UNIVERSITAS INDONESIA

RANCANG BANGUN DAN IMPLEMENTASI SISTEM MONITORING JARINGAN BERBASIS WEB UNTUK MENENTUKAN TINGKAT RESIKO ANCAMAN KEAMANAN SECARA DINAMIS

SKRIPSI

OLEH

REZA HADI SAPUTRA

04 05 03 0672

DEPARTEMEN TEKNIK ELEKTRO

FAKULTAS TEKNIK UNIVERSITAS INDONESIA

JUNI 2009

UNIVERSITAS INDONESIA

RANCANG BANGUN DAN IMPLEMENTASI SISTEM MONITORING JARINGAN BERBASIS WEB UNTUK MENENTUKAN TINGKAT RESIKO ANCAMAN KEAMANAN SECARA DINAMIS

SKRIPSI

DIAJUKAN UNTUK MELENGKAPI SEBAGIAN PERSYARATAN MENJADI SARJANA TEKNIK

OLEH

REZA HADI SAPUTRA

04 05 03 0672

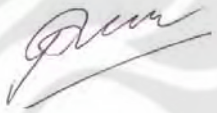
DEPARTEMEN TEKNIK ELEKTRO

FAKULTAS TEKNIK UNIVERSITAS INDONESIA

JUNI 2009

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama	**: REZA HADI SAPUTRA**
NPM	**: 0405030672**
Tanda Tangan	**:** 
Tanggal	**: Depok, 16 Juni 2009**

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama	: Reza Hadi Saputra
NPM	: 0405030672
Program Studi	: Teknik Elektro
Judul Skripsi	: Rancang Bangun Dan Implementasi Sistem Monitoring Jaringan Berbasis Web Untuk Menentukan Tingkat Resiko Ancaman Keamanan Secara Dinamis

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Elektro, Fakultas Teknik Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Muhammad Salman S.T, MIT

Penguji : Dr. Ir. Anak Agung Putri Ratna M.Eng

Penguji : Ir. Endang Sriningsih MT,Si

Ditetapkan di : Depok

Tanggal : 29 Juni 2009

UCAPAN TERIMA KASIH

Puji syukur kepada Allah SWT atas segala rahmat dan hidayah-Nya sehingga skripsi ini dapat diselesaikan. Shalawat dan salam semoga senantiasa tercurahkan kepada Nabi Muhammad saw. Ucapan terima kasih ditujukan kepada:

Muhammad Salman ST., MIT

selaku pembimbing skripsi yang telah meluangkan waktunya untuk memberikan bimbingan, saran, pengarahan dan kemudahan lainnya sehingga skripsi ini dapat diselesaikan dengan baik.

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Reza Hadi Saputra
NPM : 0405030672
Program Studi : Teknik Elektro
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

RANCANG BANGUN DAN IMPLEMENTASI SISTEM MONITORING JARINGAN BERBASIS WEB UNTUK MENENTUKAN TINGKAT RESIKO ANCAMAN KEAMANAN SECARA DINAMIS

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, danmemublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 16 Juni 2009

Yang menyatakan

(Reza Hadi Saputra)

vi

ABSTRAK

Nama : Reza Hadi Saputra

Program Studi : Teknik Elektro

Judul : RANCANG BANGUN DAN IMPLEMENTASI SISTEM MONITORING JARINGAN BERBASIS WEB UNTUK MENENTUKAN TINGKAT RESIKO ANCAMAN KEAMANAN SECARA DINAMIS

IT Risk Management merupakan suatu metodologi yang digunakan suatu perusahaan/ organisasi untuk dapat membantu mengatur resiko dari semua divais dan infrastruktur IT yang dimilikinya. Dengan IT Risk Management yang baik, maka perusahaan/ organisasi dapat mengatur seluruh aset IT yang dimiliki sehingga dapat membantu meningkatkan produktifitas perusahaan/ organisasi tersebut. IT Risk Management terdiri atas tiga tahapan, yaitu *risk assessment, risk mitigation* serta *evaluation* dan *assessment*. Pada setiap tahapan tersebut akan diperoleh *output* tertentu yang berupa *report* mengenai perusahaan/ organisasi. Untuk membantu dalam implementasi IT Risk Management, dibutuhkan Intrusion Detection System (IDS) yang akan memberikan *report* mengenai kondisi jaringan suatu perusahaan/ organisasi, meliputi pelaporan apabila terjadi gangguan serta tindakan yang akan dilakukan terhadap gangguan tersebut.

Pada skripsi ini dibuat suatu perancangan aplikasi berbasis web yang digunakan untuk perhitungan *risk level* (tingkat resiko) dalam suatu LAN pada tahapan *risk assessment*. Aplikasi tersebut digunakan untuk menghitung nilai *risk level* untuk setiap ancaman (threat) yang terdeteksi oleh IDS untuk suatu pilihan waktu yang dimasukkan oleh *user*. Aspek keamanan jaringan untuk suatu LAN merupakan hal yang sangat penting, terutama apabila di dalam LAN tersebut terdapat komputer yang didalamnya terdapat data yang sangat penting dan pada jaringan yang sama dengan komputer tersebut, terdapat komputer-komputer lain yang dipakai oleh banyak orang. Ancaman terhadap data pada komputer tersebut tidak hanya dapat berasal dari internet, tetapi juga dapat berasal dari komputer-komputer dalam LAN. Oleh karena itu, dengan adanya aplikasi ini diharapkan apabila muncul suatu serangan terhadap suatu komputer yang berasal dari komputer lain pada LAN yang sama, serangan tersebut dapat terdeteksi sehingga tindakan perlindungan data dapat dilakukan.

Pada bagian akhir dari skripsi ini, sistem tersebut diujicoba pada LAN suatu perusahaan, untuk selanjutnya dilakukan suatu ujicoba serangan. Ada tiga tahapan ujicoba dengan setiap tahapan dilihat nilai Risk Level yang dihasilkan sistem. Pada tahap pertama, yaitu pencarian *IP Address* pada suatu LAN, menghasilkan nilai kuantitatif Risk Level sebesar 4 (Low Risk Level). Pada skenario ujicoba tahap 2, yaitu pencarian informasi meliputi port dan nama komputer untuk suatu komputer, menghasilkan nilai kuantitatif Risk Level sebesar 232 (High Risk Level). Pada skenario ujicoba tahap 3, yaitu pengambilalihan suatu komputer target, menghasilkan nilai kuantitatif Risk Level sebesar 232 (High Risk Level).

Kata Kunci : IT Risk Management, Intrusion Detection System, Risk Level

ABSTRACT

Nama : Reza Hadi Saputra

Program Studi : Teknik Elektro

Judul : DESIGN AND IMPLEMENTATION OF WEB BASED NETWORK MONITORING SYSTEM FOR DYNAMIC RISK LEVEL CALCULATION

IT Risk Management is a methodology used by a company / organization that can help them to manage risk from all devices and IT infrastructure assets. With the good IT Risk Management, the company / organization can manage all IT assets owned so can help them to increase the productivity of the company / organization. IT Risk Management consists of three phases, namely risk assessment, risk mitigation and the evaluation and assessment. At each stage, there are an output in the form of a report to the company / organization. To assist in the implementation of IT Risk Management, Intrusion Detection System (IDS) is required, to provide a report on the condition of the network of a company / organization, including reporting of when an interruption occurs and the action will be taken.

In this thesis, a web-based application is designed, that is used to calculate the risk level in a LAN on the risk assessment stage. That application is used to calculate the value of the risk level for each threat detected by the IDS for a selection entered by the user. Aspects of network security for a LAN is very important, especially where in the LAN there are computers that contains a very important data and at the same with computers, there are computers that are used by many people. Threats to the data on the computers not only can come from the internet, but can also come from computers in the LAN. Therefore, this application is expected to appear when an attack against a computer that came from another computer on the same LAN, the attack can be detected so that the data protection act can be done.

At the end of this thesis, the system is tested on a corporate LAN, to be a trial of attacks. There are three stages of testing with each of the stages seen the value of the resulting Risk Level system. In the first stage, the IP Address is searched on a LAN, the quantitative value of Risk Level is 4 (Low Risk Level). In the phase 2 trial scenario, the search information includes the port and the name of the computer to a computer, the quantitative value of Risk Level is 232 (High Risk Level). In the phase 3 trial scenario, the takeovers process of a target computer, the quantitative value of Risk Level is 232(High Risk Level).

Keywords : IT Risk Management, Intrusion Detection System, Website

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR SINGKATAN

- ARP (Address Resolution Protocol)
- ICMP (Internet Control Message Protocol)
- IP (Internet Protocol)
- IT (Information Technology)
- ITRM (Information Technology Risk Management)
- LAN (Local Area Network)
- SDLC (Software Development Life Cycle)
- UML (Unified Modeling Language)
- VOIP (Voice Over Internet Protocol)

BAB 1

PENDAHULUAN

1.1 LATAR BELAKANG

Information Technology (IT)/ Teknologi Informasi kini telah menjadi faktor penting dalam kehidupan manusia. Penggunaan Komputer, internet dan telepon genggam kini telah menjadi bagian yang tidak dapat dipisahkan dari kehidupan manusia. Perkembangan IT telah banyak mengubah pola hidup masyarakat dalam banyak hal, mulai dari segi sosial, interaksi antar manusia, pekerjaan, pendidikan, hiburan dan lain-lain. Berkat perkembangan teknologi, interaksi antar manusia telah banyak berubah, yang sebelumnya hanya berupa *face-to-face* ataupun melalui surat, sekarang komunikasi antara mereka juga dapat berupa komunikasi yang hanya berupa suara (telepon**/VOIP**), maupun tulisan saja (**SMS/***Internet Messenger*) bahkan sekarang komunikasi juga berupa tatap wajah jarak jauh (*Video Call/Video Conference*). Dalam hal pendidikan sumber informasi pelajaran dapat dengan mudah diperoleh(*e-learning*), bahkan dengan adanya internet, kelas interaktif antar siswa yang berjauhan pun dapat dimungkinkan sehingga dapat membantu menghidupkan kesempatan belajar untuk semua orang dimanapun dan kapanpun juga. Pada awalnya data-data yang disimpan dalam media dijital suatu perusahaan, digunakan untuk pencatatan secara internal suatu perusahaan dan juga sebagai pengaturan informasi finansial, informasi pelanggan dan sistem pembayaran karyawan.

Namun dengan adanya pengaruh dari perkembangan IT, sistem yang ada di suatu perusahaan pun juga banyak berubah. Dengan diimplementasikannya *private network* pada suatu perusahaan agar komunikasi antara pemilik dan karyawan dapat lebih mudah dilakukan. Pemilik tidak perlu lagi kesulitan mencetak suatu dokumen untuk kemudian didistribusikan kepada karyawannya, dengan adannya intranet dokumen tersebut cukup dikirim ke masing-masing email karyawannya. Perusahaan juga dapat membangun *extranet* atau *extended network* sebagai cara agar *supplier*, *vendor* atau *customer* dapat mengakses

corporate data untuk memeriksa status *order*, *inventory* atau daftar barang, namun dengan akses yang lebih terbatas.

Namun, apabila tidak didukung dengan manajemen yang baik, IT pada suatu perusahaan dapat menjadi bumerang. Bukan membantu meningkatkan kinerja perusahaan namun malah dapat merusak keseluruhan sistem yang ada. Pengaturan jaringan yang buruk dapat menimbulkan masalah apabila tidak cepat disadari, seperti peletakan server yang salah, pengkabelan yang tidak diatur dengan baik ataupun *IP Address Assignment* yang salah. Selain itu pengaturan jaringan yang buruk dapat membuat penyebaran virus, *worm* atau *bug* bertebaran di jaringan internal perusahaan sehingga bukan saja menggangu namun dapat juga merusak dokumen penting perusahaan. Bahkan asumsi yang sangat buruk adalah adanya pihak asing/kompetitor yang dapat dengan mudah mengakses jaringan internal untuk dapat mengambil maupun merusak data-data penting dan rahasia perusahaan.

Oleh karena itu, dibutuhkan suatu pengaturan resiko yang baik dari semua divais maupun infrastruktur IT yang ada, yang disebut dengan IT Risk Management. Dengan IT Risk Management yang baik yang dimiliki oleh suatu perusahaan, maka perusahaan dapat mengatur dengan baik seluruh aset IT yang dimilikinya untuk dapat membantu meningkatkan produktifitas perusahaan.

Salah satu contoh pengaturan yang dapat dilakukan pada tahapan IT Risk Management adalah dengan membuat suatu aplikasi perhitungan IT Risk Management. Aplikasi tersebut akan menghitung nilai IT Risk Level dan mendeteksi ancaman-ancaman apa saja yang menyerang suatu host/komputer pada jaringan lokal suatu perusahaan/organisasi. Ancaman terhadap data-data tidak hanya dapat berasal dari internet atau ancaman fisik saja. Salah satu bentuk ancaman yang dapat luput dari perhatian adalah melindungi data dari ancaman pihak internal. Dengan dibuatnya suatu aplikasi yang dapat menghitung IT Risk Level serta mendeteksi apabila muncul ancaman, maka pihak pimpinan perusahaan/organisasi dapat melakukan evaluasi apakah kontrol yang dimiliki telah sanggup bertahan untuk melindungi data perusahaan/organisasi atau diperlukan kontrol baru untuk melindungi data-data tersebut.

1.2 TUJUAN

Tujuan penulisan skripsi ini adalah merancang sistem aplikasi berbasis web yang digunakan untuk menghitung nilai Risk Level untuk suatu LAN. Pelaporan mengenai nilai risk level dapat dijadikan dasar pertimbangan oleh perusahaan apakah kontrol yang dimiliki perusahaan layak untuk dipertahankan atau harus diganti dengan kontrol yang lebih baik.

1.3 PEMBATASAN MASALAH

Pada skripsi ini akan dibahas mengenai konsep dari IT Risk Management, meliputi pentingnya suatu perusahaan untuk melakukan implementasi. Selanjutnya juga akan dibahas mengenai Intrusion Detection System, yaitu suatu *tools* yang digunakan untuk melakukan pengawasan terhadap keamanan suatu jaringan. Pada bagian selanjutnya akan dilakukan perancangan aplikasi yang menghitung nilai *risk level* dari respon-respon yang dihasilkan oleh Intrusion Detection System. Pada bagian terakhir, aplikasi yang telah selesai dibuat akan diuji coba pada suatu contoh jaringan lokal perusahaan X untuk kemudian dilakukan suatu ujicoba serangan. Dengan adanya ujicoba serangan dapat dilihat seberapa baik respon-respon yang dihasilkan aplikasi yang dibuat terhadap serangan-serangan yang dibuat.

1.4 METODE PENULISAN

Metode yang digunakan dalam penelitian ini adalah**:**

a) Studi literatur dengan mempelajari informasi dari berbagai sumber literatur, seperti: buku, jurnal, dan artikel-artikel yang berkaitan dengan sistem yang akan dibuat.

b) Desain dan Implementasi aplikasi berbasis web untuk perhitungan IT Risk Management.

c) Pendekatan diskusi dengan pembimbing skripsi, dosen, serta teman mengenai skripsi yang dibuat

d) Analisa skenario dan desain.

1.5 SISTEMATIKA PENULISAN

Sistematika penulisan pada skripsi ini ialah sebagai berikut :

BAB 1 Pendahuluan

Terdiri dari latar belakang masalah, tujuan skripsi, batasan masalah dan sistematika penulisan.

BAB 2 Landasan Teori

Membahas mengenai IT Risk Management, Security Monitoring Tools dan UML Diagram

BAB 3 Perancangan

Membahas mengenai perancangan sistem berdasarkan sistematika *Software Development Life Cycle*

BAB 4 Implementasi dan Uji Coba Sistem Pada Jaringan

Membahas mengenai ujicoba dari rancangan yang telah dibuat pada bab sebelumnya pada sebuah LAN perusahaan X

BAB 5 Kesimpulan

Merupakan penutup pembahasan pada penulisan skripsi ini.

BAB 2
TAHAPAN-TAHAPAN IT RISK MANAGEMENT, SECURITY MONITORING TOOLS DAN UML DIAGRAM

2.1 IT Risk Management

Tahap-tahap dalam implementasi IT Risk Management meliputi tiga tahapan[1] :

- Risk Assessment
- Risk Mitigation
- Evaluation and Assessment

Ketiga tahapan tersebut akan dibahas lebih mendalam pada bagian selanjutnya dari tulisan ini. Bagian 2.1.2 akan menjelaskan mengenai proses lebih lanjut yang terjadi mengenai Risk Asessment, yang meliputi identifikasi dan evaluasi resiko serta dampak yang akan ditimbulkan dari resiko-resiko tersebut. Bagian 2.1.3 akan meliputi penjelasan mengenai risk mitigation, yaitu penjelasan mengenai prioritas-prioritas, tahap implementasi dan tahap bagaimana mengurangi resiko-resiko yang ditimbulkan yang diperoleh dari tahapan risk assessment. Sedangkan bagian 2.1.4 akan meliputi pembahasan mengenai proses evaluasi berkelanjutan dan hal-hal penting apa sajakah yang harus diperhatikan agar IT Risk Management dapat berhasil diimplementasikan pada suatu perusahaan.

IT Risk Management merupakan suatu proses yang mengharuskan seorang IT Manager menyeimbangkan operational dan biaya ekonomi suatu *protective measures* untuk mencapai tujuan perusahaan dengan melindungi *IT Systems and Data* yang mendukung tercapainya misi-misi tersebut. Proses ini tidak unik untuk setiap *IT Environment*, bahkan meliputi proses pengambilan keputusan dalam semua aspek kehidupan keseharian manusia. Contohnya untuk *home security*, seseorang memutuskan untuk memiliki *home security systems* di rumahnya dan membayar biaya bulanan kepada *service provider* yang memiliki sistem tersebut untuk perlindungan yang lebih baik untuk aset yang dimiliki. Kemungkinannya, pemilik rumah akan menghitung biaya yang harus dikeluarkan ketika memutuskan membeli *home security system* dari *service provider* terhadap nilai

dari aset-aset yang harus dilindungi dan keamanan keluarga. Oleh karena itu, dibutuhkan perhitungan dan perencanaan yang tepat.

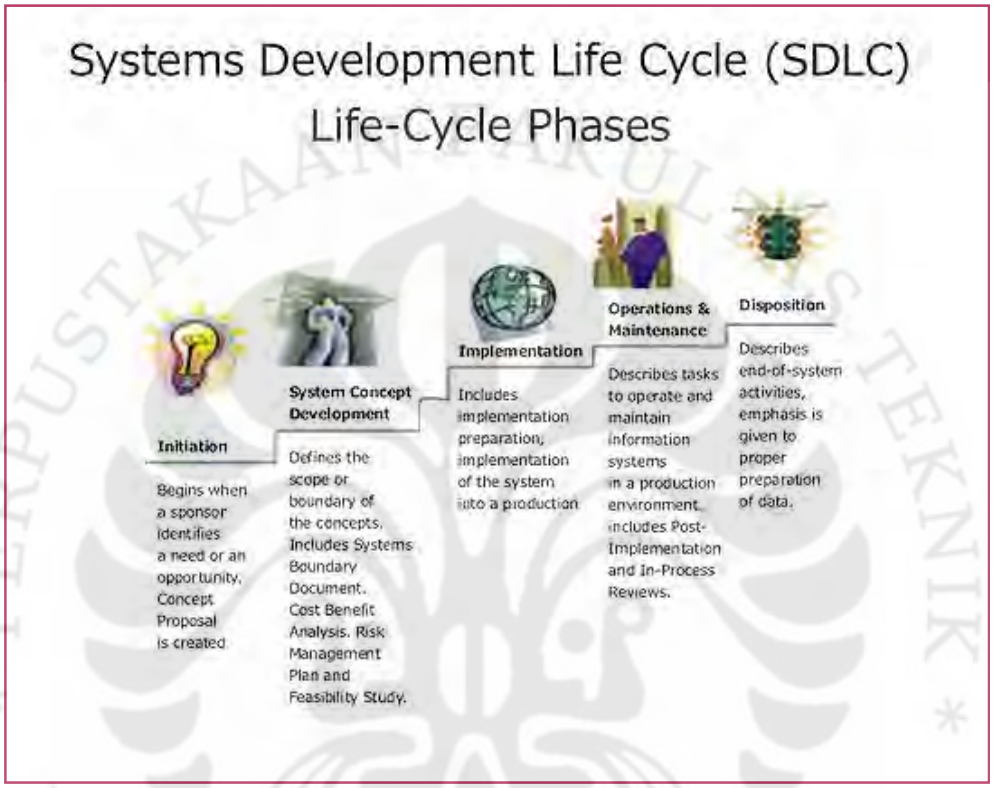
Seorang pimpinan perusahaan/organisasi harus memastikan perusahaan/organisasi yang dipimpinnya memiliki kemampuan yang dibutuhkan untuk mencapai misi tersebut. Pemilik harus menentukan level *security* manakah yang akan diimplementasikan yang sesuai dengan permasalahan yang dihadapi oleh perusahaan tersebut untuk dapat berhasil dalam dunia bisnisnya dan sesuai dengan keuangan yang dimiliki oleh perusahaan tersebut. Biasanya, biaya yang dianggarkan untuk *IT Security* pada suatu perusahaan sangat terbatas, untuk itu metodologi IT Risk Management haruslah terstruktur dengan baik sehingga dapat membantu *management* dan pemilik mencapai tujuan perusahaan yang diharapkan.

2.1.1 Integrasi IT Risk Management Dalam Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) merupakan suatu bentuk rekayasa sistem dan *sofware* yang berhubungan dengan proses pengembangan sistem, model dan metodologi, yang digunakan sesorang untuk mengembangkan suatu sistem, biasanya komputer/ *IT system.* SDLC merupakan proses berkelanjutan yang digunakan oleh sistem analis suatu perusahaan untuk mengembangkan sistem informasi, meliputi tahap-tahap *initiation, development/ acquisition, implementation, operation/ maintenance* dan *disposal*. Sebuah perancangan SDLC yang baik dapat menghasilkan suatu sistem berkualitas tinggi sehingga dapat memenuhi keinginan *customer* atau juga dapat membantu orang-orang/karyawan dalam suatu sistem perusahaan bekerja secara efektif dan efisien dalam waktu tertentu dan menghasilkan yang diharapkan.

Meminimalkan dampak buruk pada perusahaan serta kebutuhan dalam pengambilan keputusan adalah alasan yang penting mengapa suatu perusahaan harus mengimplementasikan *IT Risk Management Process* untuk *IT Systems* yang dimiki. IT Risk Management yang efektif haruslah secara utuh terintegrasi dalam SDLC. IT System SDLC memiliki 5 fase: Initiation, Development/ Acquisition, Implementation, Operation/ Maintenance dan Disposal. Pengimplementasian IT

Risk Management(ITRM) dalam suatu perusahaan dapat dilakukan pada setiap fase SDLC. Tabel 2.1 menjelaskan karakteristik untuk setiap fase SDLC dan juga menjelaskan bagaimana IT Risk Management dapat diintegrasikan untuk setiap fase tersebut.



