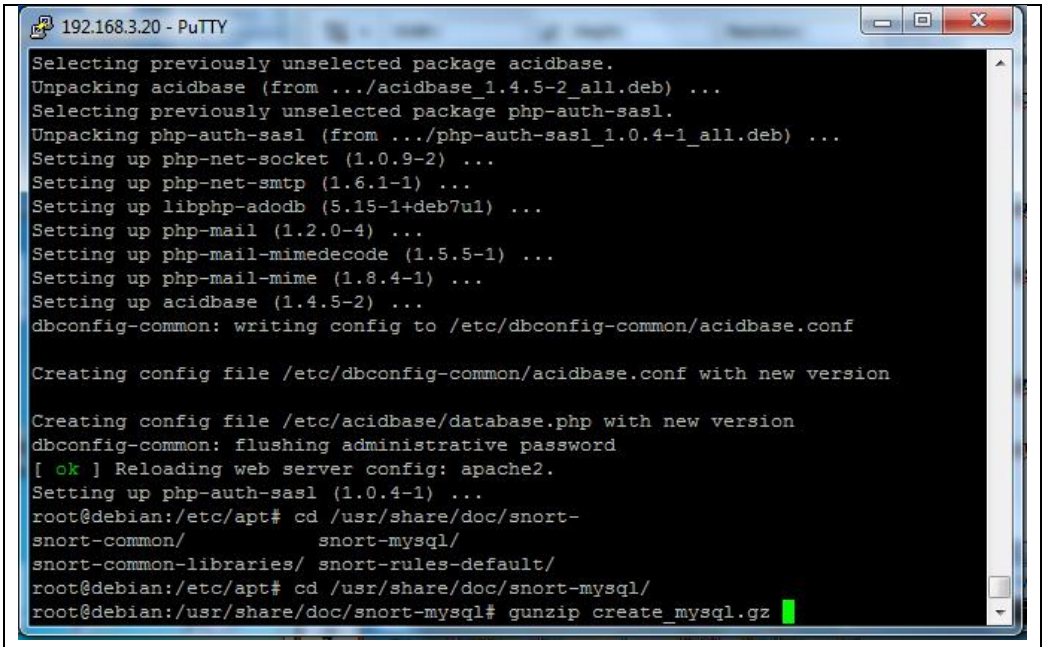
Gambar 4.2 login mysql

Langkah berikutnya setelah masuk kedalam *database mysql* selanjutnya adalah membuat *database snort* dengan menggunakan perintah *create database snort*, yang terlihat pada gambar 4.3.



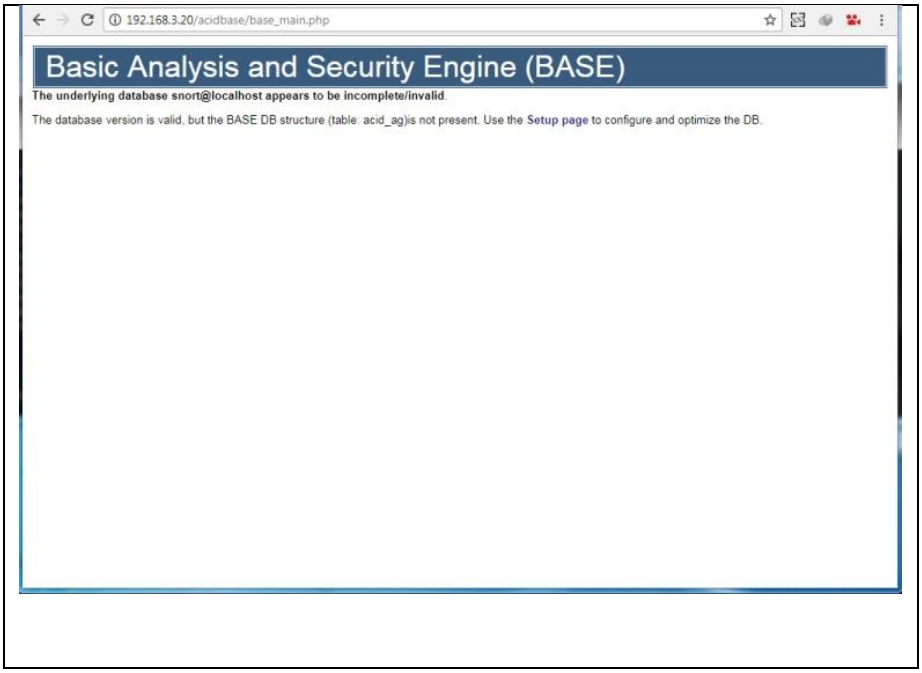
Gambar 4.3 membuat user pada database snort

Setelah membuat *database* dengan nama *snort*, langkah selanjutnya terlihat pada gambar 4.4 yaitu membuat tabel *snort* didalam *database snort* yang telah dibuat sebelumnya.



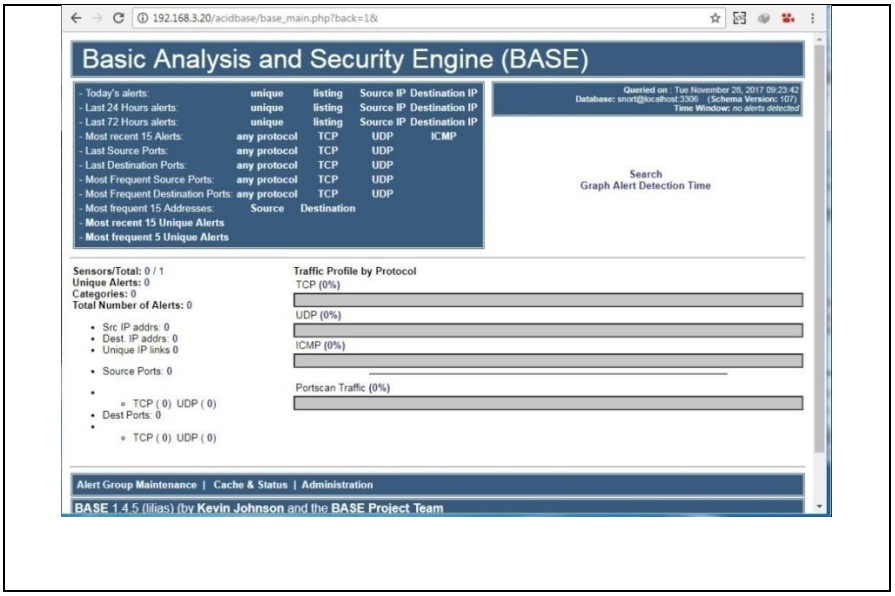
Gambar 4.4 membuat tabel snort

Setelah melakukan instalasi *database snort* langkah selanjutnya adalah menginstal dan konfigurasi BASE, dimana BASE ini betujuan untuk mempermudah *user* dalam memantau apabila terjadi serangan, karena BASE sendiri dalam bentuk *web base.*Cara Instalasi dan konfigurasi BASE dapat dilihat pada gambar 4.5 dan cara menginstallnya melalui *client* dengan cara buka *browser* dari *client*, kemudian ketikkan alamat IP *server* IDS, dimana pada percobaan ini menggunakan IP 192.168.3.20/acidbase.Pertama klik *setup page* untuk memulai instalasi BASE, selanjutnya *create BASE AG*, dan tunggu hingga proses instalasi selesai.



Gambar 4.5 konfigurasi BASE

Pada gambar 4.6 dapat dilihat tampilan BASE yang telah diinstall, dimana pada tampilan ini mempermudah dalam melihat jika terjadi serangan, pada tampilan ini terdapat menu *traffic* apabila terjadi sesuatu, yang meliputi TCP, UDP, ICMP, dan *Portscan Traffic*.



Gambar 4.6 tampilan base

4.3 Pengujian

Pengujian ini penulis menggunakan 5 (lima) buah *rule* yang akan dicoba. Adapun *rule* yang digunakan adalah sebagai berikut :

1. Rule 1 (satu)

```
Alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"Slowloris DoS tool flood"; detection_filter:track by_src, count 20, seconds 20; metadata:service http; classtype:attempted-dos; sid:1234572; rev:2;)
```

Rule ini akan mengenali pola serangan dimana pada *rule* ini akan memberikan keterangan bahwa *snort* akan memberikan peringatan pada paket data dengan protokol berjenis TCP, *rule* ini akan memberikan pesan Slowloris DoS tool flood dengan nomor id 1234572 versi 2 (dua), pada *rule* ini akan mencatat apabila selama 20 detik terdapat minimal 20 peristiwa berdasarkan IP sumber yang sama, setelah melakukan pencatatan waktu akan diulang kembali dari 0.

2. Rule 2 (dua)

```
Alert tcp any any -> any any (msg:"Possible SYN flood"; classtype:attempted-dos; sid:1999999; flags:S; flow: stateless; detection_filter: track by_dst, count 50, seconds 10;)
```

Rule ini akan mengenali pola serangan dimana pada *rule* ini akan memberikan keterangan bahwa *snort* akan memberikan peringatan pada paket data dengan protokol berjenis tcp, *rule* ini akan memberikan pesan Possible SYN flood dengan nomor id 1999999, *rule* ini akan mencatat apabila selama 10 detik terdapat minimal 50 peristiwa berdasarkan IP sumber yang sama dan *rule* ini akan mendeteksi jika ada *flags* yang berjenis Syn.

3. Rule 3 (tiga)

```
Alert tcp any any ->any any (msg:”TCP SYN flood attack detected”; flags:S; threshold: type threshold, track by_dst, count 20,seconds 60; sid:5000001;rev:1;)
```

Rule ini akan mengenali pola serangan dimana pada *rule* ini akan memberikan keterangan bahwa *snort* akan memberikan peringatan pada paket data dengan protokol berjenis tcp, *rule* ini akan memberikan pesan TCP SYN flood attack detected dengan nomor id 5000001 dan menggunakan versi 1 (satu), *rule* ini akan mencatat apabila selama 60 detik terdapat minimal 20 peristiwa berdasarkan IP sumber yang sama.*Flags* adalah kontrol bit yang menandakan atau menunjukkan *connection states* yang berbeda atau informasi begaimana sebuah paket harus ditangani. Pada *rule* 3 ini*snort* akan memberikan *alert* apabila terdapat paket dengan *flags* S (SYN). *Threshold* digunakan untuk mengurangi jumlah *event* yang dicatat dari sebuah *rule*, perintah *threshold* membatasi jumlah peristiwa yang dicatat selama interval waktu tertentu

4. Rule 4 (empat)

```
Alert udp any any -> any 80 (msg:"SLR - LOIC DoS Tool UDP Mode)"; content:"|65 73 75 64 65 73 75 64 65 73 75 7e|"; threshold: type threshold, track by_src, count 100 , seconds 5; sid:1234571; rev:1; )
```

Rule ini akan mengenali pola serangan dimana pada *rule* ini akan memberikan keterangan bahwa *snort* akan memberikan peringatan pada paket data dengan protokol berjenis udp, rule ini akan memberikan pesan SLR – LOIC DoS Tool UDP Mode dengan nomor id 1234571 dengan menggunakan versi 1, *rule* ini akan mencatat apabila selama 5 detik terdapat minimal 100 peristiwa berdasarkan IP

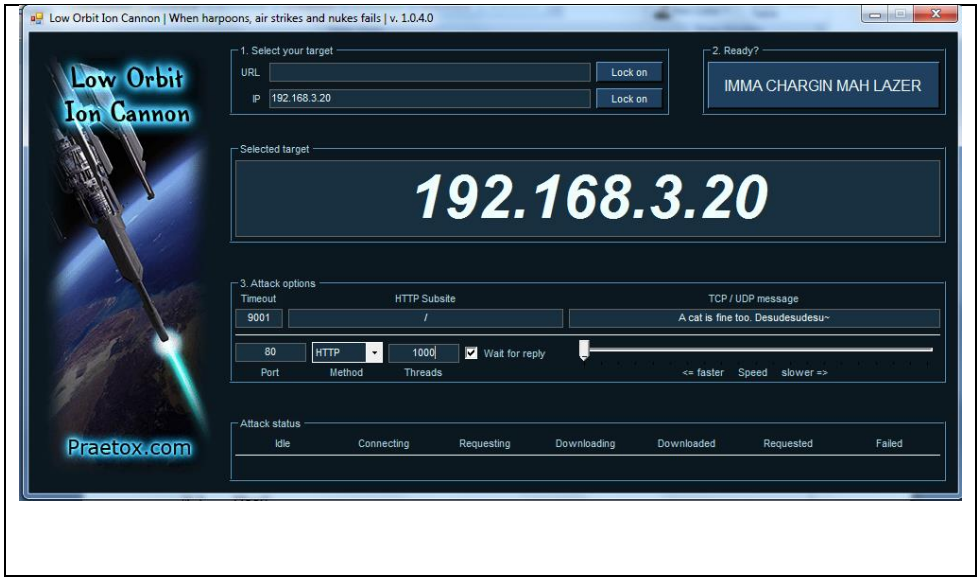
sumber yang sama dan akan mengenali berdasarkan *content* dan juga *threshold.*

5. Rule 5 (lima)

```
Alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (msg:"SLR – LOIC DoS Tool – Behavior Rule (tracking/threshold)"; threshold: type threshold, track by_src, count 100, seconds 5; reference: url, www.simpleweb.org/reports/loic-report.pdf ; classtype:misc-activity; sid:1234590; rev:1;)
```

Rule ini akan mengenali pola serangan dimana pada *rule* ini akan memberikan keterangan bahwa *snort* akan memberikan peringatan pada paket data dengan protokol berjenis tcp, *rule* ini akan memberikan pesan SLR – LOIC DoS Tool – Behavior Rule, *rule* ini akan mencatat apabila selama 5 detik terdapat minimal 100 peristiwa berdasarkan IP sumber yang sama.

Sedangkan untuk serangan menggunakan *software* LOIC dan menggunakan *software* Hping. Gambar 4.7 memperlihatkan tampilan utama dari software LOIC, untuk penggunaannya masukkan alamat url atau IP server yang akan diserang, kemudian klik *lock on*, lalu pilih *protocol* yang akan diserang dan berapa kali *threads* yang akan dikirim, kemudian klik IMMA CHARGIN MAH LAZER.



Gambar 4.7 tampilan LOIC

4.4 Hasil dan Pembahasan

Hasil dan pembahasan merupakan proses yang didapatkan setelah melakukan proses uji coba yang meliputi hasil percobaan dan pembahasan. Pada hasil ini akan dilakukan proses *training* dan proses uji, dimana proses *training* adalah langkah awal untuk melihat apakah IDS mampu mengenali serangan atau tidak. Dataset yang digunakan dapat berupa file pcap dan tcpdump, dan juga dapat menggunakan data serangan khusus, dimana pada penelitian ini menggunakan data serangan khusus yang dilakukan menggunakan serangan secara langsung ke *server*.