Gambar 2.1 Tahap-Tahap dalam SDLC[1]

Tabel 2.1 Integrasi IT Risk Management pada SDLC[1]

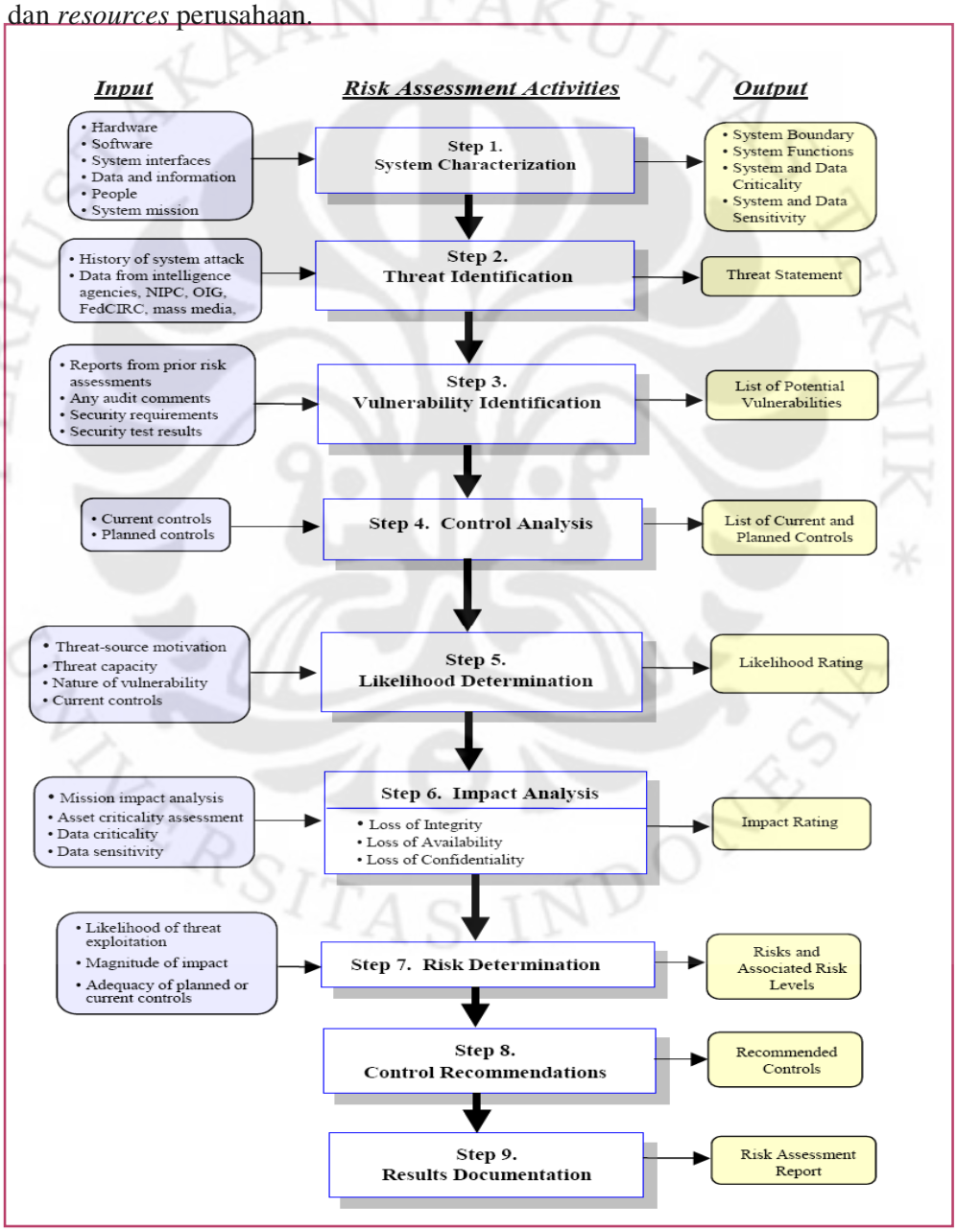
Fase-Fase SDLC	Deskripsi Fase	Aktivitas ITRM
Fase 1: Initiation	Hal-hal apa saja yang diinginkan untuk suatu *IT System* dan tujuan serta cakupan *IT System* kemudian didokumentasikan.	Pengidentifikasian resiko dilakukan untuk mendukung *system development*, meliputi *security requirement* dan *security concept.*

Fase-Fase SDLC	Deskripsi Fase	Aktivitas ITRM
Fase 2: Development/ Acquisition	IT System didisain, dibeli, diprogram dan dikembangkan	Resiko-resiko diidentifikaskan pada fase ini,dilakukan untuk mendukung *security analyses* dari *IT System* yang bertugas mendesain arsitekturnya
Fase 3: Implementation	Fitur-fitur keamanan dari suatu sistem harus dikonfigurasi, dijalankan, dites dan diverifikasi	ITRM mendukung assessment dari system implementation terhadap requirement yang dibutuhkan. Keputusan mengenai resiko-resiko harus dibuat terhadap sistem yang dibuat
Fase 4: Operation/ Maintenance	Sistem yang telah diimplementasikan kemudian dijalankan meliputi semua fitur dan fungsi yang dimiliki. Biasanya sistem tersebut dapat dimodifikasi meliputi penambahan software/hardware maupun juga dapat terjadi perubahan prosedur, aturan dan proses sistem	Kegiatan ITRM adalah menjalankan keamanan sistem untuk waktu-waktu yang periodik atau kapanpun perubahan-perubahan besar dibuat pada *IT system*.
Fase 5: Disposal	Fase ini meliputi pemindahan, pengarsipan, pembuangan maupun penghancuran informasi dan pembersihan software/hardware	ITRM memastikan hardware / software dibuang secara benar, residual data ditangani dengan benar dan migrasi ke sistem baru dilakukan secara benar, aman dan sistematis.

2.1.2 Risk Assessment

Risk Assessment merupakan tahap pertama dalam metodologi ITRM. Perusahaan menggunakan risk assessment untuk menentukan cakupan dari bahaya apa saja yang berpotensi dan resiko-resiko yang akan ditimbulkan terhadap *IT System*. Output yang diharapkan dari proses ini adalah dapat membantu mengidentifikasikan kontrol yang sesuai untuk mengurangi atau menghilangkan resiko-resiko tersebut selama proses risk mitigation.

Resiko adalah fungsi dari kemungkinan *vulnerability* yang dikerjakan oleh suatu sumber ancaman dan hasil dari dampak yang ditimbulkannya terhadap suatu perusahaan. Untuk menentukan kemungkinan dari kerugian yang akan ditimbulkan di masa depan, *threat*(ancaman) tersebut harus dianalisis dan dibandingkan dengan kemungkinan *vulnerability* yang ditimbulkan dan kontrol yang harus diberikan terhadap *IT System.* Level dari ancaman tersebut kemudian ditentukan dari pengaruhnya terhadap dampak yang ditimbulkan terhadap *IT asset* dan *resources* perusahaan.



Gambar 2.2 Flowchart Metodologi Risk Assessment[1]

Metodologi risk assessment terdiri atas 9 langkah utama, yaitu : 1). System Characterization, 2). Threat Identification, 3).Vulnerability Identification, 4). Control Analysis, 5). Likelihood Determination, 6). Impact Analysis, 7). Risk Determination, 8). Control Recommendations, 9). Result Documentation

2.1.2.1 System Characterization

Dalam menetukan suatu resiko untuk *IT system*, langkah pertama adalah menentukan cakupan dari usaha yang akan dilakukan. Pada tahap ini, batasan-batasan dari *IT system* dikenali, bersama dengan *resources* dan informasi yang membentuk suatu sistem. Pada tahap pengenalan *IT system* akan dihasilkan cakupan *risk assessment effort*, penentuan batasan *operational authorization* dan menyediakan informasi-informasi (contohnya, software, *system connectivity* dan divisi-divisi yang bertanggung jawab serta *support personnel*) yang penting untuk mendefinisikan resiko-resiko.

Pada bagian selanjutnya akan dijelaskan mengenai *system-related information,* yang digunakan untuk mengenali IT system serta keadaan operasionalnya.

2.1.2.1.1 System-Related Information

Dalam mengidentifikasi resiko untuk suatu *IT system* dibutuhkan pemahaman yang baik dari keadaan *system process*. Pihak-pihak yang mengurus risk assessment ini pertama kali harus mengumpulkan informasi mengenai *system-related*, yang biasanya informasi-informasi tersebut berupa :

- Hardware
- Sofware
- System Interfaces (seperti *internal* dan *external connectivity*)
- Orang-orang yang mendukung dan menggunakan *IT system*
- System Mission (seperti proses apa saja yang dilakukan dengan *IT system*)
- System and Data Criticality (seperti nilai dari pentingnya *IT system* terhadap perusahaan)

- System and Data Sensitivity (merupakan level proteksi yang dibutuhkan untuk menjaga sistem dan *data integrity*, kerahasiaan data dan ketersediaan data)
- Pengguna Sistem (yaitu berupa *system user* yaitu orang-orang yang menyediakan *technical support* terhadap *IT system* serta *application user* yaitu orang-orang yang menggunakan *IT system* untuk menjalankan fungsi bisnis)
- Arsitektur Keamanan Sistem
- Hal-hal yang dibutuhkan oleh *IT system*
- Network Topology yang digunakan
- Informasi mengenai bagaimana sistem proteksi yang digunakan dapat melindungi sistem meliputi keamanan keseluruhan sistem, ketersediaan data dan kerahasiaan data
- Aliran dari informasi terhadap *IT system* (seperti *system interfaces, system input* dan *output flowchart*)
- Teknik kontrol yang digunakan untuk *IT system* (seperti *built-in* ataupun *add-on* dari produk keamanan yang digunakan oleh sistem untuk mendukung identifikasi dan autentikasi, akses kontrol, audit serta metode enkripsi yang digunakan)
- *Management control* yang digunakan untuk *IT system* (seperti rencana keamanan dan aturan perilaku)
- *Operational control* yang digunakan untuk *IT system* (seperti keamanan masing-masing pengguna, *backup*, *system maintenance*, *off-site storage*, *user account* mencakup pembuatan dan penghapusannya, pembedaan metode akses seperti *privileged/non-previliged user*)
- Lingkungan keamanan fisik dari *IT system* (seperti keamanan fasilitas serta aturan pada *data center*)
- Keamanan lingkungan ketika IT system diimplementasikan di suatu tempat (seperti *control* untuk kelembapan, air, *power supply*, polusi, temperatur dan zat kimia)

Output yang dihasilkan dari tahap pertama : **karakteristik dari *IT system*, penggambaran yang baik mengenai *IT system environment* serta batasan sistem.**

2.1.2.2 Threat Identification

Threat (ancaman) adalah potensi –potensi yang berbahaya yang dihasilkan oleh suatu sumber (*threat source*) yang dapat menyerang *vulnerability* yang dimiliki suatu sistem. Suatu sumber ancaman tidak dapat menghasilkan ancaman ketika tidak ada *vulnerability.*

2.1.2.2.1 Threat Source Identification

Tujuan pada tahap ini adalah seseorang dapat mengidentifikasi sumber-sumber apa saja yang dapat menghasilkan ancaman pada suatu *IT system.* Secara umum ada tiga macam *pembagian suatu sumber ancaman :*

- *Natural Threat* seperti banjir, gempa bumi, tornado, petir dan sebagainya
- *Human Threat* merupakan suatu kejadian yang dibuat atau disebabkan oleh seorang manusia, seperti kejadian yang tidak disengaja maupun hal yang disengaja (contohnya adalah penyerangan suatu jaringan, pengiriman software yang berbahaya bagi jaringan serta akses yang tidak boleh terhadap data pribadi perusahaan)
- *Enviromental Threat* seperti kesalahan *power* dalam waktu yang lama, polusi, zat kimia berbahaya dan kebocoran gas.

2.1.2.2.2 Motivation and Threat Action

Tabel 2.2 menjelaskan macam-macam motivasi serta hal-hal yang dilakukan yang berpotensi merusak keamanan suatu *IT system.* Informasi tersebut sangat penting digunakan oleh suatu perusahaan agar dapat mempersiapkan sehingga hal-hal tersebut tidak terjadi atau apabila telah terjadi dapat diambil langkah untuk melawan bahaya tersebut.

Tabel 2.2 Bentuk Ancaman yang Disebabkan oleh Manusia Meliputi Sumber Ancaman, Motivasi serta Bentuk Ancaman[1]

Threat-Source	Motivasi	Bentuk Ancaman
Hacker, Cracker	▪ Tantangan ▪ Ego ▪ Pemberontakan	▪ *Hacking* ▪ *Social Engineering* ▪ Mengganggu sistem ▪ Akses menuju bagian yang dilarang
Computer Criminal	▪ Perusakan informasi ▪ Mengubah data ▪ Mendapatkan Uang	▪ *Spoofing* ▪ Pengacauan sistem ▪ *Computer Crime* (seperti *cyber stalking*) ▪ *Fraudulent act* (seperti *replay, impersonation* dan *interception*) ▪ *Information bribery*
Terrorist	▪ Ancaman ▪ Perusakan ▪ Eksploitasi ▪ Balas dendam	▪ Bom ▪ *Information warfare* ▪ *System attack* ▪ *System penetration* ▪ *System tampering*
Pesaing Bisnis (perusahaan, pemerintah asing atau kepentingan lainnya)	▪ Pengintaian keuangan ▪ Persaingan	▪ Pencurian data ▪ *Economic exploitation* ▪ *Social engineering* ▪ Mengganggu keharasiaan seseorang ▪ Akses menuju bagian yang dilarang
Orang Dalam	▪ Keingintahuan ▪ Ego ▪ Mendapatkan uang	▪ Serangan kepada pegawai ▪ *Blackmail* ▪ Akses menuju

	▪ Balas dendam	informasi rahasia perusahaan ▪ Perusakan komputer ▪ Pencurian ▪ *Information bribery* ▪ *Interception* ▪ *Malicious code (virus, trojan)* ▪ *System bugs* ▪ *System sabotage* ▪ Akses menuju bagian yang tidak boleh ▪ Menjual informasi rahasia perusahaan

Output yang dihasilkan pada tahap ini : **daftar mengenai sumber-sumber bahaya yang dapat merusak sistem**

2.1.2.3 Vulnerability Identification

Vulnerability adalah cacat atau kelemahan pada suatu prosedur keamanan suatu sistem, disain, implementasi atau *internal control* yang dapat secara tiba-tiba dimunculkan atau dieksploitasi oleh pihak-pihak tertentu sehingga menghasikan suatu bentuk pelanggaran keamanan atau kesalahan pada aturan keamanan suatu sistem. Suatu analisis terhadap ancaman pada suatu *IT system* harus menyertakan analisis *vulnerability* suatu sistem. Tujuan dari tahap ini adalah untuk menghasilkan suatu daftar mengenai *system vulnerability* yang dapat dieksploitasi oleh sumber ancaman yang potensial

Tabel 2.3 memberikan contoh-contoh vulnerability beserta ancaman yang dihasilkan.

Tabel 2.3 Pasangan Vulnerability serta Ancaman[1]

Vulnerability	Threat-Source	Threat Action
System ID seorang pegawai yang telah dikeluarkan tidak dihilangkan dari sistem	Pegawai yang telah dikeluarkan	Mengakses jaringan perusahaan dan mengakses data rahasia perusahaan
Firewall perusahaan memperbolehkan *inbound* telnet dan *guest ID* di-*enable* pada XYZ server	*User* yang tidak diharapkan (seperti *hacker*, pegawai yang dikeluarkan, *computer criminal* dan *terrorist*)	Menggunakan telnet menuju XYZ server dan mengakses *system files* menggunakan *guest ID*
Vendor mengidentifikasi terdapat cacat pada desain keamanan suatu sistem; namun patch yang baru belum diimplementasikan pada sistem yang ada	*User* yang tidak diharapkan (seperti *hacker*, pegawai yang dikeluarkan, *computer criminal* dan *terrorist*)	Memanfaatkan kelemahan yang ada untuk mengambil file rahasia perusahaan
Suatu data center menggunakan air yang akan muncul ketika terjadi kebakaran; sedangkan kain terpal yang digunakan untuk melindungi hardware atau peralatan lainnya dari bahaya tersiram air tidak tersedia	Kebakaran, orang yang tidak diharapkan	Air keluar menyiram data center

Output yang dihasilkan dari tahap ini : **daftar *system vulnerability* yang dapat menimbulkan ancama keamanan pada *IT system***

2.1.2.4 Control Analysis

Tujuan akhir pada fase ini adalah untuk menganalisis bentuk *control* apa saja yang telah diimplementasikan atau direncanakan untuk diimplementasikan, oleh suatu perusahaan untuk meminimalkan atau menghilangkan kemungkinan terjadinya ancaman yang dapat menyerang *vulnerability* suatu sistem.

Untuk mendapatkan suatu rating/nilai dari keseluruhan kemungkinan ancaman yang dapat menyerang *vulnerability* suatu sistem, maka dibutuhkan implementasi kontrol/ rencana kontrol apa saja yang dilakukan apabila ancaman tersebut terjadi. Contohnya, suatu *vulnerability* memiliki kemungkinan rendah untuk diancam apabila si pengancam memiliki motivasi rendah untuk menyerang *vulnerability* tersebut ataupun disebabkan karena adanya kontrol yang sangat baik yang menyebabkan ancaman dapat dihilangkan atau dikurangi.

- Metode Kontrol

  Kontrol keamanan suatu sistem mencakup adanya metode *techical* dan *nontechnical*. *Technical Control* adalah pelindung yang *bulit-in* terdapat dalam computer hardware, software atau firmware (seperti mekanisme akses kontrol, mekanisme identifikasi dan autentikasi, metode enkripsi, *security monitoring software*). *Nontechnical control* adalah kontrol berupa *management* dan *operational*, seperti aturan keamanan, prosedur operasional serta keamanan karyawan, fisik dan lingkungan

- Kategori Kontrol

  Jenis kategori kontrol untuk metode *control technical* dan *nontechnical* lebih jauh lagi dapat dibagi menjadi *preventive* atau *detective*.

  1. *Preventive controls* mencegah adanya pelanggaran suatu aturan keamanan serta menyertakan mekanisme kontrol seperti *control enforcement*, *encryption* dan *authentication*.

  2. *Detective controls* memperingatkan adanya pelanggaran atau percobaan untuk melanggar suatu aturan keamanan sistem dan menyertakan mekanisme *control* seperti *audit trails*, metode *intrusion detection* dan *checksums*.

- Control Analysis Technique

  Pada suatu *IT system* perubahan yang terjadi pada kebutuhan keamanan yang dimiliki suatu perusahaan dapat saja terjadi. Oleh karena itu, kita harus menyesuaikan perubahan tersebut untuk diimplementasikan pada metode kontrol yang dimiliki, seperti perubahan aturan keamanan, metode serta kebutuhan.

Output pada tahap ini : **daftar kontrol yang sedang/ akan digunakan oleh *IT system* untuk mengatasi resiko yang akan muncul pada perusahaan**

2.1.2.5 Likelihood Determination

Untuk menghasilkan rating/nilai dari seluruh kemungkinan ancaman yang berpotensial merusak suatu sistem perusahaan, hal-hal dibawah ini perlu diperhatikan :

- Motivasi dan kemampuan dari sumber ancaman
- Karakterisitik dari *Vulnerability*
- Terdapatnya serta dimilikinya keefektifan dari kontrol yang dimiliki saat ini

Tabel 2.4 menjelaskan level-level dari kemungkinan suatu ancaman.

Tabel 2.4 Definisi Level Kemungkinan Ancaman[1]

Level Kemungkinan Ancaman	Definisi Kemungkinan Ancaman
High	Sumber ancaman memiliki motivasi tinggi dan kemampuan yang mencukupi untuk merusak sistem dan kontrol yang dimiliki untuk melindungi sistem tidak berfungsi efektif
Medium	Sumber ancaman memiliki motivasi tinggi dan kemampuan yang mencukupi untuk merusak sistem namun kontrol yang dimiliki dapat mengatasi ancaman tersebut
Low	Sumber ancaman memiliki motivasi rendah dan kemampuan yang tidak mencukupi untuk merusak sistem dan kontrol yang dimiliki dapat mengatasi ancaman tersebut

Output dari tahap ini : ***likelihood rating (high, medium, low)***

2.1.2.6 Impact Analysis

Langkah selanjutnya adalah menentukan kemungkinan dampak yang dapat muncul ketika suatu ancaman berhasil merusak sistem yang dimiliki suatu perusahaan. Berikut ini adalah contoh dampak-dampak yang dapat ditimbulkan ketika terjadi kerusakan tersebut.

- Loss of Integrity. Sistem dan *data integrity* berhubungan dengan informasi-informasi yang harus dilindungi dari perubahan yang tidak diharapkan. *Integrity* disebut hilang ketika perubahan yang tidak dihapkan terjadi pada suatu sistem yang disebabkan oleh hal yang disengaja atau tidak disengaja. Jika hal tersebut terjadi dapat menyebabkan *corrupted data* atau *contaminated data* yang menghasilkan terjadinya ketidakakuratan pengambilan suatu keputusan . Untuk semua hal-hal tersebut, *loss of integrity* mengurangi level kepercayaan pada *IT system*
- Loss of Availability. *Loss of availability* artinya *end-user* tidak dapat mengakses yang seharusnya sehingga menyebabkan terganggunya misi perusahaan.
- Loss of Confidentiality. *Loss of Confidetiality* artinya data rahasia suatu perusahaan diketahui oleh pihak lain, hal ini dapat menyebabkan hilangnya kepercayaan masyarakat, munculnya rasa malu ataupun penuntutan dari masyarakat.
- Hilangnya pendapatan/keuntungan serta biaya perbaikan yang tinggi untuk memperbaiki sistem sehingga bahaya yang sama tidak dapat menyerang sistem lagi

Output pada tahap ini: **rating penilaian dari dampak yang mungkin terjadi pada suatu sistem.**

Tabel 2.5 Definisi-Definisi *Magnitude of Impact*[1]

Magnitude of Impact	Impact Definition
High	Serangan pada suatu *vulnerability* (1) dapat menyebabkan biaya yang tinggi karena rusaknya *asset/resource*; (2) dapat secara signifikan menggangu atau merusak misi, reputasi sebuah perusahaan atau (3) menyebabkan kematian pada seseorang
Medium	Serangan pada suatu *vulnerability* (1) dapat menyebabkan biaya pada rusaknya *asset/resource*; (2) dapat menggangu atau merusak misi, reputasi sebuah perusahaan atau (3) menyebabkan cederanya seseorang
Low	Serangan pada suatu *vulnerability* (1) dapat menyebabkan biaya pada rusaknya beberapa *asset/resource*; (2) dapat mempengaruhi misi dan reputasi suatu perusahaan

2.1.2.7 Risk Determination

Tujuan pada fase ini adalah menetapkan nilai level dari resiko untuk *sebuah IT system*. Pengaruh dari resiko-resiko untuk pasangan *threat/vulnerability* merupakan fungsi dari :

- Kemungkinan dari sebuah sumber ancaman untuk merusak *vulnerability* yang dimiliki suatu sistem
- Dampak yang ditimbulkan suatu sumber ancaman terhadap *IT system*
- Kesanggupan dari kontrol keamanan yang dimiliki atau sedang direncanakan untuk mengurangi atau menghilangkan resiko

Hal yang selanjutnya diperlukan adalah membuat matrix yang terdiri dari input Level Kemungkinan Ancaman (fase 5) dengan input fase *Magnitude of Impact* (fase 6)

Tabel 2.6 Perhitungan *Risk Level* Pada *Risk Level Matrix*[1]

Level Kemungkinan Ancaman	**Impact**		
	Low (10)	Medium (50)	High (100)
High (1.0)	Low 10 x 1.0 = 10	Medium 50 x 1.0 = 50	High 100 x 1.0 = 100
Medium (0.5)	Low 10 x 0.5 = 5	Medium 50 x 0.5 = 25	Medium 100 x 0.5 = 50
Low (0.1)	Low 10 x 0.1 = 1	Low 50 x 0.1 = 5	Low 100 x 0.1 = 10

Risk Scale : High (>50 sampai 100); Medium (>10 sampai 50); Low (1 sampai 10)

Setelah *Risk Level Matrix* dibuat, hal yang selanjutnya diperlukan adalah menentukan masing-masing risk level untuk dijelaskan deskripsi masing-masing dan langkah-langkah selanjutnya yang diperlukan.