4.4.1 Hasil Penelitian

Pada tabel 4.2 – 4.11 terdapat pengujian yang dilakukan dengan menggunakan 5 (lima) *rule snort*.

Tabel 4.2 Hasil Training Rule 1

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi

3.	LOIC (UDP)	Terdeteksi
4.	HPING3	Terdeteksi
5.	LOIC (HTTP)	Terdeteksi
6.	LOIC (TCP)	Terdeteksi
7.	LOIC (UDP)	Tidak Terdeteksi
8.	HPING3	Tidak Terdeteksi
9.	LOIC (HTTP)	Terdeteksi
10.	LOIC (TCP)	Terdeteksi
11.	LOIC (UDP)	Tidak Terdeteksi
12.	HPING3	Tidak Terdeteksi
13.	LOIC (HTTP)	Terdeteksi
14.	LOIC (TCP)	Terdeteksi
15.	LOIC (UDP)	Tidak Terdeteksi
16.	HPING3	Tidak Terdeteksi
17.	LOIC (HTTP)	Terdeteksi
18.	LOIC (TCP)	Terdeteksi
19.	LOIC (UDP)	Tidak Terdeteksi
20	HPING3	Tidak Terdeteksi

Pada tabel 4.2 didapatkan hasil dari training menggunakan rule 1, maka nilai akurasi yang didapatkan adalah sebagai berikut :

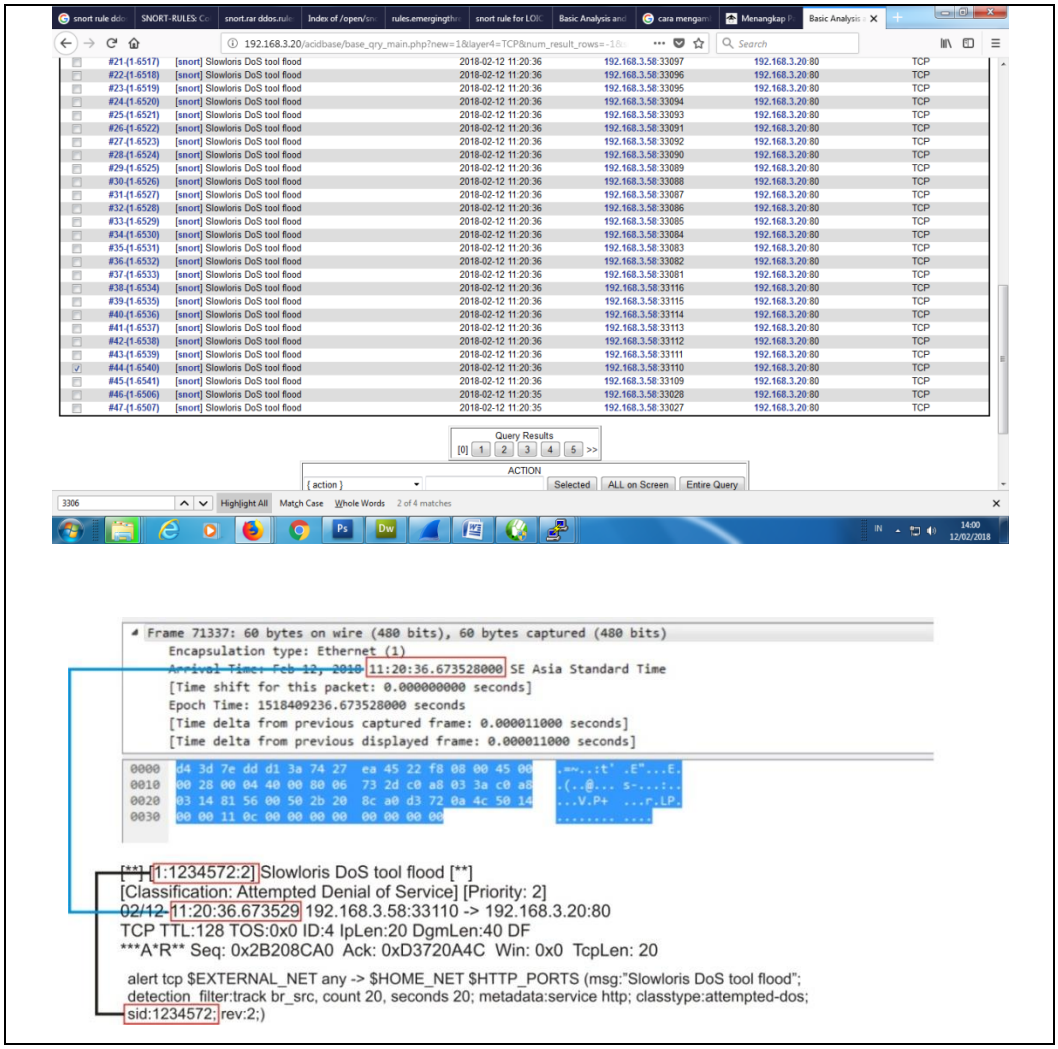
$$Accuracy = \frac{12}{20} x\ 100\ \% = 60\ \%$$

Tabel 4.3 Hasil Uji Rule 1

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Tidak Terdeteksi

Pada tabel 4.3 didapatkan hasil dari uji menggunakan rule 1, maka nilai akurasi yang didapatkan adalah sebagai berikut :

$$Accuracy = \frac{2}{4} x\ 100\ \% = 50\ \%$$



Gambar 4.8 Deteksi Rule 1

Gambar 4.8 menunjukkan *alert* yang dihasilkan oleh *snort*, dimana hasil yang ditampilkan snort dibandingkan dengan hasil yang diambil menggunakan *tcpdump* untuk melihat waktu yang dihasilkan apakah sama dengan waktu yang diberikan oleh IDS, dimana pada *rule* ini waktu yang dihasilkan *tcpdump* dan waktu yang dihasilkan acidbase menunjukkan waktu yang sama, akan tetapi *rule*

ini tidak mampu mendeteksi untuk serangan dengan protokol UDP dan menggunakan *software* HPING.

Tabel 4.4 Hasil Training Rule 2

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Tidak Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Tidak Terdeteksi
5.	LOIC (HTTP)	Terdeteksi
6.	LOIC (TCP)	Tidak Terdeteksi
7.	LOIC (UDP)	Tidak Terdeteksi
8.	HPING3	Tidak Terdeteksi
9.	LOIC (HTTP)	Terdeteksi
10.	LOIC (TCP)	Terdeteksi
11.	LOIC (UDP)	Tidak Terdeteksi
12.	HPING3	Tidak Terdeteksi
13.	LOIC (HTTP)	Terdeteksi
14.	LOIC (TCP)	Terdeteksi
15.	LOIC (UDP)	Tidak Terdeteksi
16.	HPING3	Terdeteksi
17.	LOIC (HTTP)	Terdeteksi
18.	LOIC (TCP)	Terdeteksi
19.	LOIC (UDP)	Tidak Terdeteksi
20.	HPING3	Terdeteksi

Pada tabel 4.4 didapatkan hasil dari training menggunakan rule 2, maka nilai akurasi yang didapatkan adalah sebagai berikut :