Tabel 2.7 Definisi Tiap-Tiap *Risk Level* dan Respon Terhadap Nilai Risk Level Tersebut[1]

Risk Level	**Deskripsi Resiko dan Langkah Selanjutnya Yang Diperlukan**
High	Apabila dalam perhitungan *Risk Scale* yang didapat adalah pada level high, maka diperlukan perbaikan pada sistem. Sistem yang sedang digunakan dapat terus berjalan, namun perbaikan harus terus dilakukan secepat mungkin
Medium	Apabila dalam perhitungan *Risk Scale* yang didapat adalah pada level medium, langkah perbaikan dibutuhkan dan sistem yang sedang berjalan harus mengimplementasikan perbaikan tersebut dalam waktu tertentu

Risk Level	Deskripsi Resiko dan Langkah Selanjutnya Yang Diperlukan
Low	Apabila dalam perhitungan *Risk Scale* yang didapat adalah pada level low, penanggung jawab suatu IT system harus menentukan apakah diperlukan suatu langkah perbaikan pada sistem atau memutuskan untuk menerima resiko yang ada

Output dari tahap 7 : **level resiko (*high, medium, low*)**

2.1.2.8 Control Recommendations

Selama langkah ini, kontrol yang dapat mengurangi/menghilangkan resiko-resiko yang teridentifikasi ditentukan. Tujuan dari rekomendasi kontrol tersebut adalah untuk mengurangi level resiko yang terjadi pada *IT system* dan data perusahaan menuju ke level yang lebih dapat diterima. Faktor-faktor dibawah ini perlu dipikirkan dalam merekomendasi suatu kontrol :

- Keefektifan dari pilihan rekomendasi yang dibuat (seperti *system compatibility)*
- Legislation dan regulation
- Aturan yang dimiliki suatu organisasi
- Dampak yang mungkin terjadi pada operasional perusahaan
- Keamanan dan ketahanan

Rekomendasi dari kontrol yang dibuat merupakan hasil dari proses yang terjadi pada fase risk assessment dan menyediakan input untuk fase risk mitigaion.

Harus diperhatikan bahwa tidak semua rekomendasi kontrol yang dibuat dapat diimplementasikan untuk mengurangi kerugian. Untuk menentukan rekomendasi manakah yang dapat diimplementasikan maka diperlukan analisis cost-benefit. Sebagai tambahan perhitungan dampak pada operasional perusahaan (seperti efek pada performa sistem) dan kemungkinan pada perusahaan (kesanggupan *user*, persyaratan teknis) harus juga diperhatikan.

Output pada tahap ini : **rekomendasi kontrol serta kontrol alternatif untuk mengatasi resiko**

2.1.2.9 Result Documentation

Setelah risk assessment proses selesai dilakukan (sumber ancaman dan *vulnerability* telah teridentifikasi, telah ada penentuan resiko serta rekomendasi kontrol telah tersedia), maka semua hasil tersebut harus didokumentasikan dengan baik dalam suatu bentuk *report*.

Report suatu risk assessment adalah suatu *management report* yang dapat membantu *senior management* serta *mission owners* untuk membuat pengambilan keputusan untuk pembuatan aturan, prosedur, biaya serta perubahan pada *system operational* dan *management*

Output pada langkah ini : ***Risk assessment report* yang menerangkan gangguan dan *vulnerability*, menghitung resiko yang ada serta menyediakan rekomendasi untuk implementasi kontrol yang akan digunakan**

2.1.3 Risk Mitigation

Pada tahap ini, risk mitigation, meliputi tahap pemprioritasi, evaluasi dan implementasi kontrol pengurangan resiko yang direkomendasikan dan diperoleh dari tahap risk assessment.

Karena proses menghilangkan semua resiko yang mungkin terjadi merupakan hal yang sangat tidak praktis dan mendekati mustahil, maka merupakan kewajiban dari *senior management/business managers* untuk menggunakan pendekatan *least-cost* dan mengimplementasikan kontrol yang paling penting untuk mengurangi level resiko ke level yang lebih dapat diterima, dengan mengedepankan dampak yang minimal pada *resource* dan misi organisasi.

2.1.3.1 Risk Mitigation Options

Risk mitigation merupakan suatu metodologi yang sistematis, digunakan oleh *senior management* untuk mengurangi resiko. Risk mitigation dapat dicapai menggunakan pilihan risk mitigation berikut ini:

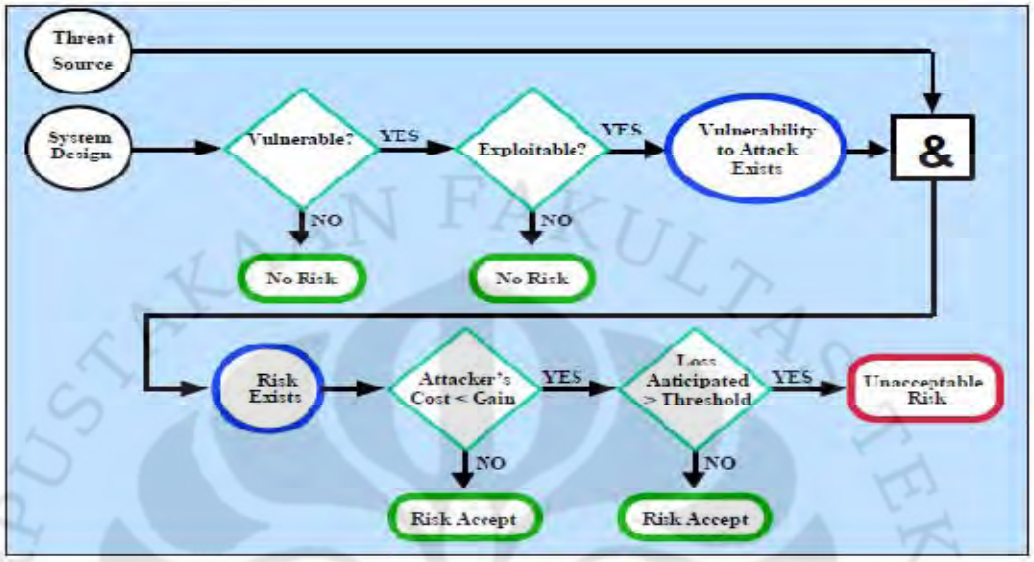
- Risk Assumption. Digunakan untuk menerima resiko-resiko yang potensial terjadi dan membuat operasi *IT system* tetap dapat terlaksana atau juga untuk mengimplementasikan kontrol untuk mengurangi resiko ke level yang lebih dapat diterima.
- Risk Avoidance. Digunakan untuk menghindari resiko dengan mengeliminasi penyebab resiko atau konsekuensi dari resiko.
- Risk Limitation. Untuk membatasi resiko yang mungkin terjadi dengan mengimplementasikan kontrol yang dapat meminimalkan kerugian dari dampak yang ditimbulkan .
- Risk Planning. Untuk mengatur resiko dengan mengembangkan rencana risk mitigation yang memprioritaskan, mengimplementasikan dan menjaga *control.*
- Research and Acknowledgemnet. Untuk mengurangi resiko kehilangan/kerugian dengan menyatakan cacat/*vulnerability* yang ada dan melakukan riset kontrol untuk mengoreksi cacat *vulnerability*/ pada sistem tersebut.
- Risk Transfer. Untuk memindahkan resiko dengan menggunakan pilihan-pilihan yang ada untuk mengkompensasi kehilangan, seperti membeli asuransi.

Tujuan dan misi suatu perusahaan harus dipikirkan dalam memilih semua pilihan risk mitigation yang ada. Merupakan hal yang mustahil untuk dapat mengidentifikasikan semua resiko yang ada, sehingga pemrioritasisasi perlu dilakukan untuk setiap *vulnerability* dan ancaman yang ada.

2.1.3.2 Risk Mitigation Strategy

Senior management/ pemiliki perusahaan, biasanya mengetahui resiko-resiko yang potensial terjadi pada perusahaan mereka dan mengetahui rekomendasi *control* yang harus dilakukan, mereka mungkin akan berkata, " Kapan dan pada kondisi yang bagaimana saya harus bertindak? Kapan saya harus mengimplementasikan *control* yang ada untuk mengurangi resiko dan melindungi organisasi?"

Risk Mitigation *chart* yang ada di gambar 2.3 menjawab pertanyaan-pertanyaan tersebut. Point-point tertentu untuk implementasi dari langkah control ditunjukkan dengan kata YES pada gambar tersebut.



Gambar 2.3 *Risk Mitigation Action Points*[1]

Strategi ini dijelaskan lebih lanjut pada penjelasan berikut ini:

- Ketika *Vulnerability* (atau cacat, kelemahan) terjadi -> melakukan langkah pengimplementasian teknik untuk memastikan terkuranginya *vulnerability* tersebut untuk dapat diekspolitasi.
- Ketika *Vulnerability* dapat dieksploitasi -> melakukan proteksi berlapis, disain yang terstrukur dan control administratif untuk meminimalkan resiko atau menghindarinya
- Ketika kerugian yang ditimbulkan dari serangan lebih kecil dari potensi pendapat yang mungkin didapatkan -> mengimplementasikan perlindungan untuk mengurangi motivasi penyerang dengan meningkatkan biaya dari serangan yang ditimbulkan (seperti menggunakan sistem *control* seperti membatasi sistem apa saja yang dapat diakses oleh *user*)

- Ketika kerugian terlalu besar -> mengimplementasikan *design principles, architectural designs* serta proteksi teknis dan nonteknis untuk membatasi serangan dapat meluas, sehingga dapat mengurangi kerugian selanjutnya.

2.1.3.3 Beberapa Pendekatan Dalam Implementasi Kontrol

Ketika suatu perusahaan memutuskan untuk mengambil langkah implementasi suatu kontrol, maka hal berikut ini perlu dipertimbangkan :

Atasi resiko terbesar yang muncul dan berusaha keras untuk mengatasi resiko tersebut dengan biaya yang terendah, dengan menghasilkan dampak yang terkecil pada perusahaan.

Untuk mencapai hal tersebut, maka diperlukan metodologi yang terstruktur pada tahap risk mitigation, yaitu :

1. Langkah-Langkah Memprioritaskan *Resource*

Berdasarkan level resiko yang diperoleh dari *risk assessment report*, dilakukan langkah memprioritaskan langkah-langkah yang penting. Untuk *resource* yang dimiliki, prioritas tertinggi harus diberikan pada resiko yang memiliki *rating* yang tinggi. *Vulnerability/threat* tersebut harus segera diatasi untuk melindungi *resource* yang dimiliki perusahaan.

Output dari tahap ini : ***action rangking dari High sampai Low***

2. Evaluasi Pilihan Rekomendasi Kontrol

Rekomendasi kontrol yang diperoleh dari *risk assessment process* mungkin bukan merupakan pilihan yang paling cocok atau tepat untuk perusahaan. Pada tahap ini, *feasibility* (seperti *compatibility, user acceptance*) serta *effectiveness* (seperti level proteksi yang dimiliki dan level risk mitigation) dari pilihan rekomendasi kontrol tersebut dianalisis dan dievaluasi. Tujuan pada langkah ini adalah untuk memilih pilihan rekomendasi kontrol apa saja yang dapat mengurangi resiko yang ditimbulkan.

Output dari tahap ini : **daftar kontrol-kontrol yang telah dievaluasi**

3. Melakukan Analisis Cost-Benefit

Untuk membantu *management* dalam pengambilan keputusan dan mengidentifikasikan kontrol yang efektif secara biaya, maka analisis cost-benefit perlu dilakukan.

Output dari tahap ini : **dilakukannya analisis cost-benefit yang menjelaskan biaya dan keuntungan dari diimplementasi/ tidak diimplementasikannya suatu kontrol.**

4. Memilih Kontrol

Berdasarkan hasil yang diperoleh dari analisis cost-benefit, *management* kemudian menentukan kontrol apa saja yang paling efektif secara biaya untuk mengurangi resiko yang akan muncul pada perusahaan. Kontrol yang dipilih harus menggabungkan elemen kontrol *technical*, *operational* dan *management* untuk memastikannya cukup untuk mengamankan *IT system* dan perusahaan.

Output dari tahap ini : **kontrol yang telah dipilih**

5. Memberikan Tanggung Jawab

Pada langkah ini dipilih orang–orang yang memiliki kemampuan yang memadai untuk mengimplementasikan kontrol yang telah dipilih sebelumnya

Output dari tahap ini : **daftar orang-orang yang diberikan tanggung jawab**

6. Mengembangkan Rencana Implementasi Yang Aman

Pada tahap ini dibuat suatu tabel yang berisi rencana implementasi yang aman untuk suatu perusahaan. Tabel tersebut berisi :

- *Vulnerability/threat* yang terjadi
- Level Resiko
- Rekomendasi Kontrol
- Langkah-Langkah Memprioritas
- Rencana Kontrol Yang Dipilih
- *Resource* Yang Dibutuhkan
- Orang-Orang Yang Dibutuhkan
- Jadwal Pelaksanaan
- Kebutuhan Pemeliharaan

Output yang didapat dari tahap ini : **rencana implementasi yang aman untuk suatu perusahaan.** Pada Lampiran A terdapat contoh tabel rencana implementasi yang aman.

7. Mengimplementasi Kontrol Yang Telah Dipilih

Pada situasi tertentu, kontrol yang diimplementasikan dapat mengurangi resiko tetapi tidak menghilangkannya.

Output yang didaptkan dari tahap ini : **residual risk**

2.1.3.4 Evaluation dan Assessment

Pada sebagian besar perusahaan, jaringan yang dimiliki biasanya akan terus-menerus mengalami perkembangan, meliputi sistem yang dimiliki, komponen-komponen yang dimiliki serta *software-application* yang akan diganti atau mengalami *update* dengan versi terbaru. Selain itu, perubahan juga akan meliputi pergantian anggota serta perubahan aturan keamanan yang dimiliki. Setiap perubahan tersebut menimbulkan permasalahan karena resiko-resiko baru akan muncul serta resiko yang sebelumnya telah dapat diatasi akan muncul kembali. Oleh karena itu, proses *risk management* harus terus berjalan dan mengalami perkembangan.

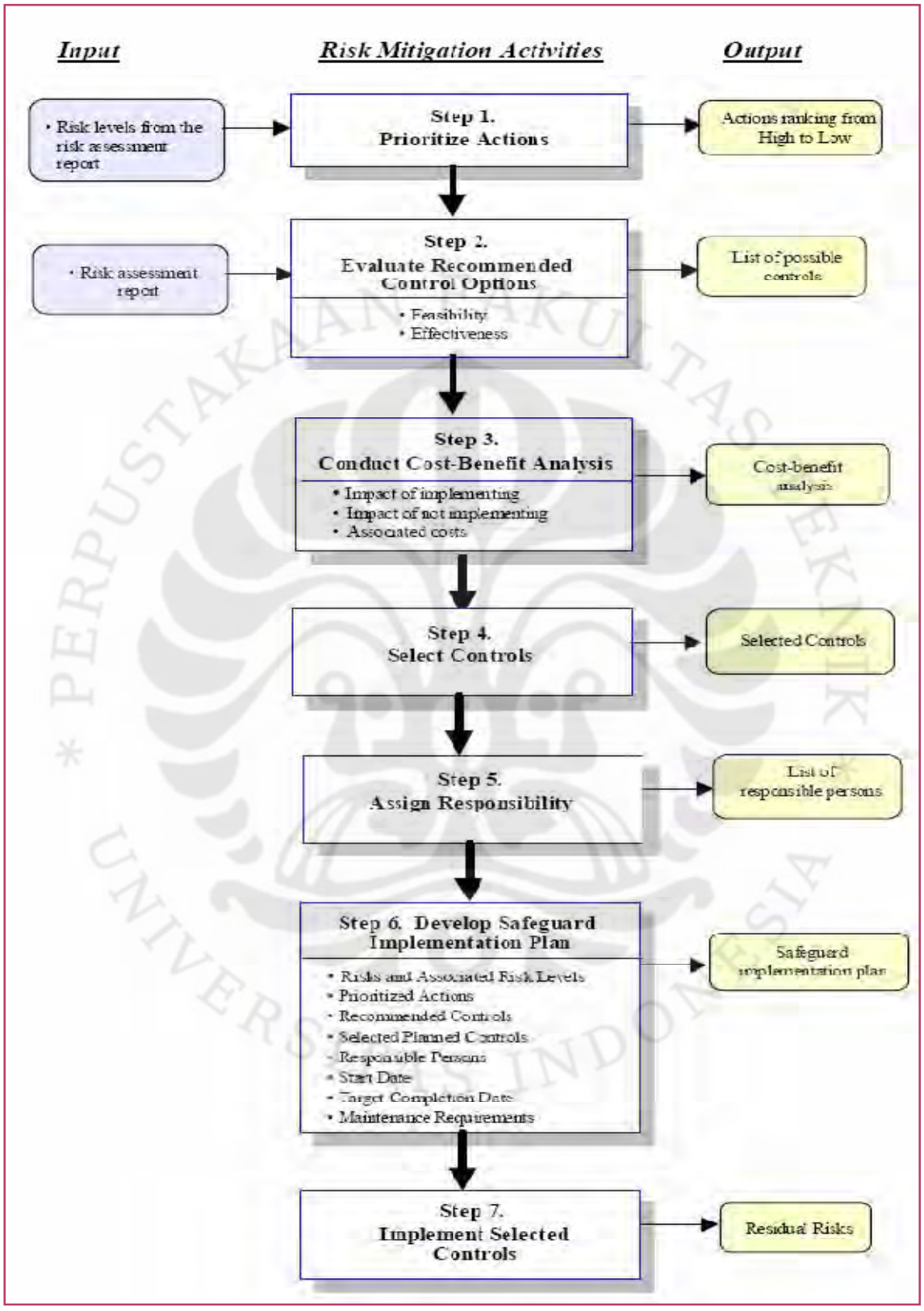
2.1.3.4.1 Latihan Pengamanan Sistem

Proses *risk assessment* biasanya harus mengalami pengulangan setiap tiga tahun sekali. Selain itu, proses *risk management* juga harus terintegrasi ke dalam SDLC pada *IT system,* bukan hanya karena ketentuan tersebut telah diatur dalam hukum/ regulasi, namun juga hal-hal tersebut sangat baik untuk latihan dan dapat membantu mendukung misi dan tujuan suatu perusahaan. Harus ada jadwal tertentu yang teratur untuk menentukan dan mengatasi resiko yang ada sehingga ketika perubahan terjadi pada *IT system* sehingga menyebabkan terjadinya perubahan aturan dan adanya teknologi baru, maka resiko-resiko yang muncul dapat diatasi dengan baik.

2.1.3.4.2 Kunci Untuk Sukses

Risk management yang berhasil biasanya bergantung pada :

1. Komitmen yang dimiliki *senior management.*
2. Partisipasi dan dukungan penuh dari *IT Team.*
3. Kompetensi yang memadai yang dimiliki *risk assessment team* dalam mengimplementasikan *risk assessment* metodologi, mengidentifikasikan resiko serta menyediakan cost-*effective safeguards* berdasarkan kebutuhan perusahaan.
4. Kepedulian setiap pengguna sistem dengan mengikuti setiap prosedur yang telah ditetapkan
5. Evaluasi berkelanjutan dari IT Risk Management yang dimiliki.



Gambar 2.4 *Flowchart* Metodologi Risk Mitigation[1]

2.2 Security Monitoring Tools

Bayangkan seseorang pria asing berdiri didepan rumah. Ia melihat sekeliling, mempelajari lingkungan yang ada di sekitar dan selanjutnya pergi mendekati pintu masuk dan memulai membuka kenop pintu, namun ternyata pintu terkunci. Kemudian, ia pindah mendekati jendela dan mencoba membukanya, namun ternyata jendelanya juga terkunci. Kelihatannya rumah tersebut aman. Jadi, mengapa memasang alarm di rumah tersebut? Pertanyaan ini sering ditanyakan berhubungan dengan perlu tidaknya suatu sistem dipasangi *intrusion system.* Mengapa harus kesulitan dan mahal memasang *intrusion system* jika pada sistem tersebut sudah terpasang *firewall*, OS yang sudah di-patch dan sudah adanya mekanisme pengecekan password? Jawabannya sederhana: karena intrusion kemungkinan besar masih terjadi! Ambil contoh lagi ketika seseorang lupa mengunci pintu atau ketika *firewall* yang digunakan salah dikonfigurasi!

Bahkan untuk sistem perlindungan yang sangat canggih sekalipun, sistem komputer tidak sepenuhnya aman. Bahkan para ahli keamanan komputer berpendapatan, sistem jaringan komputer yang sempurna tidak mungkin dapat dicapai. Oleh karena itu, seorang sistem administrator atau pemilik suatu sistem jaringan komputer harus mengembangkan suatu teknik deteksi gangguan untuk dapat merespon apabila terjadi serangan pada sistem yang dimilikinya.

Intrusion system merupakan *tools* yang sangat membantu system administrator dalam membuat IT Risk Management. Secara umum, ada 2 macam bentuk *intrusion system*, *intrusion detection system* dan *intrusion prevention system,* namun pada skripsi ini hanya akan dibahas mengenai *intrusion detection system.*

2.2.1 Intrusion Detection System

Intrusion Detection System(IDS) adalah software dan/atau hardware yang didisain untuk mendeteksi akses yang tidak diinginkan, manipulasi yang ilegal atau men-*disable* suatu fungsi pada sistem jaringan komputer[8]. Akses-akses yang tidak diinginkan tersebut biasanya berupa serangan yang tidak diharapkan dan dapat membahayakan sistem jaringan komputer dan juga data yang tersimpan di dalamnya, contohnya adalah *crackers*, *malware* atau akses ilegal dari orang-

orang yang tidak diharapkan, seperti pesaing usaha atau pihak internal(karyawan yang ingin mengambil keuntungan dengan cara mengambil data perusahaan).

Suatu IDS digunakan untuk mendeteksi beberapa tipe perilaku yang berbahaya (*malicious behaviours*) yang dapat membuat sistem komputer mempercayai akses tersebut sehingga sistem dapat berubah sesuai keinginan *attacker*. Serangan-serangan yang terjadi dapat berupa serangan melalui jaringan terhadap *vulnerability* yang dimiliki oleh suatu sistem, serangan untuk menyerang aplikasi pada sistem, serangan berbasis host seperti *privilege escalation*, *unauthorized logins*, akses terhadap file-file pribadi dan rahasia perusahaan dan juga berupa *malware* (*virusses*, *trojan horses* dan *worms*).

Suatu IDS secara umum terdiri atas tiga komponen yang memiliki fungsi yang sangat penting, yaitu sebagai berikut[8]:

1. *Information Sources*: sumber-sumber yang berbeda dari informasi suatu *event*, digunakan untuk menentukan apakah *intrusion* telah terjadi. Sumber-sumber tersebut dapat digambarkan dari berbagai level sistem, biasanya proses *monitoring* dapat berupa *monitoring* pada level *network, host* maupun *application.*

2. *Analysis :* merupakan bagian *intrusion detection system* yang mengatur dan *makes sense* (merasakan) apabila suatu ancaman terjadi, kemudian memutuskan kapan ancaman tersebut akan ditangani. Hal ini berhubungan erat dengan teknik deteksi yang digunakan oleh suatu *intrusion detection*, seperti *misuse detection* dan *anomaly detection.*

3. *Response:* merupakan sekumpulan tindakan yang akan dilakukan setiap sistem mendeteksi adanya gangguan. Biasanya tindakan yang akan dilakukan dibagi menjadi dua pendekatan, yaitu *active* dan *passive response. Active response* meliputi adanya intervensi secara automatis pada beberapa bagian sistem, sedangkan pada *passive response* tindakan yang dilakukan meliputi pelaporan deteksi yang ditemukan kepada *system administrator*, untuk kemudian mereka akan mengambil langkah-langkah sendiri.

a. *Active Responses*: ada 3 kategori yang merupakan bentuk dari *active responses.*

1. Mengumpulkan informasi tambahan. Apabila terjadi suatu gangguan pada sistem, maka IDS akan mengumpulkan informasi tambahan mengenai serangan yang diterima tersebut. Langkah yang dilakukan IDS biasanya meliputi peningkatan level sensitivitas dari sumber informasi (contohnya, peningkatan jumlah log pada sistem atatu peningkatan jumlah paket yang akan di-*capture*). Mengumpulkan informasi tambahan merupakan hal yang sangat membantu. Informasi tambahan yang didapat dapat membantu mendeteksi serangan. Pilihan ini juga membantu suatu perusahaan mendapatkan informasi yang dapat digunakan untuk membantu investigasi pihak penyerang untuk kemudian diambil tindakan hukum.
2. Mengubah Lingkungan Sistem. Langkah lainnya yang dilakukan oleh IDS adalah menghentikan serangan yang sedang terjadi dan kemudian mengeblok akses selanjutnya yang akan dilakukan oleh penyerang. Biasanya IDS tidak memiliki kemampuan untuk mengeblok akses seseorang, namun IDS memiliki kemampuan untuk dapat mengeblok *IP Address* dari penyerang yang akan datang selanjutnya. Merupakan hal yang sangat sulit untuk bisa mengeblok suatu *IP Address* tertentu, tetapi IDS dapat menghalangi serangan dengan melakukan langkah-langkah berikut ini:
    - Melakukan injeksi *TCP reset packets* pada koneksi penyerang yang menyerang suatu sistem, sehingga koneksi tersebut akan putus.
    - Melakukan konfigurasi ulang router dan firewall untuk mengeblok paket-paket yang dikirim penyerang.
    - Melakukan konfigurasi ulang *router* dan *firewall* untuk mengeblok *network ports*, *protocols* atau *services* yang sedang digunakan penyerang.