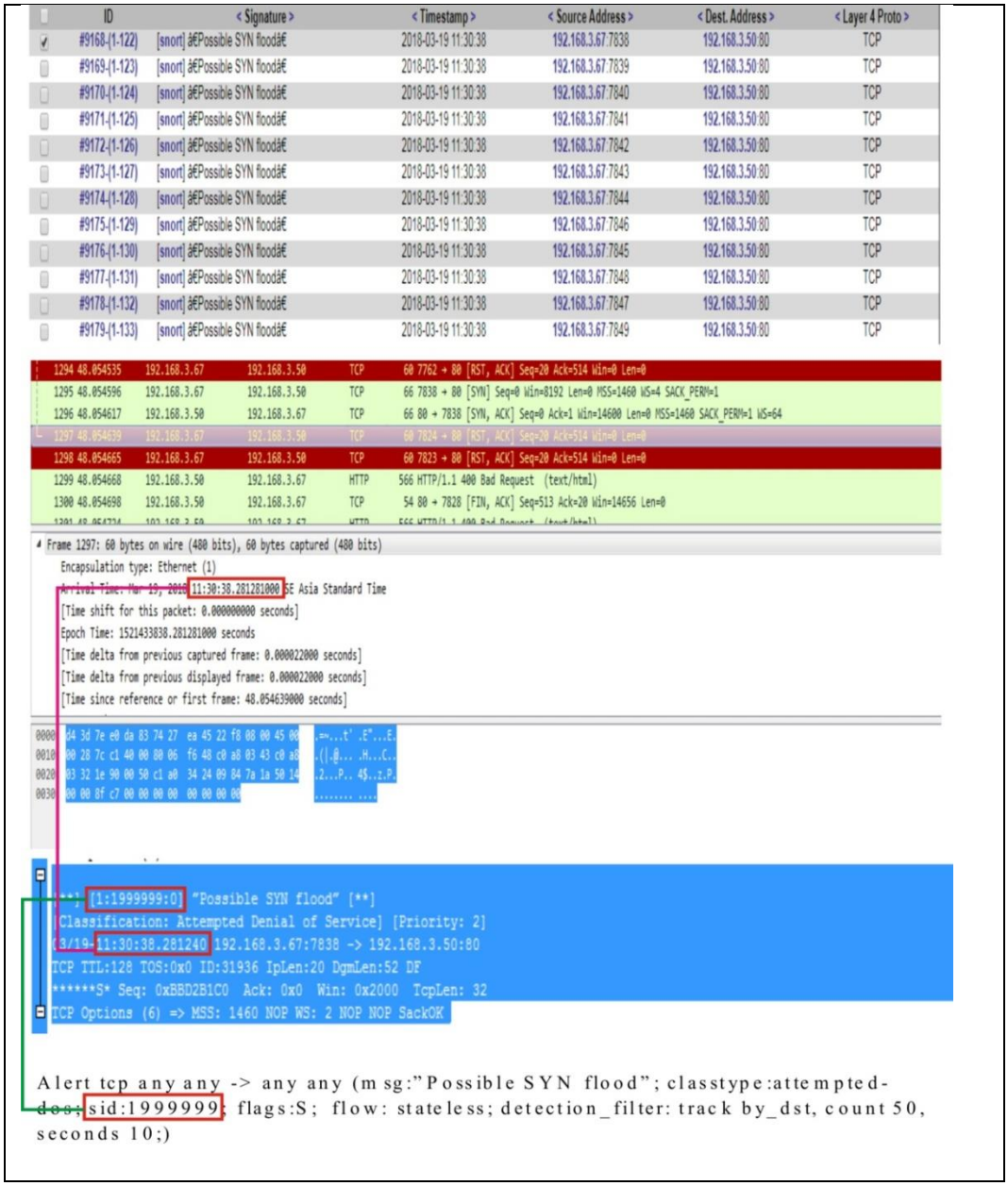
$$Accuracy = \frac{10}{20} x\ 100\ \% = 50\ \%$$

Tabel 4.5 Hasil Uji Rule 2

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Terdeteksi

Pada tabel 4.5 didapatkan hasil dari uji menggunakan rule 2, maka nilai akurasi yang didapatkan adalah sebagai berikut :

$$Accuracy = \frac{3}{4} x\ 100\ \% = 75\ \%$$



Gambar 4.9 Deteksi Rule 2

Gambar 4.9 menunjukkan *alert* yang dihasilkan oleh *snort*, dimana hasil yang ditampilkan snort dibandingkan dengan hasil yang diambil menggunakan *tcpdump* untuk melihat waktu yang dihasilkan apakah sama dengan waktu yang diberikan oleh IDS, dimana pada *rule* ini waktu yang dihasilkan *tcpdump* dan waktu yang dihasilkan acidbase menunjukkan waktu yang sedikit berbeda, akan tetapi *rule* ini tidak mampu mendeteksi untuk serangan dengan protokol UDP.

Tabel 4.6 Hasil Training Rule 3

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Terdeteksi
5.	LOIC (HTTP)	Terdeteksi
6.	LOIC (TCP)	Terdeteksi
7.	LOIC (UDP)	Tidak Terdeteksi
8.	HPING3	Terdeteksi
9.	LOIC (HTTP)	Terdeteksi
10.	LOIC (TCP)	Terdeteksi
11.	LOIC (UDP)	Tidak Terdeteksi
12.	HPING3	Terdeteksi
13.	LOIC (HTTP)	Terdeteksi
14.	LOIC (TCP)	Terdeteksi
15.	LOIC (UDP)	Tidak Terdeteksi
16.	HPING3	Terdeteksi
17.	LOIC (HTTP)	Terdeteksi
18.	LOIC (TCP)	Terdeteksi
19.	LOIC (UDP)	Tidak Terdeteksi
20.	HPING3	Terdeteksi

Pada tabel 4.6 didapatkan hasil dari training menggunakan rule 3, maka nilai akurasi yang didapatkan adalah sebagai berikut :

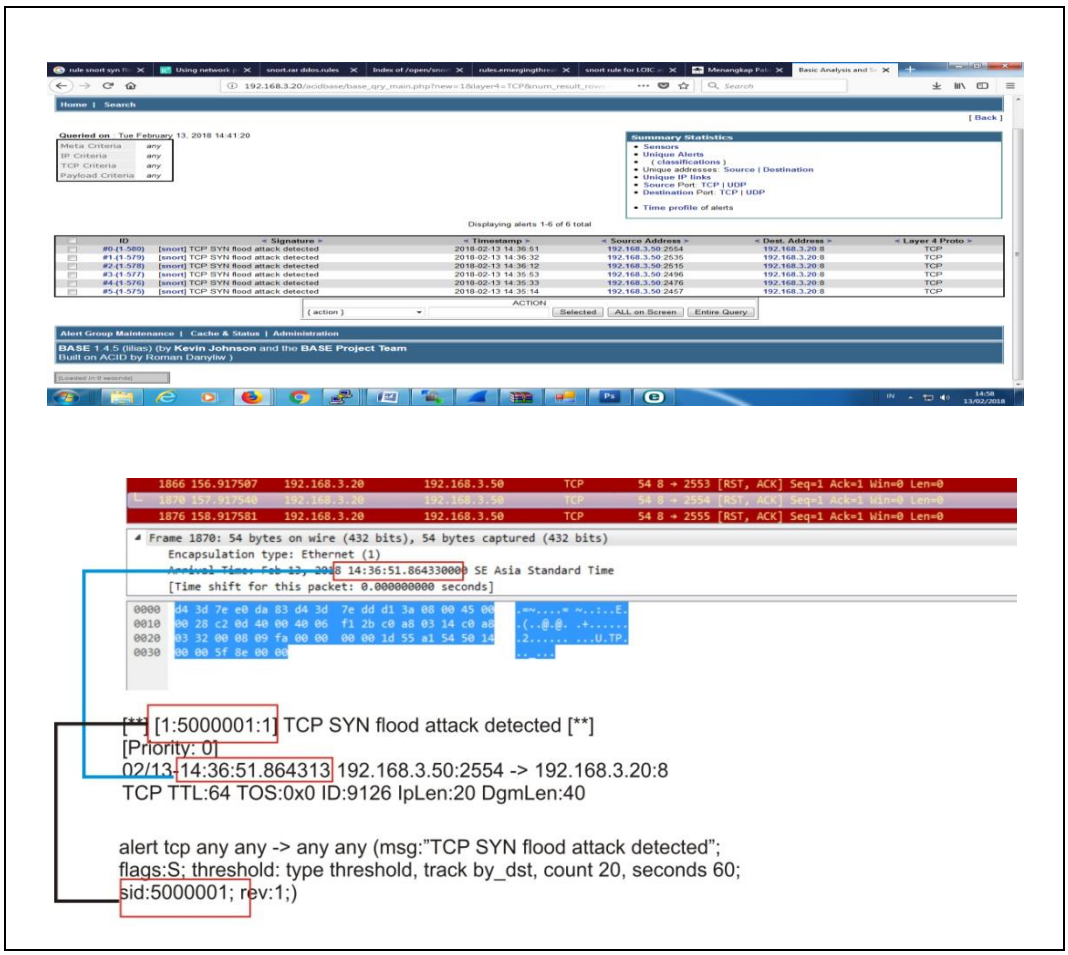
$$Accuracy = \frac{15}{20} x\ 100\ \% = 75\ \%$$

Tabel 4.7 Hasil Uji Rule 3

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Terdeteksi

Pada tabel 4.7 didapatkan hasil dari uji menggunakan rule 3, maka nilai akurasi yang didapatkan adalah sebagai berikut :

$$Accuracy = \frac{3}{4} x\ 100\ \% = 75\ \%$$



Gambar 4.10 Deteksi Rule 3

Gambar 4.10 menunjukkan *alert* yang dihasilkan oleh *snort*, dimana hasil yang ditampilkan snort dibandingkan dengan hasil yang diambil menggunakan *tcpdump* untuk melihat waktu yang dihasilkan apakah sama dengan waktu yang diberikan oleh IDS.

Tabel 4.8 Hasil Training Rule 4

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Tidak Terdeteksi
2.	LOIC (TCP)	Tidak Terdeteksi
3.	LOIC (UDP)	Terdeteksi
4.	HPING3	Tidak Terdeteksi
5.	LOIC (HTTP)	Tidak Terdeteksi
6.	LOIC (TCP)	Tidak Terdeteksi
7.	LOIC (UDP)	Terdeteksi
8.	HPING3	Tidak Terdeteksi
9.	LOIC (HTTP)	Tidak Terdeteksi
10.	LOIC (TCP)	Tidak Terdeteksi
11.	LOIC (UDP)	Terdeteksi
12.	HPING3	Tidak Terdeteksi
13.	LOIC (HTTP)	Tidak Terdeteksi
14.	LOIC (TCP)	Tidak Terdeteksi
15.	LOIC (UDP)	Terdeteksi
16.	HPING3	Tidak Terdeteksi
17.	LOIC (HTTP)	Tidak Terdeteksi
18.	LOIC (TCP)	Tidak Terdeteksi
19.	LOIC (UDP)	Terdeteksi
20.	HPING3	Tidak Terdeteksi

Pada tabel 4.8 didapatkan hasil dari training menggunakan rule 4, maka nilai akurasi yang didapatkan adalah sebagai berikut :

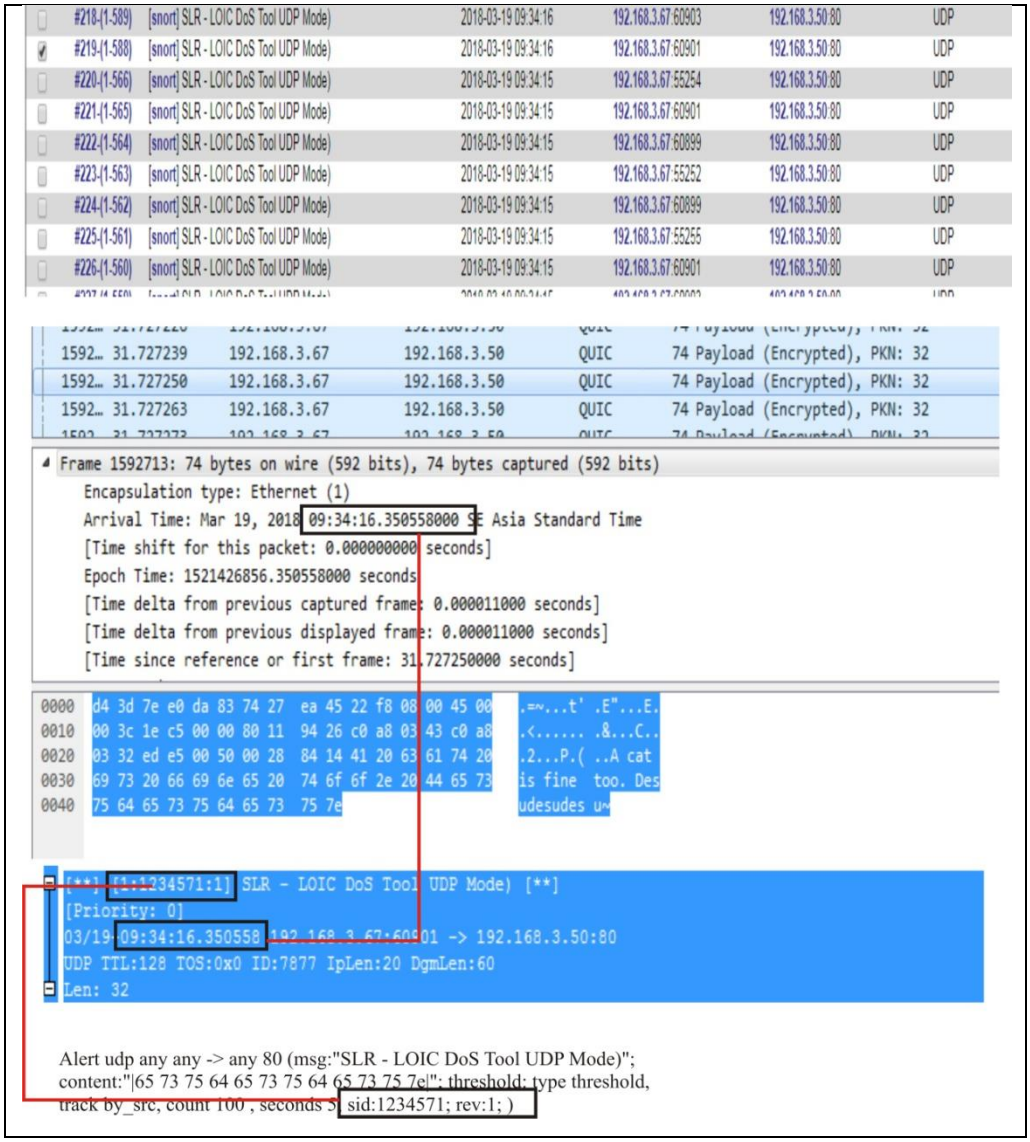
$$Accuracy = \frac{5}{20} x\ 100\ \% = 25\ \%$$