- Dalam kondisi yang sangat berbahaya IDS dapat mengkonfigurasi ulang *router* dan *firewall* untuk memutuskan semua koneksi yang menggunakan *network interfaces* tertentu.

3. Melakukan Serangan Balik: Serangan balik dapat meliputi konfigurasi ulang router untuk mengeblok alamat penyerang dan selanjutnya menyerang balik penyerang. Namun hal ini dapat berakibat buruk karena dapat saja yang diserang balik adalah pihak yang tidak bersalah. Hal ini dapat terjadi, sebagai contoh, ketika *hacker* menyerang suatu jaringan menggunakan *spoofed traffic*(*traffic* yang terlihat akan berasal dari alamat tertentu, namun sebenarnya di-*generate* dari tempat lain). IDS kemudian akan mendeteksi serangan, mengeblok alamat serangan dan melakukan serangan balik (biasanya berupa mengeksekusi *denial-of-service*) kepada *site* yang tidak bersalah

b. *Passive Responses*: ada 2 kategori yang termasuk *passive responses.*

1.Alarm dan Peringatan. Alarm dan Peringatan akan dikeluarkan IDS untuk memberitahukan *system administrator* ketika serangan terjadi dan terdeteksi. Bentuk alarm yang paling umum adalah peringatan pada layar berupa *poup window*. Bentuk lainnya dari alarm maupun peringatan adalah meliputi pengiriman alarm dan peringatan jarak jauh, seperti pengiriman pesan tersebut melalui *cellular phone*, *pager* bahkan email sehingga sangat membantu apabila diimplementasikan pada suatu perusahaan.

2. SNMP Traps and Plug-ins: beberapa IDS didisain untuk dapat mengeluarkan alarm dan peringatan untuk kemudian melaporkannya ke suatu *network management system*. Hal ini menggunakan *SNMP traps and messages* untuk kemudian mengirimkannya ke *central network management consoles*, yang kemudian diterima orang-orang yang diinginkan. Beberapa keuntungan yang diperoleh dengan model seperti ini adalah kemampuan untuk dapat beradaptasi pada keseluruhan

infrastruktur jaringan suatu perusahaan untuk dapat merespon apabila terdeteksi serangan.

Tujuan dari intrusion detection system adalah sangat sederhana, yaitu untuk mendeteksi apabila terjadinya gangguan/serangan pada sistem. Namun untuk mencapainya sangat sulit, hal itu disebabkan suatu IDS sulit untuk mendeteksi semua gangguan. IDS hanya mendeteksi petunjuk/keterangan/bukti dari gangguan yang terjadi, apakah kejadian tersebut dalam progress atau telah terjadi.

Beberapa petunjuk/keterangan/bukti yang ditemukan biasanya berhubungan dengan bentuk/manifestasi suatu serangan. Jika manifestasi tidak ada, atau kurangnya informasi mengenai suatu manifestasi serangan atau informasi yang diperoleh tidak dapat dipercaya, maka sistem tidak dapat mendeteksi gangguan/intrusion tersebut.

Sebagai contoh, suatu kamera CCTV pada sistem pengamanan rumah yang menampilkan seseorang yang berdiri didepan pintu. Data video dari camera tersebut merupakan manisfetasi terjadinya gangguan. Jika lensa camera tersebut kotor/ *out of focus*, sistem tidak dapat menentukan apakah orang tersebut pencuri atau pemilik.

2.2.1.1 Control Strategy

Pada *Control Strategy* dijelaskan mengenai bagaimana komponen-komponen pada IDS dikontrol dan juga bagaimana input dan output pada IDS diatur. Biasanya dari segi *control strategy* yang digunakan ada 3 macam bentuk yang digunakan, yaitu sebagai berikut[8]:

- **Centralized**

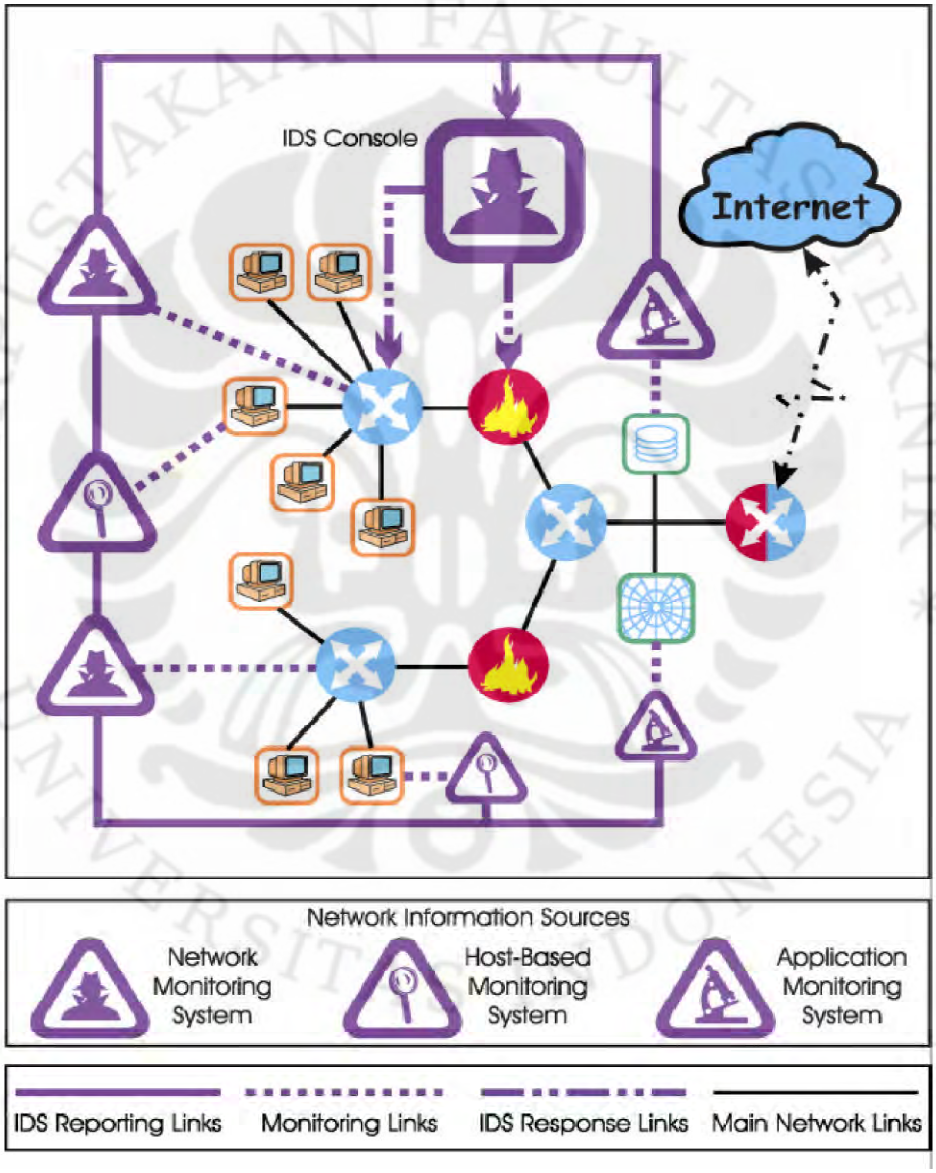
  Pada jenis *control strategy* dengan model *centralized*, semua kegiatan *monitoring, detection* dan *reporting* dikontrol langsung secara terpusat.

- **Partially Distributed**

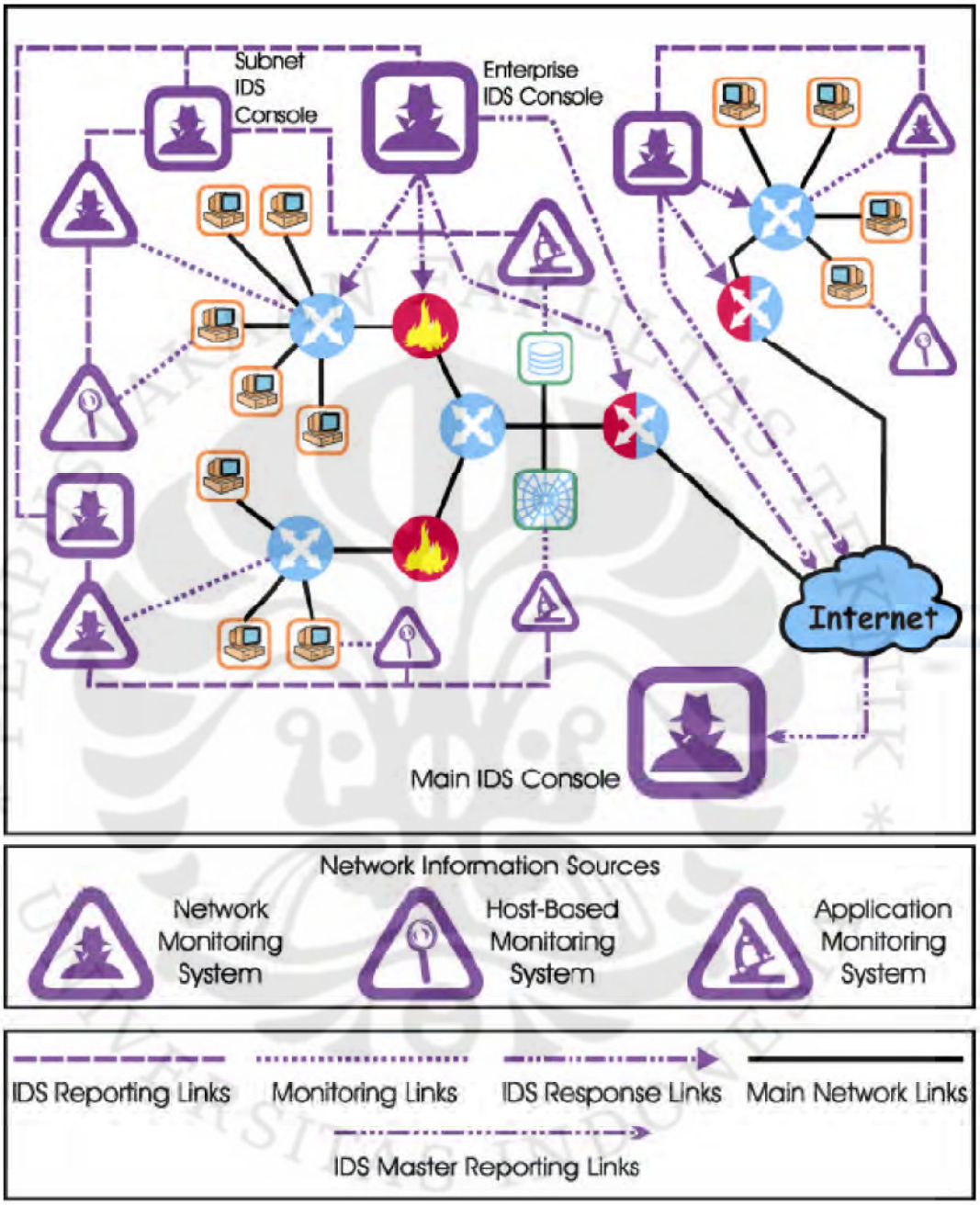
  Pada jenis ini, proses *monitoring* dan *detection* dikontrol dari *node* lokal, dengan adanya sistem pelaporan yang bertingkat (*hierarchical)* menuju satu atau lebih lokasi yang terpusat.

- **Fully Distribute**

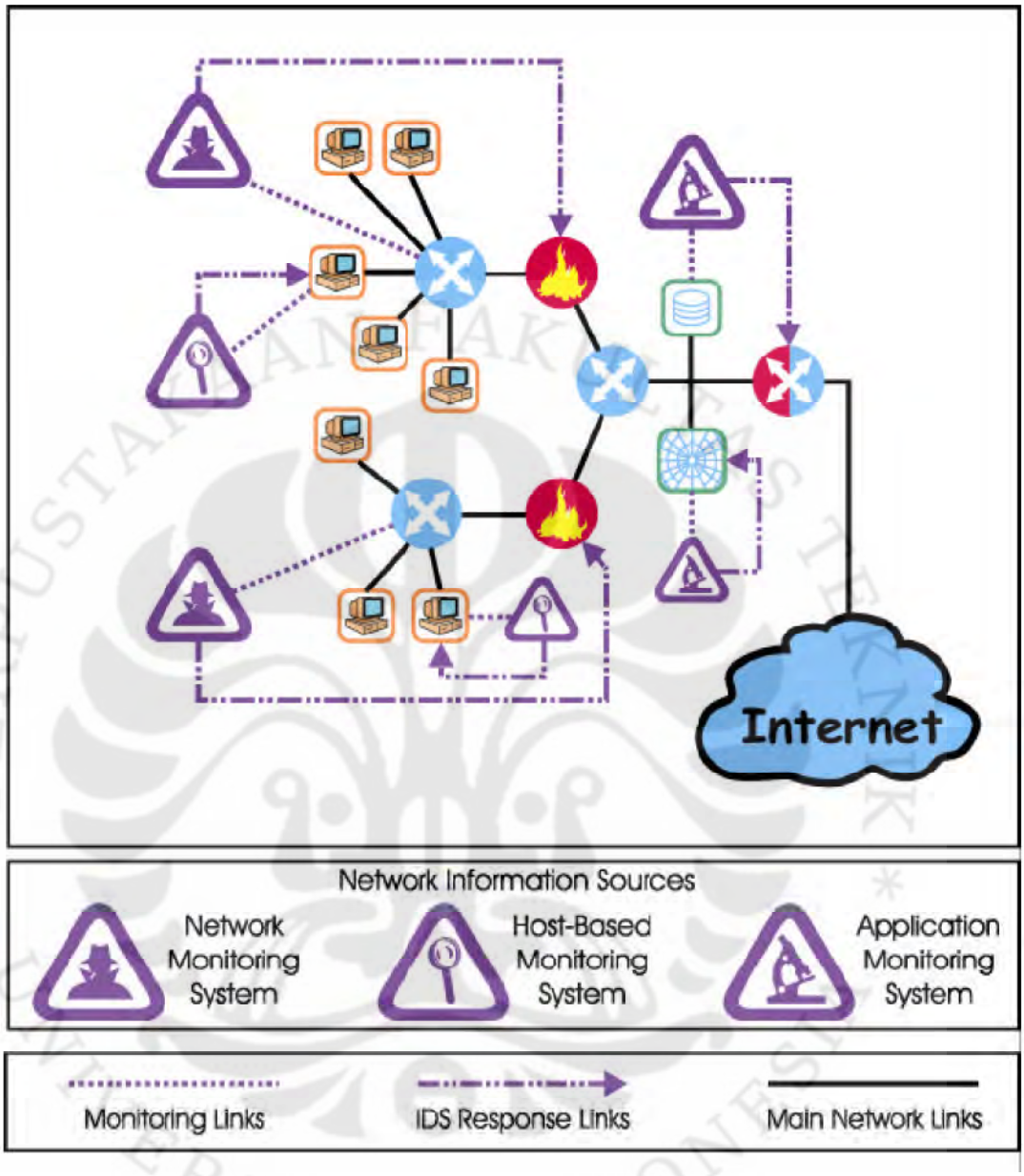
Pada jenis ini, proses *monitoring* dan *detection* dilakukan menggunakan pendekatan berbasis agen, yang mana keputusan dibuat pada analisis titik tertentu.



Gambar 2.5 Kontrol Secara *Centralized* Pada IDS[8]



Gambar 2.6 Kontrol Secara *Partially Distributed* Pada IDS[8]



Gambar 2.7 Kontrol Secara *Fully Distributed* /Berbasis Agen Pada IDS[8]

2.2.1.2 Masalah-Masalah Dalam Pengumpulan Data

Agar diperoleh *intrusion detection* yang yang akurat, maka sebuah sistem harus memiliki data yang *reliable* dan lengkap mengenai aktivitas sistem yang akan diamati. Pengumpulan data yang *reliable* dapat menjadi isu tersendiri yang sangat komplex. Beberapa OS menawarkan sistem audit yang menyediakan operation log-log tersendiri untuk *user* yang berbeda-beda. Log tersebut dapat

dibatasi untuk *security-relevan events* (seperti login yang salah) atau mereka dapat menawarkan *report* yang lengkap pada setiap *system call* meliputi setiap proses. Hal yang sama terjadi untuk router dan *firewall* yang menyediakan suatu *event log* untuk suatu aktivitas jaringan. Log-log tersebut dapat mengandung informasi simpel, seperti *network connection openings* and *closing*, atau catatan lengkap mengenai paket-paket apa saja yang ada pada suatu sambungan.

Jumlah dari data yang dikumpulkan yang dihasilkan oleh perangkat-perangkat tersebut merupakan hasil pertimbangan antara *overhead* dan *effectiveness*. Suatu sistem yang mencatat setiap kegiatan secara detail dapat saja mengalami penurunan *performance* serta memerlukan area penyimpanan data yang cukup besar. Sebagai contoh, dalam mengumpulkan log dari suatu jaringan yang menggunakan link 100-Mbit Ethernet dapat membutuhkan ratusan Gbytes setiap hari.

Proses mengumpulkan informasi sangat mahal, serta juga penting untuk mengumpulkan data yang benar. Dalam menentukan informasi untuk dikumpulkan dan dimana untuk mengumpulkan merupakan suatu permasalahan.

2.2.1.3 Teknik-Teknik Deteksi Pada Intrusion Detection System

Proses audit sistem yang dimiliki dapat menjadi sia-sia jika tidak dilakukan analisis yang memadai terhadap data yang telah dikumpulkan. Proses bagaimana suatu IDS mengumpulkan data merupakan suatu hal yang perlu dilakukan. Secara umum ada dua kategori IDS *techniques*: anomaly detection dan misuse detection.

1. Anomaly Detection.

   Pada anomaly detection digunakan suatu model yang mencatat/mengenali *behaviour* dari *user* dan aplikasi pada suatu sistem, apabila sistem mendeteksi terjadinya perilaku yang berbeda, maka sistem akan mengenalinya sebagai suatu masalah.

   Asumsi dasar yang digunakan adalah sistem akan mengenali serangan yang terjadi sebagai suatu hal yang berbeda dari hal yang normal terjadi pada sistem. Sebagai contoh, suatu kegiatan *user* sehari-hari dapat dijadikan sebagai suatu model. Anggap perilaku *user* dicatat pada suatu log yang menggunakan komputer sekitar jam 10 pagi, meliputi membaca e-mail,

melakukan transaksi database, istirahat siang sampai jam 1, dsb. Jika sistem melihat hal yang sama terjadi namun pada jam 3 pagi, menggunakan *tools compiler* dan *debugging*, dan melakukan akses file, maka sistem akan menandai aktivitas tersebut sebagai hal yang mencurigakan.

Kelebihan utama dari sistem dengan model anomaly detection adalah sistem dapat mendeteksi apabila terjadi serangan baru yang belum pernah terjadi sebelumnya. Dengan mendefinisikan hal-hal apa saja yang normal, sistem dapat mendeteksi apabila terjadi pelanggaran, entah hal tersebut bagian dari ancaman atau bukan. Namun, hal tersebut dapat juga menjadi kelemahan karena suatu sistem yang mendeteksi serangan yang sebelumnya belum dikenali dapat menyebabkan kesalahan pendeteksian pada kejadian-kejadian baru yang sebenarnya bukan merupakan kesalahan. Sehingga sistem seperti ini sulit untuk dilatih/diterapkan pada sistem yang sering mengalami perubahan.

2. Misuse Detection

Pada dasarnya sistem dengan model ini akan mendefinisikan apa yang salah. Sistem menyimpan deskripsi serangan(*signatures*) dan mencocokkannya dengan yang terdapat pada database, serta melihat buktinya untuk melihat bentuk serangan yang sebenarnya terjadi. Sebagai contoh, ketika *user* membuat *symbolic* link untuk suatu file yang menggunakan sistem password Unix dan mengeksekusi suatu *privileged application* yang mengakses *symbolic link* tersebut. Pada contoh ini, serangan akan mengeksploitasi kekurangan yang dimiliki oleh sistem pengecekan *file access.*

Kelebihan utama yang dimiliki oleh misuse detction systems adalah sistem akan berfokus pada analisis mengenai audit data yang ditangani dan biasanya akan menghasilkan kesalahan yang sedikit. Sedangkan kelemahan dari sistem ini adalah sistem hanya dapat mendeteksi serangan yang telah diketahui sebelumnya yang telah didefinisikan pada sistem, apabila terjadi serangan dengan bentuk yang baru, maka terlebih dahulu harus ditambahkan pada *signature database* sistem.

2.2.1.4 Tipe-Tipe Intrusion-Detection Systems

Pembagian paling umum dalam mengklasifikasikan IDS adalah dengan mengelompokkannya berdasarkan sumber informasi yang diperoleh. Beberapa IDS menganalisis *network packets*, meng-*capture* dari *network backbones/LAN segments*, untuk menemukan penyerang. IDS lainnya menganalisis sumber informasi yang dikeluarkan oleh *Operating System* atau *application-software* untuk menemukan gangguan yang terjadi[8].

1. **Network-Based IDS (NIDS)**

   IDS tipe ini mendeteksi serangan dengan meng-*capture* dan menganalisis paket-paket pada suatu jaringan. Dengan melakukan proses "mendengar" pada *network segment* atau *switch*, sebuah NIDS dapat memonitor *network traffic* beberapa *host* yang terhubung pada *network segment* tersebut.

   Suatu NIDS biasanya terdiri atas sebuah sensor yang dapat ditempatkan pada beberapa titik pada jaringan. Sensor tersebut bertugas memonitor *network traffic*, melakukan analisis dari *network traffic* yang diperoleh serta melaporkannya.

2. **Host-Based IDS**

   Pada IDS dengan tipe ini, IDS tersebut bekerja dengan mengumpulkan informasi dari sebuah komputer saja. Hal tersebut membuatnya dapat menganalisis kegiatan yang dilakukan sebuah *host* dengan *reliability* dan level keakuratan yang sangat baik, meliputi penentuan proses-proses apa saja dan siapa saja yang mengalami gangguan dari suatu serangan yang terjadi pada suatu OS. Selain itu, *host-based* IDS dapat melihat langsung akibat yang ditimbulkan dari suatu serangan, karena IDS tipe ini dapat melakukan akses langsung dan dapat memonitor *data files* dan *system processes* komputer yang diserang.

3. **Application-Based IDS**

   IDS tipe ini dapat menganalisis gangguan yang terjadi hingga menuju level *software-application*. Sumber informasi yang paling umum digunakan oleh *application-based* IDS adalah *log files* suatu *application transaction*. Kemampuan untuk menuju level aplikasi bisa membuat IDS tipe ini untuk dapat mendeteksi perilaku mencurigakan yang disebabkan oleh *authorized*

user yang melebihi level autorisasi yang dimilinya. Hal ini disebabkan beberapa masalah biasanya terjadi berhubungan erat dengan interaksi antara *user*, data dan aplikasi.

2.2.1.5 Mengembangkan Network-Based IDS

NIDS merupakan tipe IDS yang paling banyak digunakan. Hal ini disebabkan karena NIDS memiliki kemampuan untuk dapat memonitor *network traffic* beberapa *host* yang terhubung pada satu *network segment*. Ada beberapa pilihan dalam mengembangkan NIDS pada suatu jaringan, hal ini berhubungan erat dengan pilihan dimana menempatkan *system sensor*. Ada beberapa pilihan dalam penentuan system sensor, yaitu:

1. Dibelakang *external firewall*. (lokasi 1)

   Keuntungan:

   - Dapat melihat serangan yang berasal dari luar jaringan yang dapat menembus pertahanan sistem keamanan jaringan.
   - Dapat menandai masalah yang terjadi menggunakan aturan yang terdapat pada *firewall.*
   - Dapat melihat serangan yang mungkin mengincar *web server/ftp server.*
   - Bahkan ketika serangan tidak terjadi, IDS dapat melihat *traffic* yang keluar dari suatu jaringan.

2. Diluar *external firewall.* (lokasi 2)

   Keuntungan:

   - Dapat mendokumentasikan sejumlah serangan yang berasal dari internet yang akan menyerang jaringan.
   - Dapat mendokumentasikan berbagai macam tipe serangan yang berasal dari internet yang akan menyerang jaringan

3. Pada *network backbones* (lokasi 3)

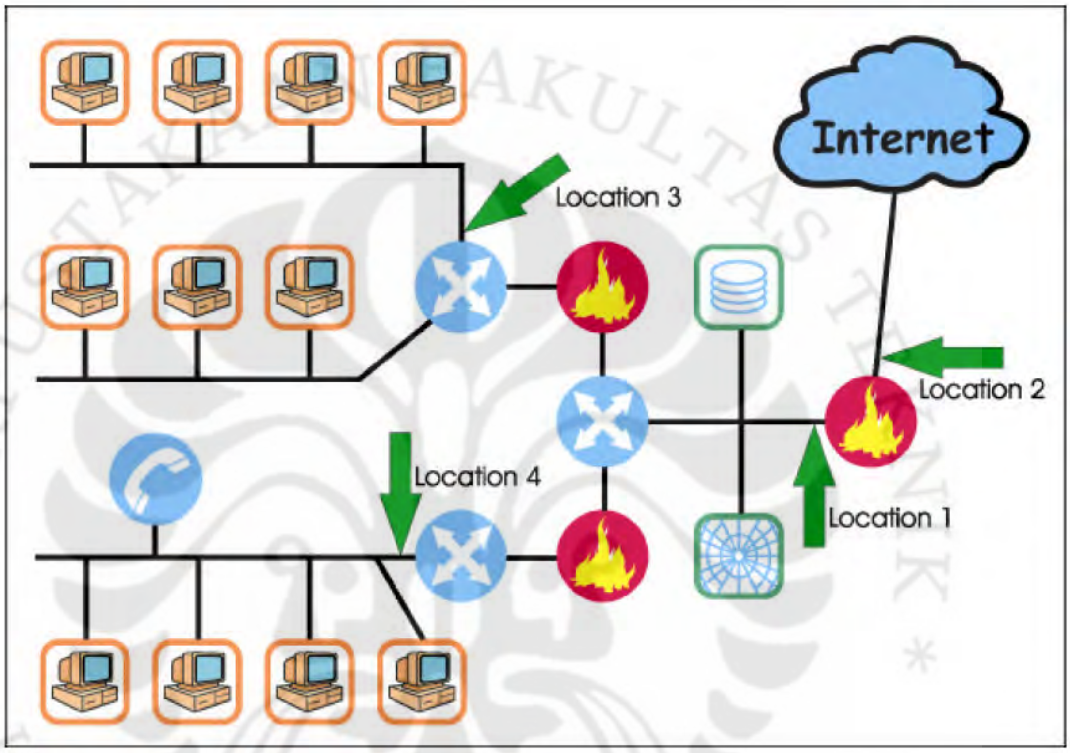
   Keuntungan:

   - Dapat memonitor *network traffic* dalam jumlah yang lebih banyak, sehingga dapat meningkatkan kemungkinan terdeteksinya serangan.
   - Dapat mendeteksi *unauthorized activity* dari *authorized users* dalam suatu jaringan.

4. Pada *critical subnets* (lokasi 4)

Keuntungan:

- Dapat mendeteksi serangan yang menyerang sistem dan resource yang sangat penting.
- Dapat melakukan pendeteksian pada resource yang lebih terbatas untuk network assest tertentu.



Gambar 2.8 Beberapa Lokasi Penempatan *System Sensors* Pada Network-Based IDS[8]

2.2.1.6 Masalah-Masalah dalam Pengembangan Intrusion Detection System

Walaupun *intrusion detection system* telah berkembang sedemikian cepat dan baik dalam beberapa tahun terakhir, namun masih ada beberapa isu penting yang masih terjadi. Pertama, *intrusion detection system* harus bekerja lebih efektif, mendeteksi dengan cakupan jenis serangan yang lebih luas dengan kesalahan positif yang lebih sedikit. Kedua, *intrusion detection system* harus tetap dapat menghadapi dengan peningkatan ukuran, kecepatan dan dinamika yang biasanya

dimiliki oleh jaringan modern. Terakhir, kita perlu menggunakan teknik analisis yang mendukung identifikasi serangan yang menyerang suatu jaringan.

1. System Effectiveness

Tantangan untuk meningkatkan keefektivan suatu sistem adalah dengan mengembangkan suatu sistem yang dapat mendeteksi mendekati sempurna serangan dengan melakukan minimal kesalahan positif. Hal inilah yang masih perlu dikembangkan dalam pengembangan sistem.

Pada sekarang ini suatu *intrusion detection system* menggunakan model misuse detection. Contohnya dari IDS ini adalah **Snort** dan **RealSecure** yang menggunakan *signature* untuk menganalisis *network traffic.* Karena pada model ini sistem hanya mengetahui jenis serangan, pengembang harus secara periodik meng-*update signature* tersebut. Pendekatan ini sangat tidak efektif. Hal yang ideal adalah dengan menggunakan kemampuan yang dimiliki model anomaly detection yang secara otomatis akan mendeteksi serangan baru, namun tanpa memiliki tingkat kesalahan positif yang tinggi. Para analis kemudian mengembangkan suatu sistem yang menggunakan teknik *hybrid*(anomaly-misuse detection), namun harus disertai dengan investigasi dan analisis lebih lanjut.

2. Performance

Suatu sistem yang hanya melakukan pendeteksian untuk berbagai macam serangan adalah tidak cukup. Suatu IDS harus juga memperhatikan *input-event stream* yang dihasikan pada suatu *high-speed networks* dan *high performance network nodes,* yang paling banyak digunakan antara lain adalah gigabit ethernet dan fiber optic. *Network node* juga kian menjadi semakin cepat, memproses data dengan lebih banyak dan juga menghasilkan audit log yang semakin banyak pula. Hal-hal tersebut dapat menghasilkan masalah untuk *system administrator* dalam mengolah dan mengatur data-data yang dihasilkan. Ada dua cara yang digunakan untuk menganalisa jumlah data yang dihasilkan dengan real-time: membagi aliran-aliran *event* atau menggunakan *peripheral network sensor.*

Dalam pendekatan pertama, komponen yang disebut "*slicer*" akan membagi aliran *event* menjadi lebih kecil, aliran yang membuat IDS sensor dapat lebih mudah mengaturnya sehingga dapat dianalisa dengan real-time. Untuk melakukannya, seluruh aliran *event* harus dapat diakses pada satu lokasi saja. Untuk itu, para analis lebih menyarankan sistem ini digunakan pada sistem yang bertipe *centralized* atau pada *network gateways*

Masalah yang dihadapi dengan pendekatan ini adalah "*slicer*" harus dapat membagi aliran *event* dengan asumsi dapat mendeteksi semua skenarion serangan yang mungkin terjadi. Jika aliran event dibagi secara acak, sensor mungkin tidak dapat menerima data yang cukup dalam mendeteksi intrusion, karena bagian manifestasi serangan yang berbeda dapat menunjukkan "*slice*" yang berbeda.

Pendekatan kedua adalah menyebarkan/mengimplementasikan *multiple sensor* pada *network periphery*, dekat dengan *host/system* yang akan dilindungi. Pendekatan ini menggunakan asumsi bahwa dengan memindahkan analisis pada *network periphery*, pembagian yang *natural* pada *traffic* dapat terjadi.

Masalah dengan menggunakan pendekatan seperti ini adalah sulit untuk mengimplementasikan dan mengatur sensor dalam jumlah banyak. Pertama, posisi sensor yang benar sulit untuk didapatkan. Serangan yang berdasarkan topologi jaringan tertentu sepert serangan berdasarkan *routing* dan *spoofing*, membutuhkan sensor pendeteksi yang dapat ditempatkan di tempat yang lebih spesifik. Kedua adalah mengenai isu kontrol dan koordinasi. Network merupakan *entity* yang terus-menerus berubah dengan ancaman yang juga berubah. Serangan baru terus muncul setiap hari, oleh karena itu infrastruktur *sensing*-nya pun harus juga berevolusi.

3. Network-Wide Analisis

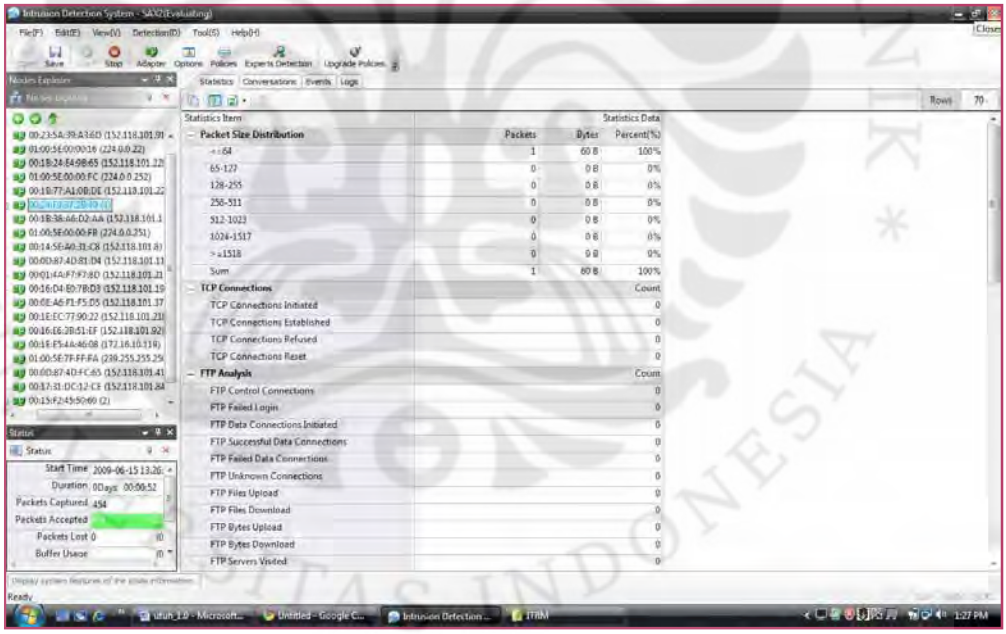
Menempatkan sensor pada tempat yang kritikal membantu *administrator* untuk dapat mendeteksi serangan yang mengancam suatu jaringan secara keseluruhan. Untuk itu, *sensing network* dapat menyediakan *big picture view* untuk suatu status keamanan suatu jaringan. Suatu bentuk serangan yang

mungkin muncul tidak relevan pada suatu *host* kemungkinan dapat membahayakan keseluruhan jaringan.

Sebagai contoh, suatu serangan yang meliputi beberapa langkah. Anggap saja setiap langkah tersebut terjadi pada host yang berbeda, namun karena jaringan yang ada menggunakan *shared file system*, efek yang ditimbulkan dapat mengancam keseluruhan sistem jaringan. Sistem mungkin tidak dapat mengidentifikasi langkah-langkah tersendiri pada setiap *host* dari sebuah sensor di sekitar *host* itu sebagai ancaman, sehingga masih dibutuhkan suatu sistem pengenalan berdasarkan informasi gabungan yang dimiliki semua sensor dalam suatu sistem jaringan sehingga hal tersebut dapat dihindari.

2.2.2 SAX2- Network Based Intrusion Detection System

IDS yang akan digunakan untuk mendeteksi ancaman-ancaman pada skripsi ini adalah SAX2. Gambar 2.9 adalah menu utama dari SAX2.

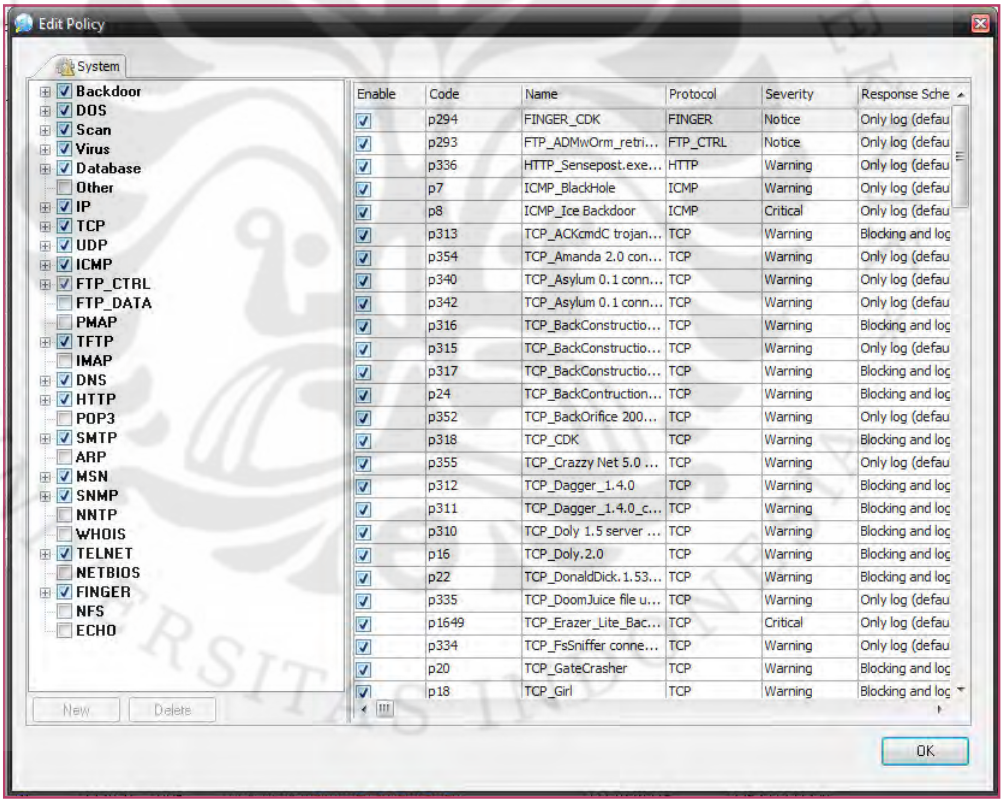


Gambar 2.9 Tampilan Menu Utama SAX2

Pada kiri atas gambar merupakan tampilan *nodes/hosts* apa saja yang berada dalam satu LAN yang sama dengan komputer dimana SAX2 di-*install*. Selanjutnya ada 4 bagian menu mengenai hasil monitoring SAX2, yaitu Statistics, Conversations, Events dan Logs. Salah satu fitur penting pada SAX2 yang dapat

dimanfaatkan pada proses skripsi ini adalah kemampuannya untuk membuat suatu file berformat *.txt. File tersebut kemudian dapat diolah oleh *script* php pada aplikasi yang akan dibuat untuk skripsi yang akan mengolah file tersebut sehingga akan dihasilkan nilai IT Risk Management dari setiap *threat* yang muncul dan terdeteksi.

Untuk dapat mendeteksi ancaman yang muncul, IDS SAX2 memiliki sekumpulan aturan (*policy*) yang berisi bermacam-macam bentuk ancaman serta pengkategorian seberapa bahaya ancaman-ancaman tersebut. Pada IDS SAX2 pengkategorian ancamana-ancaman tersebut berada pada grup severity, yang terbagi atas 4 level, mulai dari *information, notice, warning* dan *critical*. Berikut ini adalah gambar bagian *policy* dari IDS SAX2.



Gambar 2.10 Tampilan Kumpulan Policy Pada IDS SAX2

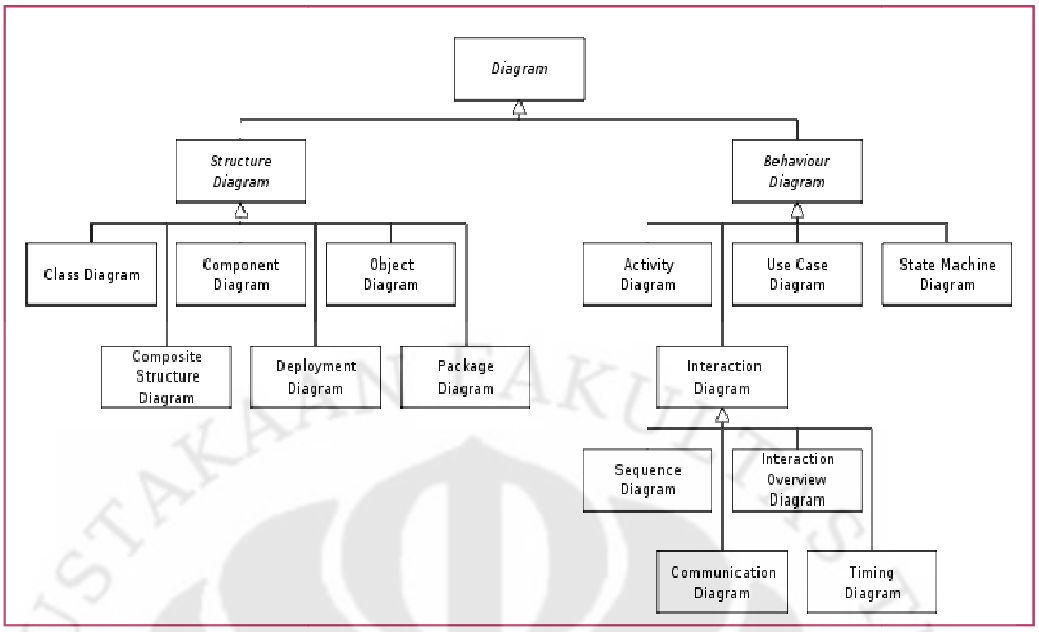
2.3 Unified Modeling Language

Unified Modeling Language (UML) adalah suatu bahasa yang digunakan dalam proses spesifikasi, visualisasi, konstruksi dan dokumentasi *artifact*_ suatu *software* yang sedang dibuat, namun juga dapat digunakan dalam proses permodelan bisnis (*business modeling*) serta dalam sistem *non-software*[3]. Mengembangkan model suatu sistem sangat berguna dalam pengembangan *software* itu sendiri sama seperti memiliki *blueprint* untuk sebuah bangunan. Model yang bagus sangat penting dalam komunikasi antar tim serta menjamin adanya kerjasama yang baik diantara mereka. Suatu sistem yang kompleks dimodelkan karena kita tidak dapat memahami keseluruhan sistem sekaligus. Ketika sistem tersebut menjadi semakin kompleks, maka memodelkan sistem merupakan hal yang sangat penting. Suatu bahasa permodelan yang baik harus menyertakan:

- Model elements : konsep permodelan dan semantics
- Notation : penggambaran visual elemen-elemen permodelan
- Guideline : istilah yang dipergunakan pada permodelan

Permodelan *software* juga diperlukan dalam dunia industri. Mereka memerlukan teknik untuk menangani kompleksitas suatu sistem yang akan terus meningkat. Pengembangan suatu *software* menjadi hal yang sangat penting sekarang ini. Perusahaan-perusahaan banyak yang telah menjadikan *software* sebagai aset yang memiliki nilai yang sangat penting sehingga mereka mencari teknik-teknik otomatisati produksi *software* yang dapat meningkatkan kualitas *software*, mengurangi biaya pembuatannya dan tepat waktu. UML berperan dalam menciptakan kumpulan semantic dan *notation* sehingga permasalah kompleksitas sistem dapat teratasi.

Permodelan dengan UML 2.0 (UML versi terbaru) terbagi dalam tiga kategori diagram, yaitu structure diagram, behavior diagram serta interaction diagram.



Gambar 2.11 Pengekategorian Diagram-Diagram Pada UML 2.0[3]

2.3.1 Structure Diagram

Structure diagram menekankan pada hal-hal apa saja yang harus ada dari sistem yang dimodelkan. Structure diagram terdiri atas diagram-diagram berikut ini[3]:

- Class diagram: menggambarkan struktur suatu sistem dengan manampilkan *class system*, atribut-atribut yang dimiliki serta hubungan antar class
- Component diagram: menggambarkan bagaimana suatu sistem *software* dibagi/dipecah menjadi komponen-komponen dan komponen-komponen tersebut ditampilkan keterhubungannya satu sama lain
- Composite structure diagam: menjelaskan bagaimana struktur internal suatu *class* serta kolaborasi antar struktur tersebut
- Deployment diagram: membantu untuk memodelkan *hardware* yang digunakan serta eksekusi dari *hardware* tersebut
- Object diagram: menampilkan tinjauan yang sempurna atau beberapa dari sistem yang dimodelkan pada saat tertentu

- Package diagram: menggambarkan bagaimana suatu sistem dibagi menjadi grup-grup logikal dengan menunjukkan keterhubungan antar grup-grup tersebut

2.3.2 Behaviour Diagram

Behavior diagram menekankan pada apa yang harus terjadi dari sistem yang dimodelkan, terdiri atas diagram-diagram berikut ini[3]:

- Activity diagram: merepresentasikan langkah demi langkah dari aliran komponen-komponen. Activity diagram menunjukkan keseluruhan aliran sistem yang terjadi
- State machine diagram: menunjukkan standarisasi notasi yang digunakan untuk menjelaskan sistem-sistem yang ada
- Use case diagram: menunjukkan fungsi-fungsi sistem dari sudut pandang actor, tujuan yang diharapkan oleh actor direpresentasikan oleh use case diagram.