Tabel 4.9 Hasil Uji Rule 4

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Tidak Terdeteksi
2.	LOIC (TCP)	Tidak Terdeteksi
3.	LOIC (UDP)	Terdeteksi
4.	HPING3	Tidak Terdeteksi

Pada tabel 4.9 didapatkan hasil dari uji menggunakan rule 4, maka nilai akurasi yang didapatkan adalah sebagai berikut :

$$Accuracy = \frac{1}{4} x\ 100\ \% = 25\ \%$$



Gambar 4.11 Deteksi Rule 4

Gambar 4.11 menunjukkan *alert* yang dihasilkan oleh *snort*, dimana hasil yang ditampilkan snort dibandingkan dengan hasil yang diambil menggunakan *tcpdump* untuk melihat waktu yang dihasilkan apakah sama dengan waktu yang diberikan oleh IDS, dimana pada *rule* ini waktu yang dihasilkan *tcpdump* dan waktu yang dihasilkan acidbase menunjukkan waktu yang sama, akan tetapi *rule* ini tidak mampu mendeteksi untuk serangan dengan protokol TCP dan HTTP dan menggunakan *software* HPING.

Tabel 4.10 Hasil Training Rule 5

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Tidak Terdeteksi
5.	LOIC (HTTP)	Terdeteksi
6.	LOIC (TCP)	Terdeteksi
7.	LOIC (UDP)	Tidak Terdeteksi
8.	HPING3	Tidak Terdeteksi
9.	LOIC (HTTP)	Terdeteksi
10.	LOIC (TCP)	Terdeteksi
11.	LOIC (UDP)	Tidak Terdeteksi
12.	HPING3	Tidak Terdeteksi
13.	LOIC (HTTP)	Terdeteksi
14.	LOIC (TCP)	Terdeteksi
15.	LOIC (UDP)	Tidak Terdeteksi
16.	HPING3	Tidak Terdeteksi
17.	LOIC (HTTP)	Terdeteksi
18.	LOIC (TCP)	Terdeteksi
19.	LOIC (UDP)	Tidak Terdeteksi
20.	HPING3	Tidak Terdeteksi

Pada tabel 4.10 didapatkan hasil dari training menggunakan rule 5, maka nilai akurasi yang didapatkan adalah sebagai berikut :

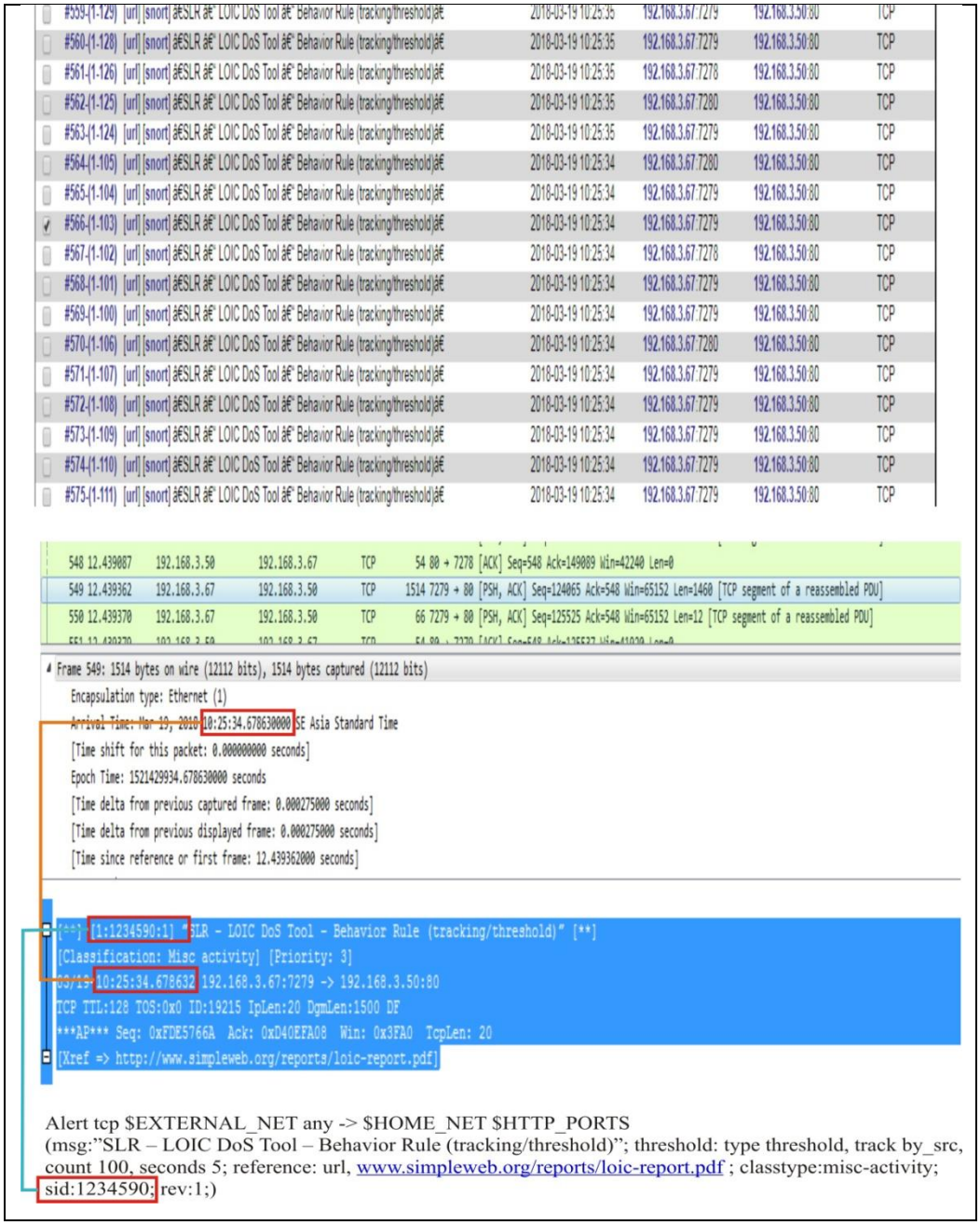
$$Accuracy = \frac{10}{20} x\ 100\ \% = 50\ \%$$

Tabel 4.11 Hasil Uji Rule 5

No.	Serangan	Keterangan
1.	LOIC (HTTP)	Terdeteksi
2.	LOIC (TCP)	Terdeteksi
3.	LOIC (UDP)	Tidak Terdeteksi
4.	HPING3	Tidak Terdeteksi

Pada tabel 4.11 didapatkan hasil dari uji menggunakan rule 5, maka nilai akurasi yang didapatkan adalah sebagai berikut :

$$Accuracy = \frac{2}{4} x\ 100\ \% = 50\ \%$$



Gambar 4.12 Deteksi Rule 5

Gambar 4.12 menunjukkan *alert* yang dihasilkan oleh *snort*, dimana hasil yang ditampilkan snort dibandingkan dengan hasil yang diambil menggunakan *tcpdump* untuk melihat waktu yang dihasilkan apakah sama dengan waktu yang diberikan oleh IDS, dimana pada *rule* ini waktu yang dihasilkan *tcpdump* dan waktu yang dihasilkan acidbase menunjukkan waktu yang sama, akan tetapi *rule*

ini tidak mampu mendeteksi untuk serangan dengan protokol UDP dan menggunakan *software* HPING.