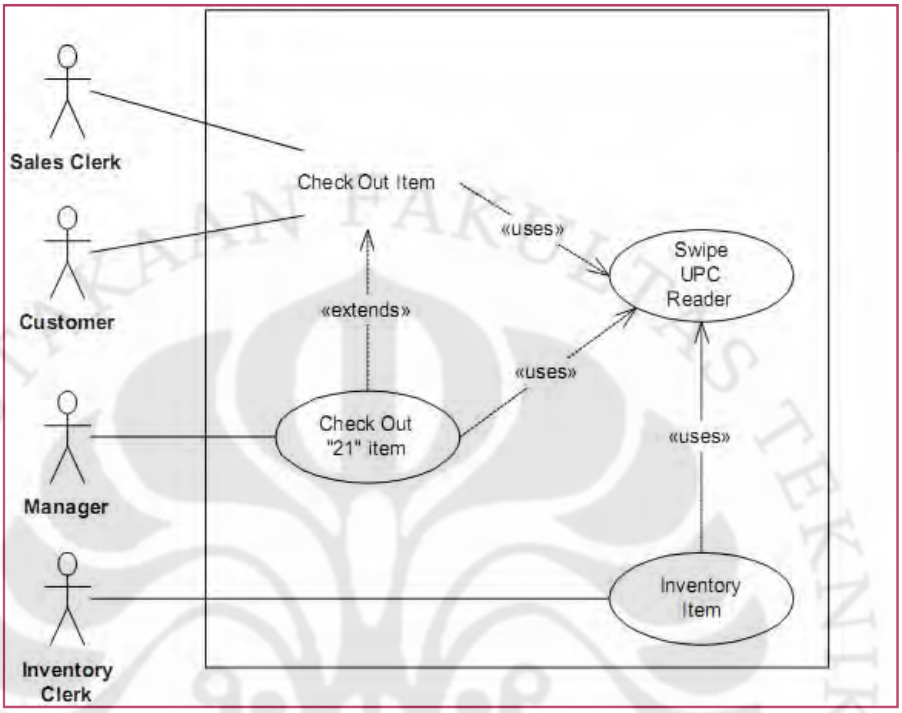
2.3.3 Interaction Diagram

Interaction diagram, merupakan bagian dari behavior diagram, menekankan pada kontrol aliran dan data antara hal-hal yang ada dari sistem yang dimodelkan. Interaction diagram terdiri atas diagram-diagram berikut ini[3]:

- Communication diagram: menunjukkan interaksi antar objek atau bagian dalam waktu yang berurutan. Ia merupakan kombinasi dari informasi yang diperoleh dari class, sequence serta use case diagram yang menggambarkan struktur statis serta kelakukan dinamis dari sistem
- Interaction overview diagram: merupakan tipe activity diagram yang setiap *node* merepreresentasikan interaction diagram
- Sequence diagram: menunjukkan bagaimana setiap objek saling berkomunikasi dalam waktu yang berurutan, juga menggambarkan lamanya waktu dari masing-masing komunikasi tersebut
- Timing diagram : merupakan tipe yang lebih spesisfik dari interaction diagram, lebih fokus pada kendala pewaktuan

2.3.4 Beberapa Contoh Diagram-Diagram UML: Class Diagram, Use Case Diagram, Sequence Diagram

- Use Case Diagram



Gambar 2.12 Contoh Use Case Diagram[3]

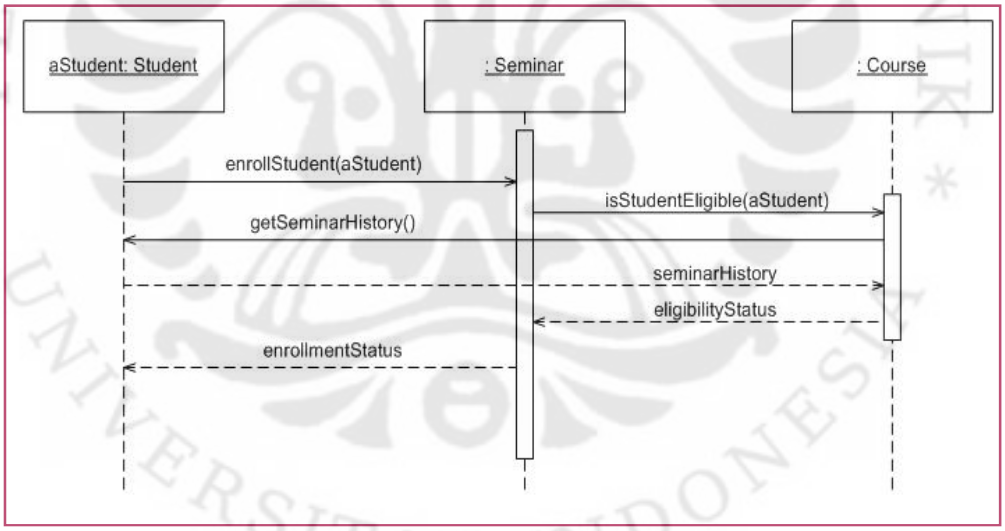
Gambar 2.12 Merupakan contoh dari use case diagram mengenai sistem penjualan suatu toko. Suatu use case diagram terbagi atas dua bagian, yaitu internal dan external. Internal berarti keseluruhan bagian (pada gambar diatas yaitu bagian didalam kotak) tersebut merupakan bagian dari sistem penjualan yang terdiri atas beberapa macam use case (digambarkan berbentuk oval) sedangkan actor (*sales, customer, manager* dan *inventory*) termasuk bagian external dan terpisah dari system penjualan toko. Pada gambar tersebut berarti *Customer* membeli suatu barang dari toko kemudian *Sales* yang menerima barang dari *Customer* mencocokkan barang tersebut dengan menggunakan (uses) UPC Reader untuk memeriksa harga barang tersebut. Uses digunakan karena dalam menggunakan suatu use case (check out item) terdapat keterikatan dengan use case lainnya, yaitu Swipe UPC Reader dalam hal ini dapat disebut juga jika uses merupakan suatu *function call* atau *routine*. Pada bagian lainnya *Actor Inventory*

Clerk mengecek *inventory item* kemudian ia memerlukan bantuan dari use case swipe UPC Reader. Sedangkan *relation* extends pada gambar tersebut dapat diartikan sebagai berikut:

Pada suatu kasus ada kasus khusus ketika seorang *sales* yang akan mengecek suatu item harus menunggu persetujuan *manager* sehingga transaksi tersebut dapat disetujui. Dalam hal ini *sales* akan meminta bantuan *manager* untuk membuka transaksi (ditandai dengan relation extends) untuk item tersebut sehingga transaksi dapat dilanjutkan.

- Sequence Diagram

Sequence diagram menggambarkan suatu bentuk interaksi dalam suatu kumpulan pesan-pesan (messages) antara ClassifierRoles dalam suatu Collaboration. Gambar 2.13 merupakan contoh sequence diagram yang menggambarkan proses pendaftaran (enrollment) mahasiswa dalam seminar.



Gambar 2.13 Contoh Sequence Diagram[3]

Kotak-kotak yang terletak di bagian atas gambar 2.13, yaitu aStudent: Student, : Seminar dan : Course, merupakan ClassifierRoles yang biasanya adalah use case, object, class atau actor. Garis putus-putus yang terletak dibawah kotak disebut object lifeline, merepresentasikan umur/durasi dari object dalam skenario

yang dimodelkan. Sedangkan kotak kecil dan panjang disebut activationbox menjelaskan proses yang sedang dilakukan oleh object/class tujuan dalam menyelesaikan suatu message. Proses pendaftaran mahasiswa untuk mendaftar dalam suatu seminar tersebut dimulai dengan mahasiswa yang masuk ke suatu sistem status seminar, contohnya dalam hal ini adalah sistem seminar berbasis web. Mahasiswa masuk ke *website* kemudian melakukan proses pendaftaran. Sistem kemudian melakukan validasi apakah mahasiswa tersebut dapat melakukan pendaftaran seminar. Validasi dilakukan dengan melihat *track record* seminar yang dilakukan dan melihat status akademis mahasiswa tersebut. Apabila mahasiswa tersebut memiliki status dapat melakukan pendaftaran seminar, maka sistem akan menampilkan status kepada mahasiswa tersebut bahwa ia telah terdaftar dalam seminar.

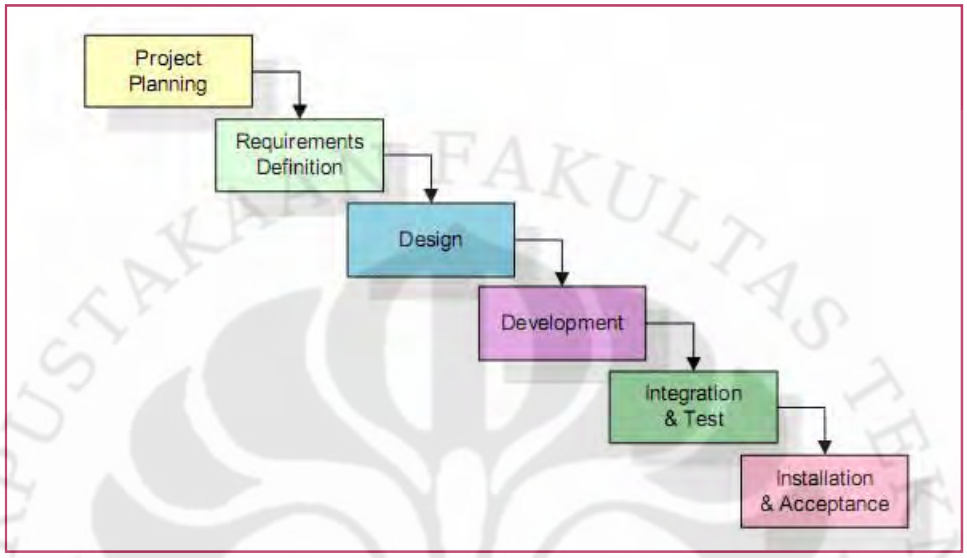
BAB 3

PERANCANGAN APLIKASI BERBASIS WEB PERHITUNGAN IT RISK LEVEL

Setelah beberapa dasar pengetahuan mengenai IT Risk Management yang memiliki beberapa tahapan, mulai dari tahapan *risk assessment*, *risk mitigation* serta *assessment* dan *evaluation* dilanjutkan dengan penjelasan mengenai Intrusion Detection System, penulis akan membuat suatu sistem berbasis web yang akan menghitung IT Risk Level dari suatu topologi LAN, menggunakan SAX2 IDS yang menghasilkan log-log dari suatu keadaan jaringan yang dimonitor, kemudian akan diolah log-log tersebut dan setelah melalui perhitungan dengan standar tertentu maka akan dihasilkan suatu nilai *risk level* untuk suatu ancaman maupun nilai *risk level* keseluruhan suatu topologi jaringan. Web tersebut bernama "**Web-Based Intrusion Detection and Network Risk Monitoring**". Mendapatkan nilai *risk level* merupakan salah satu tahapan dari *risk assessment*. Ketika suatu topologi jaringan telah diperoleh nilai *risk level*-nya, maka nilai-nilai tersebut dapat membantu *administrator* jaringan maupun pihak yang bertanggung jawab pada suatu perusahaan/organisasi dalam menentukan kontrol yang akan digunakan selanjutnya oleh perusahaan/organisasi tersebut, apakah mereka akan mempertahankan kontrol sebelumnya atau akan menggunakan kontrol baru, sehingga ancaman-ancaman yang ada dapat dihilangkan atau pun ancaman-ancaman baru yang mungkin muncul dapat dihindari. Sistem yang dibuat selanjutnya akan diuji coba pada suatu jaringan, hal ini dilakukan untuk melihat kinerja dari sistem tersebut .

Untuk mendapatkan suatu aplikasi yang tepat serta terstruktur dengan baik, maka dalam pengembangan pembuatan aplikasi ini, penulis membangunnya berdasarkan Software Development Life Cycle (SDLC). SDLC merupakan proses berkelanjutan yang digunakan oleh sistem analis suatu perusahaan untuk mengembangkan sistem informasi, meliputi tahap-tahap project planning, requirements definition, design, development, integration & test dan installation & acceptance. Sebuah perancangan SDLC yang baik dapat menghasilkan suatu

sistem berkualitas tinggi sehingga dapat memenuhi keinginan *customer* atau juga dapat membantu orang-orang/karyawan dalam suatu sistem perusahaan bekerja secara efektif dan efisien dalam waktu tertentu dan menghasilkan yang diharapkan.



Gambar 3.1 Tahapan System Development Life Cycle[4]

3.1 Project Planning

Pada tahap ini dibuat mengenai latar belakang pembuatan aplikasi serta tujuan yang diharapkan setelah sistem web tersebut dibuat. Tujuan utama dari pembuatan aplikasi ini adalah untuk mendukung dalam implementasi IT Risk Management suatu perusahaan/organisasi. Pada IT Risk Management aspek yang dilihat sangat luas, mulai dari aspek fisik seperti pengkabelan, aspek jaringan seperti topologi jaringan, pemberian IP Address sampai aspek keamanan jaringan. Aplikasi ini kemudian dibuat untuk membantu implementasi ITRM dari aspek keamanan jaringan. Aspek keamanan jaringan merupakan salah satu hal yang perlu ditinjau secara detail. Hal ini disebabkan data-data suatu perusahaan/organisasi banyak yang disimpan berbentuk data dijital, seperti data karyawan, data transaksi operasional serta data keuangan. Proteksi terutama diperlukan untuk melindungi data-data tersebut dari pencurian serta perusakan data oleh pihak internal

perusahaan sendiri dan juga kemungkinan terjadinya kesalahan prosedur dari pihak internal perusahaan.

Tujuan yang diharapkan dari pembuatan aplikasi ini adalah adanya laporan-laporan mengenai keadaan jaringan LAN perusahaan/ organisasi. Dari laporan-laporan tersebut pihak perusahaan/organisasi dapat memutuskan apakah kontrol internal yang dimiliki telah mampu mencegah suatu *threat*/ancaman menggangu kinerja perusahaan/organisasi, atau dibutuhkan kontrol baru sehingga ancaman yang telah muncul dapat dicegah. Selain itu, dari laporan-laporan tersebut perusahaan dapat mencegah ancaman yang ada bertindak lebih jauh sehingga dapat membahayakan perusahan/organisasi

3.2 Requirements Definition

Pada tahap ini dilakukan proses identifikasi mengenai kebutuhan-kebutuhan apa saja yang harus disiapkan sehingga tujuan pembuatan aplikasi dapat dipenuhi. Kebutuhan-kebutuhan tersebut meliputi :

3.2.1 Identifikasi dan klasifikasi aset perusahaan/organisasi

Aset merupakan segala sesuatu yang bernilai bagi perusahaan/organisasi. Pada tahap ini dilakukan proses identifikasi dan klasifikasi dari semua aset yang dimiliki oleh perusahaan/organisasi yang memiki hubungan dengan aplikasi yang dibuat, yaitu aset-aset apa saja yang terhubung dengan jaringan. Pada umumnya aset-aset yang termasuk adalah komputer, laptop atau *server*. Selanjutnya setelah semua aset diidentifikasi, diperlukan pengklasifikasian terhadap aset-aset tersebut menjadi kedalam kelas-kelas aset. Aset-aset yang tersebut dimasukkan kedalam grup-grup tertentu yang masing-masing grup tersebut merepresentasikan dampak yang muncul terhadap perusahaan/organisasi apabila aset-aset tersebut diserang. Selain itu, dengan adanya grup-grup tertentu, memudahkan pihak perusahaan/organiasi memproritaskan langkah terlebih dahulu terhadap aset yang lebih penting.

Microsoft melalui Microsoft Security Risk Management Process [5] memiliki standar untuk klasifikasi dari aset-aset perusahaan berdasarkan dampaknya terhadap perusahaan, yaitu:

- High Business Impact : Aset-aset yang digolongkan pada grup ini, apabila terjadi masalah atau hilangnya *confidentiality, integrity* atau *availability* pada aset-aset tersebut, maka akan berdampak besar atau sangat berbahaya bagi perusahaan. Dampak yang ditimbulkan dapat berupa kerugian bagi keuangan perusahaan seperti data penting yang dicuri atau juga dampak tidak langsung, seperti rusaknya reputasi perusahaan serta terganggunnya produktifitas perusahaan
- Moderate Business Impact : Aset-aset yang digolongkan pada grup ini, apabila terjadi masalah atau hilangnya *confidentiality, integrity* atau *availability* pada aset-aset tersebut, maka menimbulkan dampak kerugian pada perusahaan, walaupun tidak terlalu besar, namun akan menggangu kinerja beberapa fungsi-fungsi organisasi perusahaan.
- Low Business Impact : Aset-aset pada grup ini apabila terganggu/ bermasalah, maka tidak berpengaruh signifikan pada kinerja operasional/finansial perusahaan.

Untuk kepentingan impelementasi, business impact yang sebelumnya dikelompokkan secara kualitatif kemudian dikelompokkan secara kuantitatif sehingga setiap business impact bersesuaian dengan nilai-nilai tertentu, yaitu:

- High business impact bersesuaian dengan impact class bernilai 10
- Medium business impact bersesuaian dengan impact class bernilai 5
- Low business impact bersesuaian dengan impact class bernilai 2

3.2.2 Menentukan Asset Exposure

Sebelum *Asset Exposure* ditentukan, maka diperlukan pendefinisian terlebih dahulu terhadap *threat* dan *vulnerability*. *Vulnerability* dan *threat* merupakan hal yang saling berhubungan, *threat*/ancaman yang ditimbulkan oleh suatu sumber ancaman (*threat source)* akan menyerang suatu kelemahan (*vulnerability*) pada suatu perusahaan. Informasi mengenai *threat* dan *vulnerability* diperoleh dari hasil wawancara dengan *stakeholder* perusahaan/organisasi mengenai aset-aset apa saja yang harus dijaga serta kerugian apa saja yang perlu dihindari terhadap aset-aset tersebut. Setelah kedua data tersebut diperoleh hal selanjutnya yang dilakukan

adalah memperkirakan tingkat kerusakan potensial pada aset ketika suatu *threat* menyerang *vulnerability*, tingkat kerusakan tersebut disebut *asset exposure.*

Asset Exposure memiliki tiga tingkatan, yaitu :

- High exposure : menyebabkan kerusakan yang sangat besar pada aset
- Moderate exposure : menyebabkan kerusakan pada aset, namun tidak terlalu besar
- Low exposure : kerusakan yang ditimbulkan kecil atau tidak rusak sama sekali

Selanjutnya untuk kepentingan implementasi asset exposure diubah menjadi hitungan kuantitatif menjadi 5 tingkat dan disebut exposure rating, yaitu:

- Exposure Rating 5: menyebabkan kerusakan total pada aset, seperti kerusakan mempengaruhi pendapatan atau kesuksesan yang akan didapatkan perusahaan dan dapat dilihat oleh pihak luar/masyarakat. Memiliki nilai perhitungan 1.
- Exposure Rating 4 : menyebabkan kerusakan yang serius namun tidak menyebabkan kerusakan total pada aset, seperti mempengaruhi pendapatan/kesuksesan yang akan didapatkan perusahaan/organisasi dan kemungkinan dapat dilihat pihak luar/masyarakat. Memiliki nilai perhitungan 0,8.
- Exposure Rating 3 : menyebabkan kerusakan sedang pada perusahaan/organisasi, mempengaruhi kinerja internal perusahaan sehingga menyebabkan peningkatan biaya operasional atau mengurangi pendapatan. Memiliki nilai perhitungan 0,6.
- Exposure Rating 2 : menyebabkan kerusakan rendah, mempengaruhi kinerja internal perusahaan. Memiliki nilai perhitungan 0,4.
- Exposure Rating 1 : menyebabkan kerusakan yang sangat kecil atau bahkan tidak ada kerugian yang timbul. Memiliki nilai perhitungan 0,2.

Asset exposure pada program yang dibuat diperoleh dari program IDS SAX2. Untuk setiap kejadian *threat* yang muncul dan terdeteksi oleh IDS SAX2, maka program tersebut akan menghasilkan respon yang terdiri atas :

1. Waktu terjadinya gangguan
2. *Source* serta *destination IP address* tiap-tiap gangguan

3. *Source* serta *destination Source port* tiap-tiap gangguan
4. *Protocol* yang berhubungan dengan *threat* yang muncul
5. Informasi mengenai *threat* yang muncul
6. *Severity* atau level seberapa bahayakah *threat* tersebut

Severity pada IDS SAX2 terbagi atas empat tingkatan, mulai dari dari yang paling rendah yaitu information, notice, warning dan critical. Asset exposure rating kemudian diperoleh bersesuaian dengan severity level dari IDS SAX2, dengan catatan asset exposure rating 1 tidak disetakan sehingga akan diperoleh hal berikut ini:

- *Exposure Rating 2* bersesuaian dengan *severity level information*
- *Exposure Rating 3* bersesuaian dengan *severity level notice*
- *Exposure Rating 4* berseuaian dengan *severity level warning*
- *Exposure Rating 5* berseuaian dengan *severity level critical*

3.2.3 Memperkirakan Kemungkinan Munculnya *Threat*

Setelah memperkirakan dampak potensial yang muncul untuk suatu *threat* terhadap perusahaan/organisai, langkah selanjutnya adalah memperkirakan berapa kali kemungkinan munculnya *threat* tersebut. Biasanya jumlah tersebut diketahui berdasarkan pendapat dari *stakeholder* perusahaan/organisasi dan dapat pula diperoleh dari pengalaman perusahaan/organisasi di masa lalu.

Level jumlah kemungkinan muncul *threat* memiliki tiga level, yaitu:

- High. *Threat* diharapkan muncul setidaknya satu kali dalam satu tahun
- Medium. *Threat* diharapkan muncul setidaknya satu kali dalam dua/tiga tahun
- Low. *Threat* muncul setidaknya satu kali dalam tiga tahun

Selanjutnya standar tersebut disesuaikan untuk perhitungan kuantitatif sehingga setiap level diwakili oleh range tertentu:

- High : 10-7 kali muncul *threat* dalam satu tahun
- Medium: 4-6 kali muncul *threat* dalam satu tahun
- Low : 0-3 kali muncul *threat* dalam satu tahun

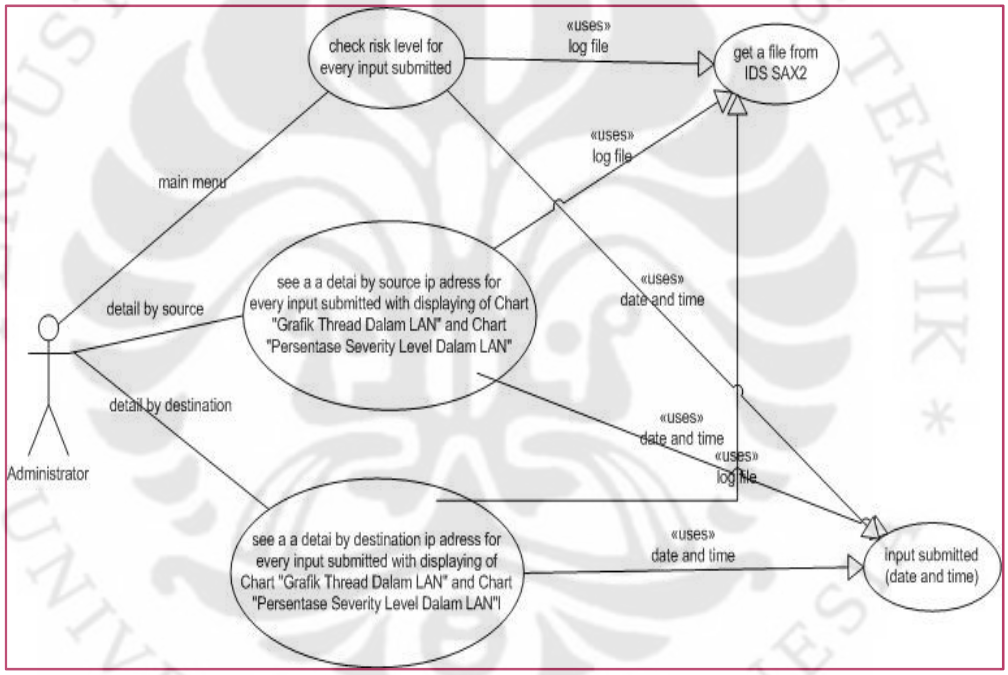
3.3 Design

Pada tahap ini, dilakukan penggambaran mengenai fitur-fitur dari software/sistem yang akan dibuat, meliputi diagram dari fungsi-fungsi yang diharapkan muncul serta diagram mengenai cara kerja pada fungsi internal. Keseluruhan hal tersebut dibutuhkan agar adanya penggambaran software/sistem dengan detail yang mencukupi sehingga programmer dapat mengembangkan software tersebut dengan input tambahan yang minimal. Dalam penulisan skripsi ini, penulis menggunakan bantuan UML diagram untuk menggambarkan rancangan sistem yang dibutuhkan pada tahapan design ini. Pembuatan UML diagram tersebut dilakukan menggunkan software Visual Paradigm versi 7.0. Penulis menggunakan 2 buah UML diagram, yaitu UML Use Case Diagram dan UML Sequence Diagram. Alasan mengapa ketiga UML Diagram tersebut digunakan adalah karena keduanya telah mewakili pengkategorian UML diagram, yaitu Use Case Diagram untuk Behaviour Diagram serta Sequence Diagram untuk Interaction Diagram.

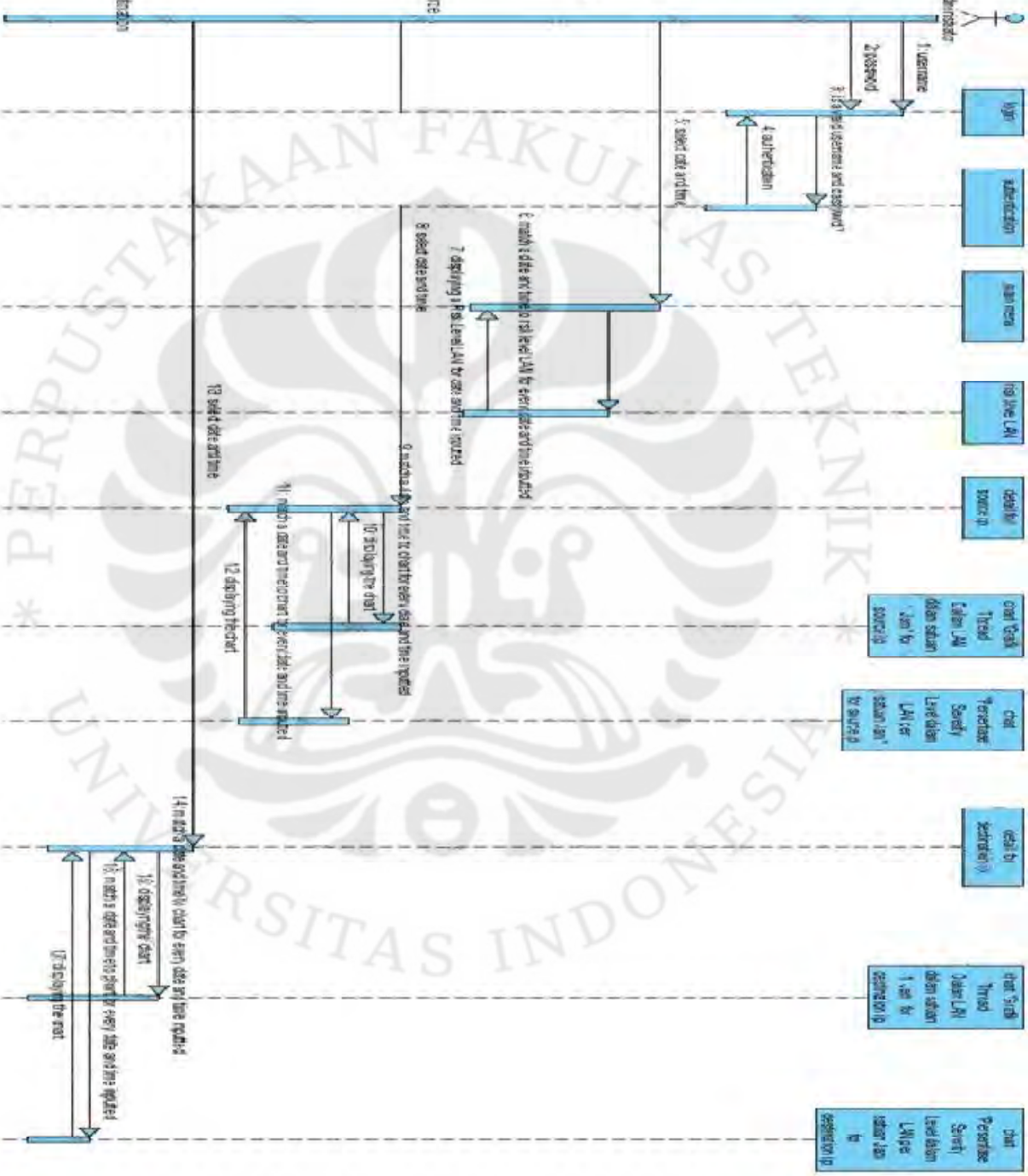
3.3.1 Use Case Diagram

Use Case Diagram digunakan untuk menjelaskan hal-hal apa saja yang seharusnya terjadi pada sistem yang dimodelkan, dipandang dari sudut pandang *actor* dan menjelaskan fungsi-fungsi dari sistem serta tujuan yang ingin dicapai oleh *actor* tersebut. *Actor* dari *software* yang dibuat adalah seorang *administrator* jaringan yang berkewajiban untuk melakukan pengawasan terhadap LAN suatu perusahaan/organisasi, yaitu setelah dia melihat hasil dari *software* tersebut dia bertugas untuk mengatasi *threat* yang muncul apabila *threat* tersebut telah dideteksi oleh sistem, selain itu ia juga berkewajiban membuat suatu laporan mengenai keadaan risk level dari jaringan tersebut. Pada saat mengakses *website*, *administrator* masuk menuju main menu, dimana menu ini digunakan untuk melihat nilai risk level dari setiap input yang dimasukkan oleh *administrator*. Input yang dimasukkan meliputi tanggal, bulan, tahun, serta jam yang akan diinspeksi. Sebagai contoh ketika input 1-January-2009-00:00 dimasukkan, maka website akan menghitung nilai risk level pada tanggal 1 Januari 2009 mulai dari jam 00:00 sampai jam 01:00. Dua menu selanjutnya yaitu detail by source ip serta

detail by destination ip. Keduanya sama-sama menampilkan dua buah grafik, yaitu Grafik *Threat* Dalam LAN serta Grafik Persentase Severity Level dalam LAN, perbedaannya hanya terletak dari segi inspeksi IP Address yang dilakukan, yaitu inspeksi ip address sumber dari suatu *threat* untuk detail by source ip dan inspeksi ip address tujuan dari suatu *threat* untuk detail by destination ip. Semua data-data yang berada pada setiap menu website memiliki kesamaan dalam hal dari mana data-data yang digunakan pada website tersebut berasal, yaitu dari *log file* yang dihasilkan IDS SAX2



Gambar 3.2 Use Case Diagram "Web-Based Intrusion Detection and Network Risk Monitoring"



Gambar 3.3 Sequence Diagram “Web-Based Intrusion Detection and Network Risk Monitoring”

3.3.2 Sequence Diagram

Sequence Diagram digunakan untuk menjelaskan aliran data yang terjadi pada sistem yang dimodelkan, yaitu dari segi bagaimana setiap objek yang ada saling berkomunikasi dalam waktu yang berurutan. Gambar 3.3 merupakan sequence diagram dari model yang akan dibuat. Sudut pandang dari diagram tersebut memperlihatkan administrator yang akan mengakses website "Web-Based Intrusion Detection and Network Monitoring". Aliran yang terjadi disesuaikan dengan kegunaan dari website tersebut, yaitu sebagai *tools* yang diperlukan oleh manajemen dalam melihat implementasi dari security control yang telah diterapkan oleh perusahaan/organisasi. Oleh karena itu, administrator berkewajiban untuk melaporkan *output* apa saja yang dihasilkan oleh website tersebut kepada pihak manajemen.

Berikut ini merupakan langkah-langkah yang dilakukan oleh administrator tersebut:

1. Administrator *login* ke website: langkah ini merupakan langkah awal yang dilakukan ketika administrator ingin masuk ke website tersebut, ia harus melakukan autentikasi terlebih dahulu dengan memasukkan username dan password. Hal ini dilakukan untuk mencegah pihak-pihak yang tidak berhak untuk masuk ke website tersebut, yang mana didalam website terdapat kerahasiaan perusahaan menyangkut level security jaringan yang dimiliki perusahaan.
2. Administrator masuk ke menu utama: setelah administrator berhasil melakukan autentikasi, maka ia langsung masuk ke menu utama. Di menu ini ia harus memasukkan input berupa tanggal dan waktu kemudian akan ditampilkan Risk Level LAN dari tanggal dan waktu tersebut
3. Administrator masuk ke menu detail for destination ip: hal yang terjadi pada menu ini sama seperti pada menu detail for source ip hanya saja ada sedikit perbedaan dari segi inspeksi IP Address yang dilakukan, yaitu pada menu ini inspeksi dilakukan untuk IP Address tujuan.
4. Setelah semua proses diatas dilakukan maka administrator berkewajiban melaporkan hasil-hasil yang telah diperoleh, yaitu hasil tiap jam untuk Risk Level LAN serta hasil dari Grafik *Threat* dalam LAN dan Grafik

Persentase Severity Level dalam LAN untuk masing-masing source ip dan destination ip.

3.4 Implementation

Bagian ini menjelaskan bagaimana permodelan yang telah dilakukan pada bagian design kemudian diimplementasikan kedalam file-file bahasa pemrograman. Dalam pembuatan website "Web-Based Intrusion Detection and Network Risk Monitoring" ini, dipergunakan tiga bahasa pemrograman, yaitu HTML, PHP dan SQL. HTML digunakan dalam pembuatan fungsi-fungsi dasar web, seperti GUI tingkat dasar, PHP digunakan untuk memproses *input* dari *user* serta mengolahnya untuk kemudian dihasilkan *output* yang diinginkan, SQL digunakan untuk memproses database, sedangkan untuk implementasinya website tersebut akan diletakkan pada server local. Ada empat bagian utama dari program yang akan dibuat, yaitu bagian menu login, menu utama, halaman detail for source dan halaman ip detail for destination ip. Berikut ini merupakan penjelasan yang lebih detail untuk masing-masing bagian tersebut:

3.4.1 Menu Login

Menu pertama dari website adalah *login*. *Login* diperlukan sebagai autentikasi sehingga hanya administrator saja atau orang-orang yang telah diberi hak oleh administrator yang dapat mengakses website tersebut. Administrator/*user* harus mengakses halaman login.php untuk dapat masuk ke menu login. Gambar 3.4 merupakan rancangan halaman menu login dari website yang dibuat.

Gambar 3.4 Rancangan Menu Login "Web-Based Intrusion Detection and Network Risk Monitoring"

User/administrator diharuskan terlebih dahulu memasukkan username dan password untuk kemudian dilakukan autentikasi yang dalam hal ini dilakukan oleh auth.php. Pada login.php terdapat peringatan bahwa halaman tersebut hanya boleh dimasuki oleh orang-orang yang memiliki hak atau diberi hak oleh administrator. Selanjutnya setelah user memasukkan username dan password maka username dan password tersebut akan dikirim ke halaman auth.php melalui melalui mekanisme post. Data-data tersebut kemudian diterima oleh variable $user untuk username serta variabel $pass untuk password dan selanjutnya dilakukan pencocokan dengan data yang terdapat pada database.

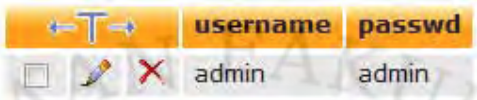
3.4.1.1 Langkah-Langkah Pemrosesan Data Pada Menu Login

Berikut ini adalah langkah-langkah yang dilakukan oleh auth.php untuk dapat mencocokkan data dari variable $user dan $pass dengan data yang terdapat dalam database skripsi_php:

1. Memanggil halaman connect_db.php. Halaman ini digunakan untuk membuka koneksi dengan database. Untuk dapat membuka koneksi dengan database diperlukan keterangan mengenai alamat dari situs yang

digunakan, username dan password database serta nama database yang akan digunakan

2. Mencocokkan variable $user dan $pass dengan data yang terletak pada database. Berikut ini merupakan gambar rancangan dari isi tabel login yang digunakan untuk mecocokkan variable-variable tersebut.

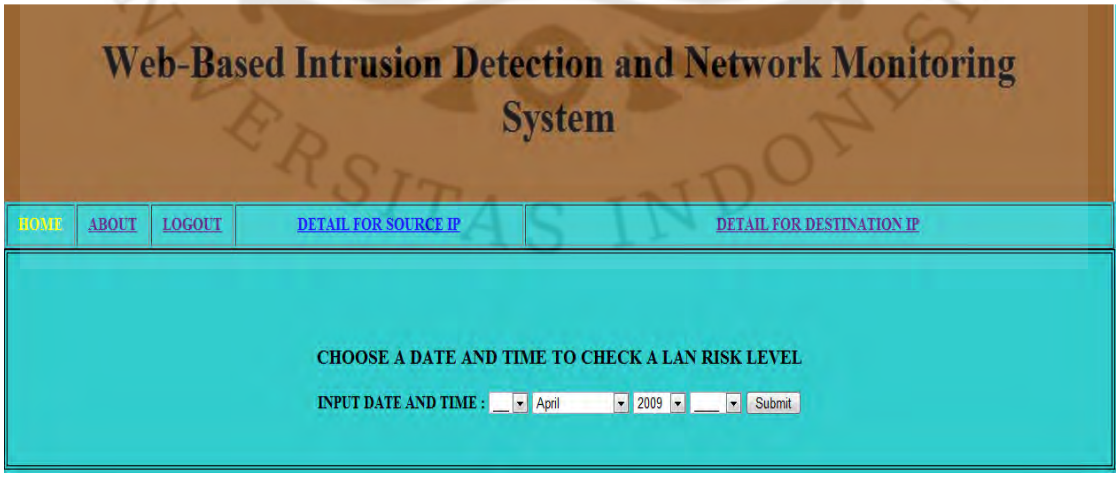


Gambar 3.5 Rancangan Tabel Login dari Database Skripsi_PHP

3. Apabila variable $user dan $pass yang dimasukkan sesuai dengan yang terdapat pada tabel login (dalam hal ini username=admin dan password=admin) maka administrator akan langsung diarahkan menuju halaman home.php namun apabila $user dan $pass tersebut salah maka akan muncul peringatan bahwa dia salah memasukkan username dan password kemudian akan diarahkan ulang ke halaman login.php

3.4.2 Menu utama

Setelah user berhasil memasukkan username dan password yang sesuai, maka user akan berada di menu utama, yaitu halaman home.php. Gambar 3.6 merupakan rancangan tampilan menu utama:



Gambar 3.6 Rancangan Menu Utama "Web-Based Intrusion Detection and Network Risk Monitoring"

Pada bagian menu utama ini user dapat melihat hasil perhitungan risk level dalam suatu LAN. Namun sebelumnya user diharuskan terlebih dahulu memasukkan tanggal serta jam dari data yang akan diinspeksi. Setelah dipilih maka akan ditampilkan risk level LAN sesuai dengan input yang dipilih.

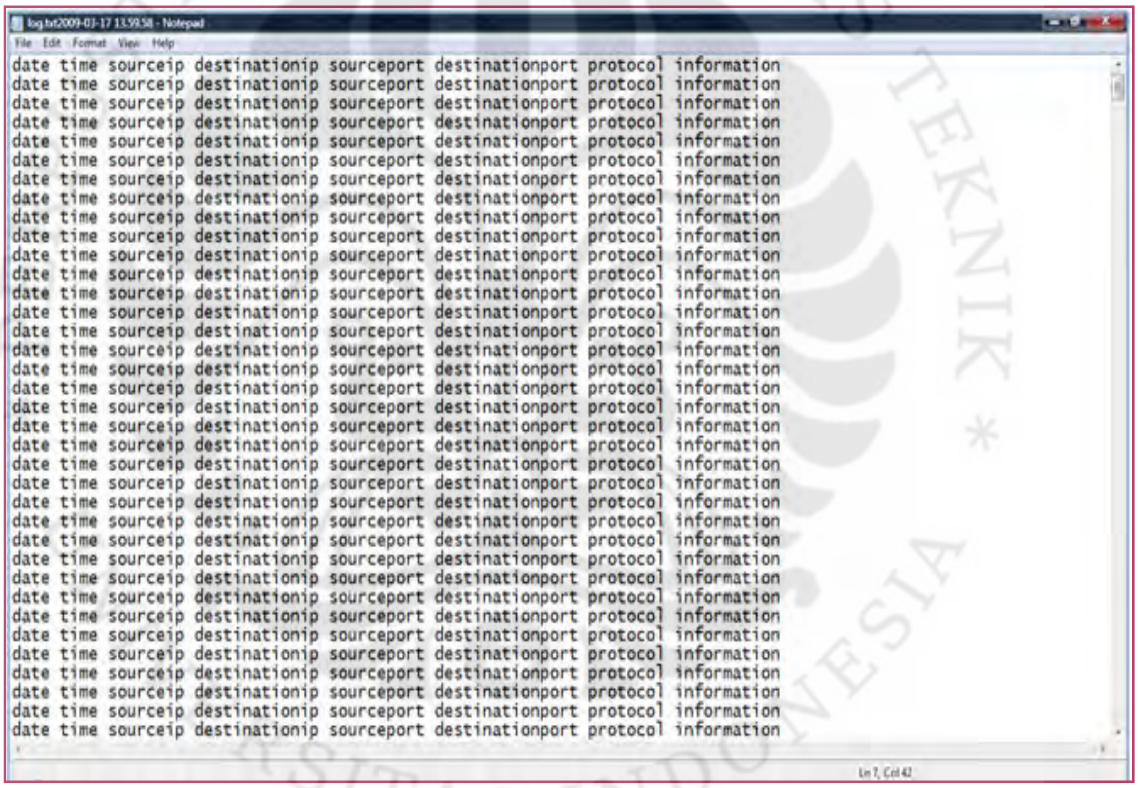
3.4.2.1. Langkah-Langkah Pemrosesan Data Pada Menu Utama

Berikut ini merupakan langkah-langkah yang terjadi pada home.php untuk dapat menghasilkan output sesuai input yang diinginkan:

1. Memanggil halaman datepicker.php. Halaman tersebut berfungsi untuk menampilkan *dropdown button*. Dropdown button berguna sebagai box-box pilihan sehingga *user* yang akan memilih tanggal serta jam tidak perlu menulisnya secara manual, selain merepotkan hal tersebut dapat menyebabkan kesalahan dalam memasukkan *input* karena user mungkin tidak mengetahui standar penulisan input-input tesebut. Dengan adanya *dropdown button* user memilih box-box input yang sesuai, dalam hal ini user harus memilih tiap-tiap box sehingga tidak boleh ada box yang kosong.
2. Setelah user memilih input dalam box dan menekan box submit, maka untuk input tanggal akan ditangkap oleh variable $date sedangkan input jam akan ditangkap oleh variable $hour.
3. Variabel $hour dan $date diperlukan agar data yang ditampilkan sesuai dengan *input* yang dimasukkan user
4. Script php kemudian akan membuka koneksi dengan database dengan memanggil halaman connect_db.php untuk kemudian masuk ke database skripsi_php.
5. Berikutnya script akan masuk ke line ini:

```
$fh = fopen("log.txt2009-03-17 13.59.58.log", "r");
$line = fgets($fh);
$data = explode(" ",$line,9);
```

Line tersebut digunakan untuk membaca file log.txt2009-03-17 13.59.58.log. File tersebut dihasilkan program SAX2 IDS yang berisi hasil pengawasan dari LAN dimana IDS tersebut dipasang. Setelah data dari baris file log dibaca maka data akan masuk ke variable $fh, kemudian melalui perintah explode tiap-tiap bagian data dari satu baris log akan masuk ke masing-masing anggota dari array data. Batas/pembeda untuk masing-masing array dibatasi dengan adanya spasi, namun karena selanjutnya terdapat limit dari jumlah array, maka hanya akan terdapat 9 buah masing array dari variable $data. Berikut ini merupakan rancangan dari output log yang dihasilkan oleh IDS SAX2.



Gambar 3.7 Rancangan Output Log IDS SAX2

6. Selanjutnya data masing-masing baris akan dimasukkan ke dalam database skripsi_php tabel coba melalui perintah:

$query = "INSERT INTO `coba` VALUES ('','$data[0]','$data[1]','$data[2]','$data[3]','$data[4]','$data[5]','$data[6]','$data[7]');

Gambar berikut ini merupakan isi dari tabel coba:

	id	date	time	source_ip	destination_ip	source_port	destination_port	protocol	information
	1	date	time	source_ip	destination_ip	source_por	destinatio	protocol	information

Gambar 3.8 Rancangan Isi Tabel Coba Database Skripsi_PHP

Terdapat kolom tambahan yaitu kolom id, digunakan sebagai penanda data yang masuk sebagai data yang keberapa, selain itu id juga digunakan untuk mencegah adanya masalah saat terdapat data yang sama persis dari kolom date sampai kolom information.

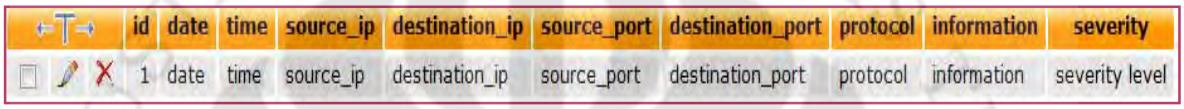
7. Dalam mengolah log dari SAX2 terdapat masalah pada perbedaan jumlah kolom yang dihasilkan log SAX2 dibandingkan dengan jumlah kolom yang ditampilkan pada program SAX2. Perbedaan terletak pada ketidakadaannya kolom severity pada log yang dihasilkan SAX2. Hal ini menimbulkan masalah karena dalam proses-proses selanjutnya kolom severity sangat dibutuhkan. Oleh karena itu dibutuhkan suatu mekanisme dengan bantuan tabel baru dapat dimungkinkan adanya penggabungan dua buah tabel database sehingga dapat menambah kolom-kolom tertentu. Hal tersebut dapat terjadi dengan dibuatnya suatu tabel yang bernama coba1 yang berisi kolom severity dan satu buah kolom lagi yang dapat menghubungkan tabel coba dan tabel coba1, dalam hal ini ditambahkan kolom information pada tabel coba1. Selanjutnya tabel coba1 akan berisi data-data berikut ini:

	severity	information
	severity level	information

Gambar 3.9 Rancangan Isi Tabel Coba1 Database Skripsi_PHP

Selanjutnya kedua tabel tersebut digabungkan dan akan dihasilkan suatu tabel utuh yang memiliki kolom severity melalui query berikut ini:

$query = "SELECT coba.id, coba.date, coba.time, coba.source_ip, coba.destination_ip, coba.source_port, coba.destination_port, coba.protocol, coba.information, coba1.severity FROM coba INNER JOIN coba1 ON coba.information=coba1.information where date='$date' && source_ip='$thisIp' && time between '$timeSelect' and '$timeEnd'";



id	date	time	source_ip	destination_ip	source_port	destination_port	protocol	information	severity
1	date	time	source_ip	destination_ip	source_port	destination_port	protocol	information	severity level

Gambar 3.10 Rancangan Isi Penggabungan Tabel Coba dan Tabel Coba1 Database Skripsi_PHP

8. Setelah tabel coba dan coba1 digabungkan, proses selanjutnya adalah menghitung nilai business impact untuk masing-masing *threat*. Pada tahap requirement disebutkan bahwa business impact dibagi-bagi dalam tiga jenis klasifikasi, yaitu high business impact, medium business impact dan low business impact. Agar ketiga klasifikasi tersebut dapat dikelompokkan untuk setiap threat dari log SAX2, maka pengelompokkan didasarkan pada destination_ip untuk setiap threat. Kolom destination_ip pada log SAX2 menjelaskan tujuan serangan dari setiap *threat*. Pada script php home.php agar klasifikasi tersebut dapat berjalan, maka dilakukan suatu kondisi yang dilakukan oleh switch, sehingga setiap destination_ip yang sesuai akan digolongkan dengan business impact yang diinginkan. Pengelompokkan dilakukan melalui perhitungan kuantitatif dengan setiap business impact dimasukkan pada variable $impactClass.

```
switch ($result_row[4])
{
```

```
case $result_row[4]=="destination_ip1":
$impactClass = "10";
break;
case $result_row[4]=="destination_ip2":
$impactClass = "5";
break;
case $result_row[4]=="destination_ip3":
$impactClass = "2";
break;
default:
$impactClass =”0”

}
```

result_row[] berisi destination_ip, yang kemudian akan digolongkan kedalam impact class yang diinginkan. Script diatas dapat disesuaikan dengan kebutuhan implementasi. Contoh tersebut menggambarkan pengelompokkan untuk suatu LAN yang terdiri atas tiga buah komputer, untuk masing-masing komputer dikelompokkan pada impact class yang berbeda-beda.

9. Perhitungan selanjutnya adalah mendapatkan nilai exposure factor untuk setiap threat yang muncul. Pengelompokkan dilakukan berdasarkan nilai severity pada kolom severity dan menggunakan mekanisme switch dengan setiap exposure factor masuk ke variable $exposure factor.

```
switch ($result_row[9])
{
case $result_row[9]=="severity1":
$exposureFactor ="0.4";
break;
```

```
case $result_row[9]=="severity2":
$exposureFactor ="0.6";
break;
case $result_row[9]=="severity3":
$exposureFactor ="0.8";
break;
case $result_row[9]=="severity4":
$exposureFactor ="1";
default:
$exposureFactor ="0";

}
```

10. Langkah selanjutnya adalah memperoleh nilai impact rating untuk setiap threat (diwakili oleh variable $impactRating), diperoleh dengan mengalikan nilai impact class dan exposure factor.