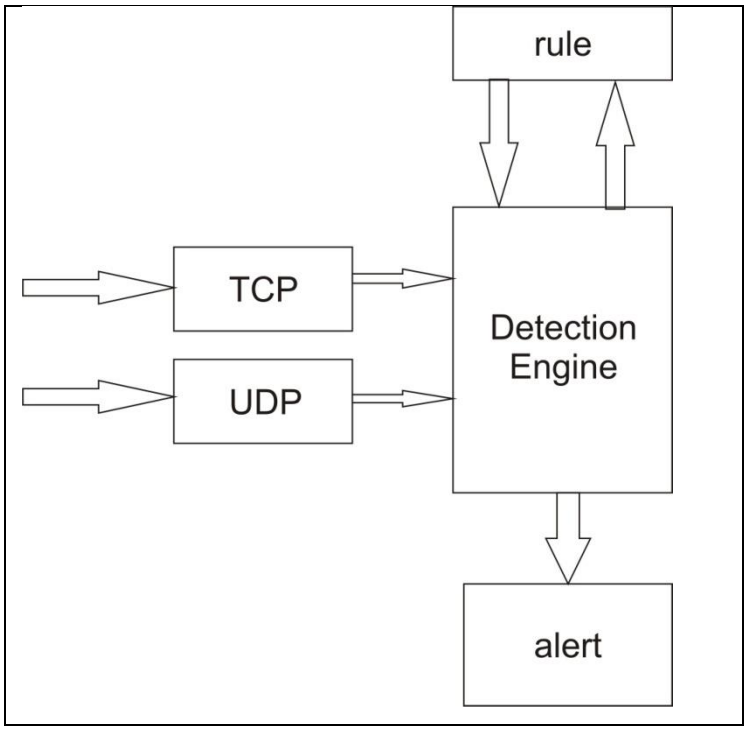
4.4.2 Pembahasan

Berdasarkan hasil yang didapat, kinerja dari integritas antara sistem operasi dan *snort* sangat baik, dengan kecepatan deteksi *threat* yang dapat dikatakan akurat, contohnya pada saat peneliti melakukan pengujian, IDS memberikan sebuah *alert message* secara bersamaan dengan pendeteksian. *Snort* IDS memilki sebuah *rules* yang dapat digunakan secara bebas, akan tetapi*rule* tersebut dapat dibuat sesuai dengan kebutuhan di lingkungan UIGM dengan mengedit files *local.rules.* untuk menerapkan *rule* yang akan digunakan di UIGM menggunakan kombinasi dari beberapa *rule* yang telah diujikan di atas, penggunaan *rule* 3 dan *rule* 4 direkomendasikan untuk diterapkan di UIGM.

Hasil pengujian ini dinilai baik dan akurat dilihat dari waktu serangan dengan data waktu yang ditunjukkan pada monitoring yang dihasilkan snort dan rule yang diterapkan mampu menangkap tindakan penyusupan dengan tepat, dan juga dari *false positif* yang dihasilkan.

Berdasarkan hasil pengujian didapatkan bahwa *rule* 3 dapat mendeteksi serangan dengan tingkat akurasi 75%, dengan pengenalan waktu keamanan dapat mencapai hampir 24 jam selama *server* aktif, sedangkan *rule* 4 dapat mendeteksi serangan dengan akurasi 25%, dengan waktu keamanan yang dapat mencapai hampir 24 jam selama *server* aktif. Mengacu pada hasil pengujian, maka direkomendasikan untuk menggabungkan *rule* 3 dan *rule* 4 untuk mendeteksi serangan yang dialamatkan pada jaringan yang terdapat di UIGM.

Penyusunan kombinasi *rule* dapat digambarkan pada model dibawah ini :



Gambar 4.13 model pendeteksian IDS

Berdasarkan hasil pemodelan diatas, deteksi serangan dapat digunakan untuk mendeteksi serangan dengan jenis TCP dan UDP pada UIGM.Model deteksi ini diimplementasikan pada gambar 4.14 yang digunakan pada jaringan yang ada di UIGM.

Otoritas untuk mengubah atau mengganti *rule* dilakukan oleh pihak administrator jaringan dengan tugas super admin atas persetujuan kepala Biro Pelaksana Teknis UIGM.

Tahapan otoritas :

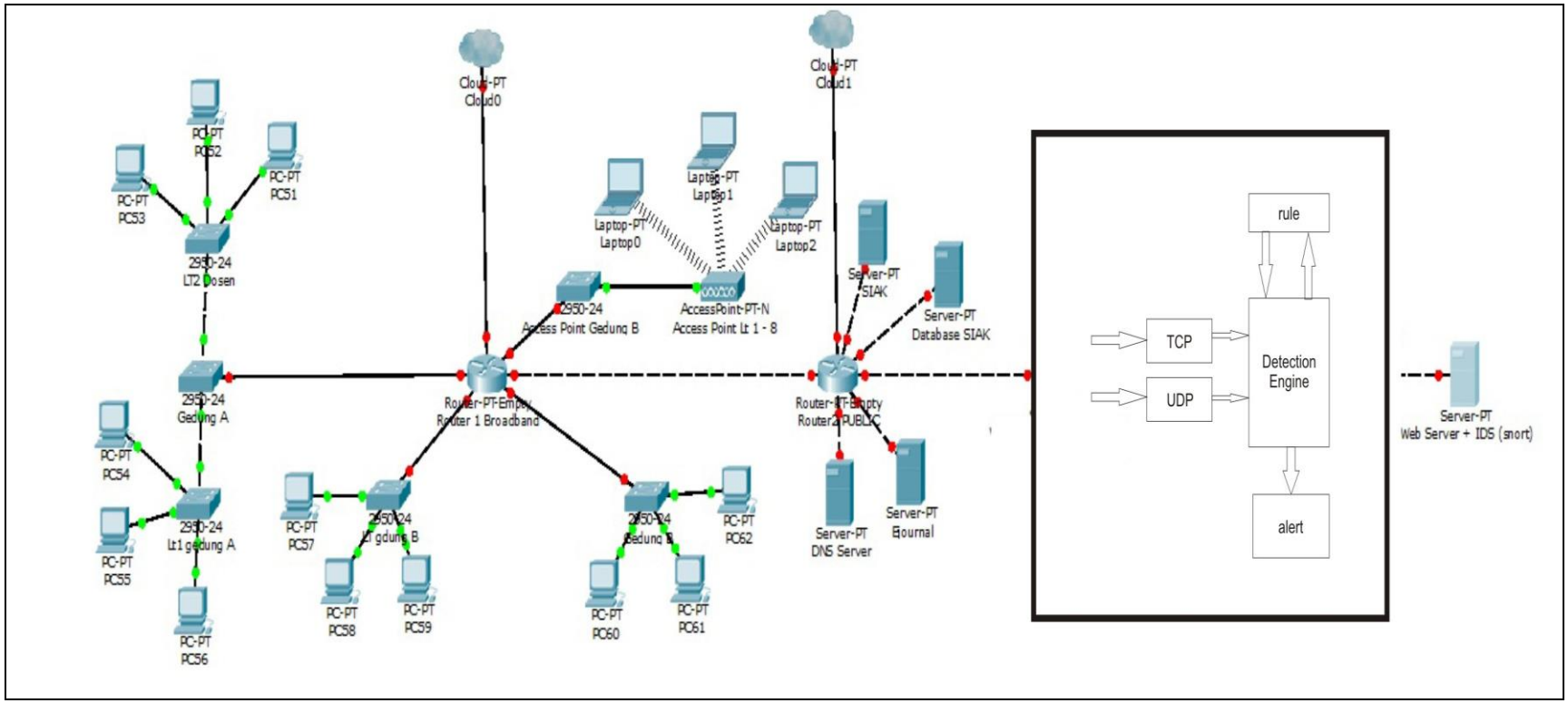
1. Administrator mendapatkan informasi bahwa terjadi serangan.
2. Administrator melakukan pencegahan jika serangan mampu untuk diatasi.
3. Administrator melaporkan kepada kepala Biro jika ingin melakukan perubahan atau penggantian *rule*.

4. Kepala Biro akan memberi info kepada Rektor dan Wakil Rektor untuk apabila ada perbaikan dalam jaringan.

Tahapan saat terjadi deteksi serangan :

1. Melakukan tindakan pengaman jika serangan masih mampu untuk dicegah.
2. Melakukan *reboot* sistem jika serangan membuat trafik pada jaringan penuh.
3. Melakukan *shutdown* sistem untuk sementara jika setelah *reboot* masih terdapat serangan.

Implementasi deteksi serangan DDoS dapat mengenali dan mendeteksi serangan yang terdapat di jaringan, baik serangan yang dilakukan dari dalam maupun serangan yang dilakukan dari luar lingkungan UIGM.



Gambar 4.14 Topologi yang diusulkan

BAB 5
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, peneliti menarik kesimpulan sebagai berikut :

1. Pengujian yang dilakukan dengan menggunakan 5 *rule* mampu mendeteksi semua serangan DDoS, dimana *rule* 1 mampu mendeteksi 60% untuk hasil *training* dan 50% untuk hasil uji, *rule* 2 mampu mendeteksi 50% untuk hasil *training* dan 75% untuk hasil uji, *rule* 3 mampu mendeteksi 75% untuk *training* dan hasil uji, *rule* 4 mampu mendeteksi 25% untuk hasil *training* dan hasil uji, dan *rule* 5 mampu mendeteksi 50% untuk hasil *training* dan hasil uji.
2. Metode *signatured based* yang diterapkan mampu mengenali pola data serangan dalam mendeteksi protokol TCP dan UDP.
3. *Monitoring* yang dibuat mampu mendeteksi semua serangan dengan tampilan *web base*.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, adapun saran yang ingin disampaikan adalah sebagai berikut :

1. Penelitian selanjutnya agar *rule* yang digunakan dapat diperluas lagi tidak hanya menggunakan *rule* yang mampu mendeteksi serangan DDoS.
2. Menggunakan metode *regular expression* atau metode yang lain dalam menggunakan *rule snort* sehingga dapat meningkatkan akurasi *rule* dalam mendeteksi serangan.
3. Menggabungkan penerapan IDS dan IPS agar ancaman yang terjadi dapat ditangani oleh sistem.

DAFTAR PUSTAKA

Putri, L. (2011) *Implementasi Intrusion Detection System (Ids) Menggunakan Snort Pada Jaringan Wireless (Studi Kasus : Smk Triguna Ciputat)*.

A.Masse, F., Nurul Hidayat, A. and Badrianto (2015) 'Penerapan Network Intrusion Detection System Menggunakan Snort Berbasis Database MySQL Pada Hotspot Kota', *Jurnal Elektronik Sistem Informasi Dan Komputer*, 1(2), pp. 1–16.

Affandi, M. and Setyowibowo, S. (no date) 'Implementasi Snort Sebagai Alat Pendeteksi Intrusi', *Jurnal Teknologi Informasi*, 4(2).

Alamsyah (2011) 'Implementasi keamanan instrusion detection system (ids) dan instrusion prevention system (ips) menggunakan clearos', *Jurnal SMARTek, Vol. 9 No. 3*, 9(3), pp. 223–229.

Budiman, S. A., Iswahyudi, C. and Sholeh, M. (2014) 'Implementasi Intrusion Detection System (Ids)', (November), pp. 1–8.

Ginta, P. W., Kusuma, G. P. and Negara, E. K. (2013) 'Implementasi Tools Network Mapper Pada Lokal Area Network (Lan)', *Jaringan Komputer*, 9(2), pp. 118–139.

Pratama, I Putu Agus Eka. (2015) *Handbook Jaringan Komputer*. 2nd edn. Informatika Bandung.

Knowledge, R. (2010) *Trik Memonitor Jaringan*. Jakarta: Elex Media Komputindo.

Mada, G. (2013) 'Evaluasi Keamanan Akses Jaringan Komputer Nirkabel (Kasus : Kantor Pusat Fakultas Teknik Universitas', *Evaluasi Keamanan Akses Jaringan Komputer Nirkabel(Kasus:Kantor Pusat Fakultas Teknik Universita Gadjah Mada)*, 1(1), p. 5. doi: 10.22146/JNTETI.V1I1.3.

Manual Snort (no date). Available at: www.snort.org.

Mentang, R., Sinsuw, A. A. E. and Najoan, X. B. N. (2015) 'Perancangan Dan Analisis Keamanan Jaringan Nirkabel Menggunakan Wireless Intrusion Detection System', *E-Jourrnal Teknik Elektro dan KomputerMentang, R., Sinsuw, A. A. E., Najoan, X. B. N., & Elektro-ft, J. T. (2015). Perancangan*

Dan Analisis Keamanan Jaringan Nirkabel Menggunakan Wireless Intrusion Detection System. E-Jourrnal Teknik Elektro Dan Komputer, 5(7), pp. 35–44.

Mustofa, M. M. and Aribowo, E. (2013) 'Penerapan Sistem Keamanan Honeypot Dan Ids Pada', *Sistem Keamanan Honeypot dan IDS*, 1(1), pp. 111–118.

Panduan Skripsi (2017). Palembang.

Pratomo, B. A. and Ijtihadie, R. M. (2016) 'Sistem Deteksi Intrusi Menggunakan N-Gram Dan Cosine Similarity', *JUTI: Jurnal Ilmiah Teknologi Informasi*, 14(1), p. 108. doi: 10.12962/j24068535.v14i1.a516.

Rafiudin, R. (2010) *Mengganyang Hacker Dengan Snort*. Yogyakarta: Andi.

Raharjo, S. and Informatika, P. T. (2015) 'Jurnal JARKOM Vol . 2 No . 2 Juni2015 ISSN : 2338-6313', *Implementasi Konsep Multi-Nas Dengan Mengintegrasikan Vpn Server Dan Freeradius Server Dalam Membangun Sistem Otentikasi Jaringan Wifi*, 2(2), pp. 16–27. doi: Muh. Ibnu Habil Hanafi, Suwanto Raharjo, Suraya.

Serangan_DoS (no date). Available at: https://id.wikipedia.org/wiki/Serangan_DoS (Accessed: 11 December 2017).

Sofana, I. (2015) *Membangun Jaringan Komputer*. Bandung: Informatika.

Tzeyoung Max Wu (2009) *Information Assurance Tools Report - Intrusion Detection System Sixth Edition*. Herndon, United State: IATAC.

Wagner, D. (2007) 'Intrusion Detection System'. Bandung, pp. 1–6.

Wicaksono, A. P. *et al.* (2014) 'Sistem Deteksi Intrusi dengan Snort (Intrusion Detection System with Snort)', III, pp. 31–34.