    ```
    $impactRating = $impactClass*$exposureFactor;
    ```

11. Pada tahap ini akan dihitung nilai risk level untuk masing-masing *threat*, diperoleh dengan mengalikan impact rating dengan jumlah kejadian dimana threat tersebut muncul.

    ```
    $riskLevel = $impactRating*sumofeachimpactRating;
    ```

12. Selanjutnya semua risk level akan dijumlahkan untuk kemudian dirata-ratakan sehingga diperoleh nilai rata-rata risk level. Setelah itu, nilai tersebut dimasukkan ke dalam variable $averagerisklevel dan akan dikelompokkan kedalam nilai kuantitatif dari risk level (high, medium, low). Standar pengelompokkan dapat disesuaikan dengan standar organisasi/perusahaan, biasanya didasarkan dengan kondisi LAN perusahaan atau juga dari pengalaman masa lalu.

```
switch ($averagerisklevel)
{
case ($riskLevel>=?&&$riskLevel<=?):
$risklevelQuan = low
break;
case ($riskLevel>=?&&$riskLevel<=?):
$risklevelQuan = medium
break;
case ($riskLevel>=?&&$riskLevel<=?):
$risklevelQuan = high
default:
$risklevelQuan = 0

}
```

3.4.3 Menu Detail for Source IP dan Menu Detail for Destination IP

Selain dapat melihat nilai risk level LAN, user juga dapat melihat lebih detail setiap *threat* yang muncul. Hal itu ditampilkan dengan bantuan dua buah grafik, yaitu grafik *threat* dalam LAN serta grafik persentase severity level dalam LAN. Untuk masing-masing grafik akan diinspeksi dari source ip address pada menu detail for source ip dan dari destination ip address pada menu detail for destination ip. Secara umum kedua menu tersebut tidak jauh berbeda, perbedaaan hanya terletak dari ip address mana yang akan diinspeksi (source ip address atau destination ip address). Berikut ini merupakan penjelasan dari masing-masing grafik yang terletak pada menu-menu tersebut.

3.4.3.1 Grafik Threat Dalam LAN

Pada grafik ini akan ditampilkan nilai severity maksimum dari masing-masing threat yang dikelompokkan untuk setiap ip address, baik destination maupun source ip address. Grafik yang ditampilkan berupa grafik berbentuk garis yang

menghubungkan masing-masing kelompok waktu dan severity yang sesuai untuk setiap ip address dalam LAN. Dalam menampilkan grafik tersebut user harus memilih inputan tanggal dan jam terlebih dahulu sebelum grafik tersebut muncul. Sebagai contoh, ketika user memilih 1 January 2009 Jam 10.00, maka grafik yang akan muncul adalah semua data pada tanggal 1 January 2009. Semua *threat* yang masuk ke dalam range tersebut akan dipilih oleh script php untuk kemudian dikelompokkan per bagian 10 menit. Artinya ketika user memilih waktu pukul 10.00 maka tiap-tiap data aka dimasukkan dalam kelompok-kelompok berikut ini:

1. 10.00-10.10
2. 10.10-10.20
3. 10.20-10.30
4. 10.30-10.40
5. 10.40-10.50
6. 10.50-10.59

Dari masing-masing pengelompokan tersebut akan dipilih nilai maksimal severity yang kemudian akan ditampilkan. Sebagai contoh ketika antara pukul 10.00-10.10 ada 5 data dengan masing-masing nilai severity-nya adalah 3,2,4,1,3 maka pada grafik akan ditampilkan nilai 4 pada bagian x=10.

3.4.3.2 Grafik Persentase Severity Dalam LAN

Pada tipe grafik ini akan ditampilkan distribusi masing-masing severity level untuk setiap threat yang terjadi pada masing-masing ip address yang ada dalam LAN. Grafik ini merupakan grafik berbentuk pie yang akan menunjukkan masing-masing severity level muncul berapa kali untuk setiap ip address untuk tiap inputan yang dipilih oleh user. Contohnya jika antara pukul 10.00-10.59, pada source ip address1 muncul 10 kali kejadian ber-severity 1, 5 kali kejadian ber-severity 2, 20 kali kejadian ber-severity 3 dan 2 kali kejadian ber-severity 4.

3.4.3.3 Langkah-Langkah Pemrosesan Data Pada Menu Detail For Source IP dan Detail For Destination IP

Agar halaman detail for source ip address (detailforsourceip.php) dan halaman detail for destination ip address (detailfordestinationip.php) dapat menampilkan kedua grafik tersebut maka dibutuhkan langkah-langkah tertentu, yaitu:

1. Empat langkah pertama yang terjadi sama persis dengan langkah-langkah pada menu home.php, mulai dari user yang menginput tanggal dan jam sampai pada tahap script php yang masuk kedalam database skripsi_php.
2. Pada tahap ini dilakukan inisiasi terhadap data-data untuk grafik yang akan digunakan. Agar setiap chart dapat ditampilkan dibutuhkan script berikut ini:

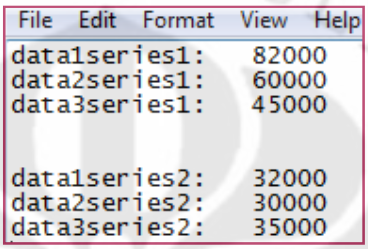
```
<img src="../../jpowered/graph/line-graph.php?
dbinfo=../../skripsi/fullwebsite/[urltodatasource]
config=../../skripsi/fullwebsite/configurationchart1.txt"
width=500
height=400>

<img src="../../jpowered/graph/pie-chart.php?
dbinfo=../../skripsi/fullwebsite/[urltodatasource]
config=../../skripsi/fullwebsite/configurationchart1.txt"
width=500
height=400>
```

Bagian pertama adalah script untuk menampilkan grafik *threat* dalam LAN sedangkan bagian kedua adalah script untuk menampilkan grafik persentase severity dalam LAN. Script diatas merupakan add-on dari ***jpoweredgraph***, yang berguna untuk menampilkan chart yang disesuaikan dengan keinginan user.

Perintah img src berisi directory dimana jenis chart apa yang akan digunakan. Line-graph.php berarti chart yang digunakan berbentuk line sedangkan pie-chart.php berarti chart yang digunakan berbentuk pie. Pada

halaman tersebut berisi pengaturan bagaimana cara membangun sebuah chart berbentuk line yang disesuaikan dengan konfigurasi yang diinginkan user. Bagian dbinfo berisi tempat dimana sumber data diperoleh. Apabila data yang diinginkan bersifat statis maka file dapat berbentuk textfile, namun apabila data yang diinginkan bersifat dinamis, dapat berubah-ubah sesuai inputan, maka file source berbentuk script php dimana didalamnya terdapat mekanisme perubahan database. Berikut ini merupakan contoh dari sebuah file data grafik yang statis:



Gambar 3.11 Contoh Suatu File Source Untuk Suatu Grafik Pada Jpoweredgraph

Bagian berikunya adalah config yang berisi perintah mencari directory file konfigurasi grafik. Konfigurasi secara umum meliputi berapa buah anggota axis (sumbu x) dan ordinat (sumbu y), nama grafik serta font dari tulisan yang terdapat di grafik tersebut.

3. Setelah proses inisiasi grafik tahapan selanjutnya adalah memanipulasi data yang akan dimasukkan ke dalam grafik tersebut.

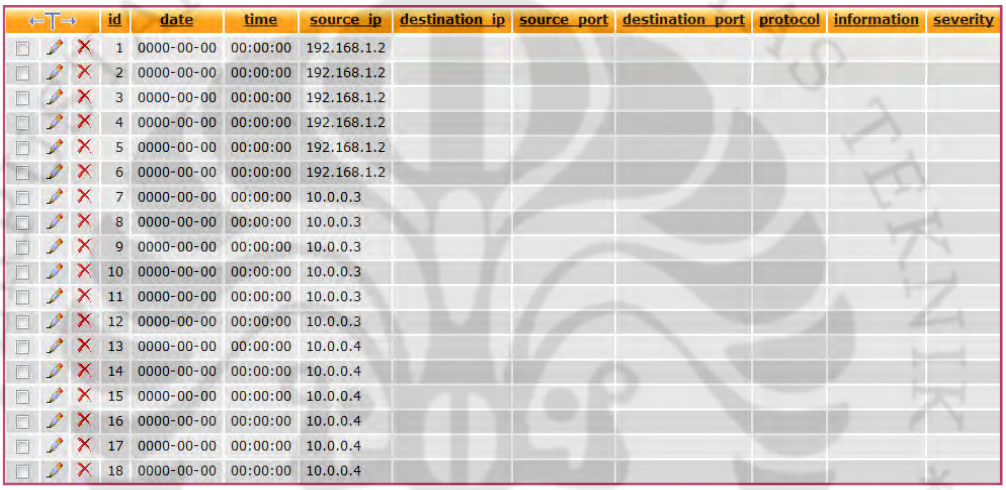
3.4.3.3.1 Langkah-Langkah Pemrosesan Data Untuk Grafik *Threat* dalam LAN

Berikut ini merupakan tahapan manipulasi data untuk grafik *Threat* dalam LAN

- Proses dimulai dengan membaca file log yang dihasilkan oleh program SAX2 IDS, memasukkannya kedalam database skripsi_php tabel coba dan menggabungan tabel coba dengan tabel coba1. Proses tersebut sama dengan yang terjadi pada halaman home.php bagian nomor 5-7.
- Tahapan selanjutnya mengelompokkan masing-masing *threat* kedalam kelompok-kelompok waktu 10 menit untuk kemudian dipilih nilai

severity yang paling tinggi untuk masing-masing kelompok waktu tersebut.

- Setelah nilai-nilai tersebut diperoleh maka satu line yang termasuk dengan pengelompokkan tersebut dimasukkan kedalam tabel data2. Secara default tabel data2 akan berisi 6 buah data yang menjelaskan 6 buah segmen data pembagian per 10 menit dalam satu jam dikali jumlah ip yang akan diinspeksi pada LAN. Berikut ini merupakan contoh isi tabel data2 yang berisi inpeksi untuk 3 komputer dalam LAN



id	date	time	source_ip	destination_ip	source_port	destination_port	protocol	information	severity
1	0000-00-00	00:00:00	192.168.1.2						
2	0000-00-00	00:00:00	192.168.1.2						
3	0000-00-00	00:00:00	192.168.1.2						
4	0000-00-00	00:00:00	192.168.1.2						
5	0000-00-00	00:00:00	192.168.1.2						
6	0000-00-00	00:00:00	192.168.1.2						
7	0000-00-00	00:00:00	10.0.0.3						
8	0000-00-00	00:00:00	10.0.0.3						
9	0000-00-00	00:00:00	10.0.0.3						
10	0000-00-00	00:00:00	10.0.0.3						
11	0000-00-00	00:00:00	10.0.0.3						
12	0000-00-00	00:00:00	10.0.0.3						
13	0000-00-00	00:00:00	10.0.0.4						
14	0000-00-00	00:00:00	10.0.0.4						
15	0000-00-00	00:00:00	10.0.0.4						
16	0000-00-00	00:00:00	10.0.0.4						
17	0000-00-00	00:00:00	10.0.0.4						
18	0000-00-00	00:00:00	10.0.0.4						

Gambar 3.12 Contoh Data-Data Pada Tabel Data2 Database Skripsi_PHP

- Proses akan dilanjutkan dengan pembacaan bagian dbinfo untuk masing-masing source ip (loginchart1.php) dan destination ip (loginchart1-a.php). Untuk masing-masing halaman tersebut proses yang terjadi adalah dilakukannya proses pembacaan data dari database data2 kemudian data-data tersebut akan ditampilkan pada grafik. Berikut ini merupakan script dari loginchart1.php dan loginchart1-a.php yang menjelaskan mekanisme pembacaan database tersebut.

```
$jpDatabase["data"][0]["query"]      = "SELECT severity FROM data1
where source_ip='ipaddress1' ORDER BY time ";
$jpDatabase["data"][0]["valueField"]  = "severity";
```

```
$jpDatabase["data"][1]["query"]        = "SELECT severity FROM data1
where source_ip='ipaddress2' ORDER BY time ";
$jpDatabase["data"][1]["valueField"]   = "severity";

$jpDatabase["data"][0]["query"]        = "SELECT severity FROM data1
where destination_ip='ipaddress1' ORDER BY time ";
$jpDatabase["data"][0]["valueField"]   = "severity";

$jpDatabase["data"][1]["query"]        = "SELECT severity FROM data1
where destination_ip='ipaddress2' ORDER BY time ";
$jpDatabase["data"][1]["valueField"]   = "severity";
```

Untuk setiap ip address secara umum ada 2 langkah yang dilakukan. Sebagai contoh adalah untuk source ip address pada ipaddress1. Langkah pertama adalah perintah untuk memilih kolom severity dari tabel data1 dimana terdapat ipaddress1. Langkah selanjutnya adalah memilih nilai severity tersebut untuk dimasukkan kedalam variable $jpDatabase["data"][0]["valueField"]. Apabila ip address dimasukkan berjumlah 3 buah ip, maka ip ketiga dimasukkan kedalam variable $jpDatabase["data"][2]["valueField"].

3.4.3.3.2 Langkah-Langkah Pemrosesan Data Untukk Grafik Persentase Severity Level dalam LAN

Berbeda dengan grafik sebelumnya, dimana proses manipulasi data dilakukan oleh script php sehingga pada halaman dbinfo (loginchart1.php dan loginchart1-a.php), pada grafik ini proses manipulasi data dilakukan semuanya pada halaman dbinfo (loginchart2.php dan loginchart2-a.php). Hal ini dilakukan karena proses manipulasi data pada grafik ini tidak terlalu sulit dibandingkan pada grafik sebelumnya.

$jpDatabase["data"][0]["query"] = "SELECT COUNT(*) FROM coba INNER JOIN coba1 ON coba.information=coba1.information where source_ip = '192.168.1.2' and severity = '1' and date = '$date' and time between '$timeSelect' and '$timeEnd' union all SELECT COUNT(*) FROM coba INNER JOIN coba1 ON coba.information=coba1.information where source_ip = '192.168.1.2' and severity = '2' and date = '$date' and time between '$timeSelect' and '$timeEnd' union all SELECT COUNT(*) FROM coba INNER JOIN coba1 ON coba.information=coba1.information where source_ip = '192.168.1.2' and severity = '3' and date = '$date' and time between '$timeSelect' and '$timeEnd' union all SELECT COUNT(*) FROM coba INNER JOIN coba1 ON coba.information=coba1.information where source_ip = '192.168.1.2' and severity = '4' and date = '$date' and time between '$timeSelect' and '$timeEnd'";

$jpDatabase["data"][0]["valueField"] = "COUNT(*)";

Script diatas merupakan script dari halaman loginchart2.php, yaitu untuk source ip address. Untuk perhitungan detail pada destination ip address perubahan cukup dilakukan pada tulisan source_ip yang diganti dengan destination_ip. Pada script tersebut terdapat berberapa langkah yang dilakukan.

- Pertama adalah perintah untuk melakukan penggabungan (join) tabel coba1 dan tabel coba untuk mendapatkan satu baris informasi yang utuh, kemudian dilakukan filtering mengenai kondisi-kondisi, dimana masing-masing kondisi dihubungkan oleh logika AND sehingga ketiganya harus benar. Ada tiga buah kondisi, yaitu berupa source_ip, severity dan timebetween. Severity pada langkah ini adalah severity level1 sedangkan timebetween berarti range waktu yang menunjukkan waktu yang telah dipilih user.
- Hasil dari langkah pertama adalah jumlah baris database mengenai *threat* yang memiliki nilai severity level 1yang sesuai dengan kondisi yang telah didefinisikan.
- Langkah ketiga sampai langkahkelima hampir sama dengan langkah pertama, perbedaan hanya terdapat pada nilai severity level yang dihitung,

yaitu severity level2 untuk langkah ketiga, severity level3 untuk langkah keempat dan severity level4 untuk langkah kelima.

- Hasil severity pada langkah pertama, ketiga, keempat dan kelima kemudian digabung sehingga keempat nilai severity tersebut akan berada dalam satu line.
- Satu line tersebut kemudian akan dibaca oleh script php dan dimasukkan dalam variable $jpDatabase["data"][0]["valueField"]. Untuk ip address selanjutnya maka perubahan cukup dilakukan pada angka dimensi array yang kedua.

BAB 4
IMPLEMENTASI DAN UJI COBA SISTEM PADA JARINGAN

Setelah pada akhir bab 3, dimana proses perancangan website ”Web-Based Intrusion Detection and Network Risk Monitoring” selesai dilakukan, pada bab 4 ini proses dilanjutkan dengan melakukan implementasi dan uji coba sistem pada suatu jaringan. Untuk itu diperlukan suatu suatu uji coba dimana terjadi penyesuaian variabel-variabel pada website yang disesuaikan terhadap kebutuhan untuk studi kasus tersebut.

4.1 Uji Coba Pada LAN Perusahaan X

Studi kasus yang akan dibuat adalah perancangan program perhitungan IT Risk Level pada LAN perusahaan X. Perusahaan X merupakan perusahaan berskala menengah yang baru berdiri sekitar satu tahun. Perusahaan X merupakan perusahaan jasa pengiriman barang, melayani pengiriman jasa dari dalam negeri. Konsumen yang ingin mengirim barang datang langsung ke kantor(baru memiliki satu kantor) dan semua catatan pengiriman disimpan pada satu komputer yang diberi id komputer operasional. Perusahaan X memiliki topologi LAN sederhana dengan 1 switch yang langsung terhubung dengan penyedia layanan internet. Dari LAN tersebut terhubung 8 komputer, yaitu:

1. Server operasional. Pada server ini terdapat semua catatan operasional perusahaan, mulai dari transaksi konsumen serta catatan keuangan perusahaan
2. Server web aplikasi. Perusahaan X juga memiliki web-based application sehingga pelanggan yang tidak sempat datang langsung untuk mengirim barang, dapat langsung memasukkan data mengenai barang apa yang ingin dikirim. User menginput nama, alamat asal dan tujuan pengiriman, kemudian karyawan dari perusahaan X akan datang ke alamat asal konsumen, menyelesaikan pembayaran kemudian mengirimkan barang tersebut ke alamat tujuan
3. Server HRD. Pada server ini terdapat data karyawan, meliputi absensi serta data-data pribadi karyawan

4. 5 komputer selanjutnya merupakan PC yang terdiri dari 1 PC direktur perusahaan, serta 4 PC untuk karyawan yang dipakai bergantian.

Perusahaan X berencana untuk mengimplementasikan IT Risk Management. Hal tersebut dilakukan karena perusahaan X memiliki rencana jangka panjang untuk memperluas area usaha sampai ke luar negeri. Untuk mencapai rencana jangka panjang tersebut, perusahaan X berencana untuk mengadopsi sistem yang lebih maju, seperti pemanfaatan terknologi informasi dan komputer. Agar implementasi teknologi informasi dan komputer tersebut berjalan lancar, maka pengimplementasian IT Risk Management sangat diperlukan. Langkah awal yang dilakukan adalah dari segi pengawasan pada LAN perusahaan. Sebelum mengupayakan pembenahan yang lebih kompleks, pimpinan perusahaan X berencana untuk membenahi sistem internal perusahaan. Tiap-tiap server perusahaan harus dijaga ketat, terlebih lagi dari kemungkinan pencurian/perusakan data dari pihak internal. LAN perusahaan yang hanya memiliki satu segmen jaringan mempermudah pihak internal perusahaan untuk melakukan hal-hal tersebut. Kemungkinan *threat* yang muncul sangat besar berasal dari 4 komputer yang dipakai bergantian oleh karyawan perusahaan X. Salah satu cara yang ditempuh untuk melindungi server-server perusahaan adalah dengan mengimplementasikan website ”Web-Based Intrusion Detection and Network Risk Monitoring”

4.2 Penyesuaian dan Perubahan Sistem Pada Website “Web-Based Intrusion Detection and Network Risk Monitoring”

Penyesuaian dan perubahan pada sistem dilakukan karena perancangan yang diperoleh pada akhir bab 3 masih berupa source code yang belum memiliki nilai-nilai pasti. Penyesuaian dan perubahan dilakukan meliputi hal-hal berikut ini:

- Perubahan Pada Klasifikasi Aset Perusahaan

  Berdasarkan penggolongan yang ditetapkan pada tahapan identifikasi, maka 8 buah komputer perusahaan yang merupakan aset perusahaan dibagi dalam kelompok berikut ini:

1. Server Operasional dikelompokkan dalam High Business Impact-Asset Class karena perannya yang sangat penting pada perusahaan.
2. Server Web Aplikasi dikelompokkan dalam Mediun Business Impact-Asset Class karena aset ini akan menimbulkan kerugian pada perusahaan apabila terjadi masalah tetapi tidak menggangu keseluruhan operasional perusahaan.
3. Server HRD dikelompokkan dalam Low Business Impact-Asset Class karena bila terjadi masalah pada aset ini, tidak berpengaruh signifikan pada kinerja operasional perusahaan.
4. Semua komputer perusahaan dikategorikan dalam Medium Business Impact-Asset class
5. Apabila terjadi scan protocol tertentu, sehingga source dan destination IP akan bernilai 0, maka Impact-Asset Class untuk IP tersebut akan dikelompokkan pada Medium Business Impact-Asset Class. Hal tersebut disebabkan munculnya scan protocol dapat dikategorikan langkah awal terjadinya serangan.

- Perubahan Pada Kemungkinan Munculnya Threat

Perubahan dilakukan karena pada standar yang menyebutkan *range* munculnya *threat* didasarkan per tahun tidak cocok apabila diimplementasikan pada perusahaan X. Hal tersebut disebabkan perubahan dan *threat* yang terjadi pada LAN sangat cepat terjadi, bahkan dapat terjadi dalam hitungan jam. Oleh karena itu, perubahan sangat perlu dilakukan dari yang awalnya didasarkan per tahun menjadi per jam, sehingga standar pada kemungkinan munculnya threat menjadi :

- High : 61-∞ kali muncul threat dalam satu jam
- Medium: 31-60 kali muncul threat dalam satu jam
- Low : 0-30 kali muncul threat dalam satu jam

Sehingga pada total risk level rata-rata, terjadi perubahan yaitu:

- High Risk Level: 420-∞
- Medium Risk Level: 200-400
- Low Risk Level: 0-180

4.3 Menentukan Skenario Penyerangan Terhadap LAN Perusahaan X

Skenario yang dilakukan bertujuan untuk mengetes website yang telah dibuat. Secara ringkas skenario ini memiliki tujuan utama untuk mengubah data yang dimiliki pada salah satu server perusahaan X. Pada skenario ini saya akan mengubah data pada server HRD. Pada server HRD telah ter-*install* database karyawan perusahaan X, disimpan menggunakan mysql database. Database perusahaan disimpan pada database dbperusahaan. Pada database dbperusahaan memiliki bermacam-macam tabel, yaitu tabel biodata, tabel gaji karyawan, tabel prestasi karyawan. Pemilihan untuk membuat satu database yang terdiri atas bermacam-macam tabel daripada memilih membuat satu tabel yang dilakukan agar mempermudah pengaturan apabila ada data yang diubah. Selain itu, dengan adanya tabel yang bermacam-macam dapat lebih mudah apabila database tersebut akan digunakan untuk pengolahan. Skenario penyerangan dilakukan dalam tiga tahap, dalam setiap tahapan tersebut pada website akan dilakukan inspeksi pada setiap halamannya untuk melihat output yang dihasilkan pada setiap halaman tersebut.

Simulasi jaringan dilakukan menggunakan VMware. Vmware digunakan untuk membuat suatu OS virtual yang dapat dijalankan bersama-sama dengan OS utama dari komputer. Selain itu, dengan menggunakan Vmware dapat dimungkinkan kedua komputer tersebut saling berhubungan dalam suatu LAN virtual. OS komputer pertama pada studi kasus ini merupakan komputer penyerang, memiliki IP 10.0.0.50. Sedangkan komputer pada Vmware adalah server HRD, memiliki IP 10.0.0.10.

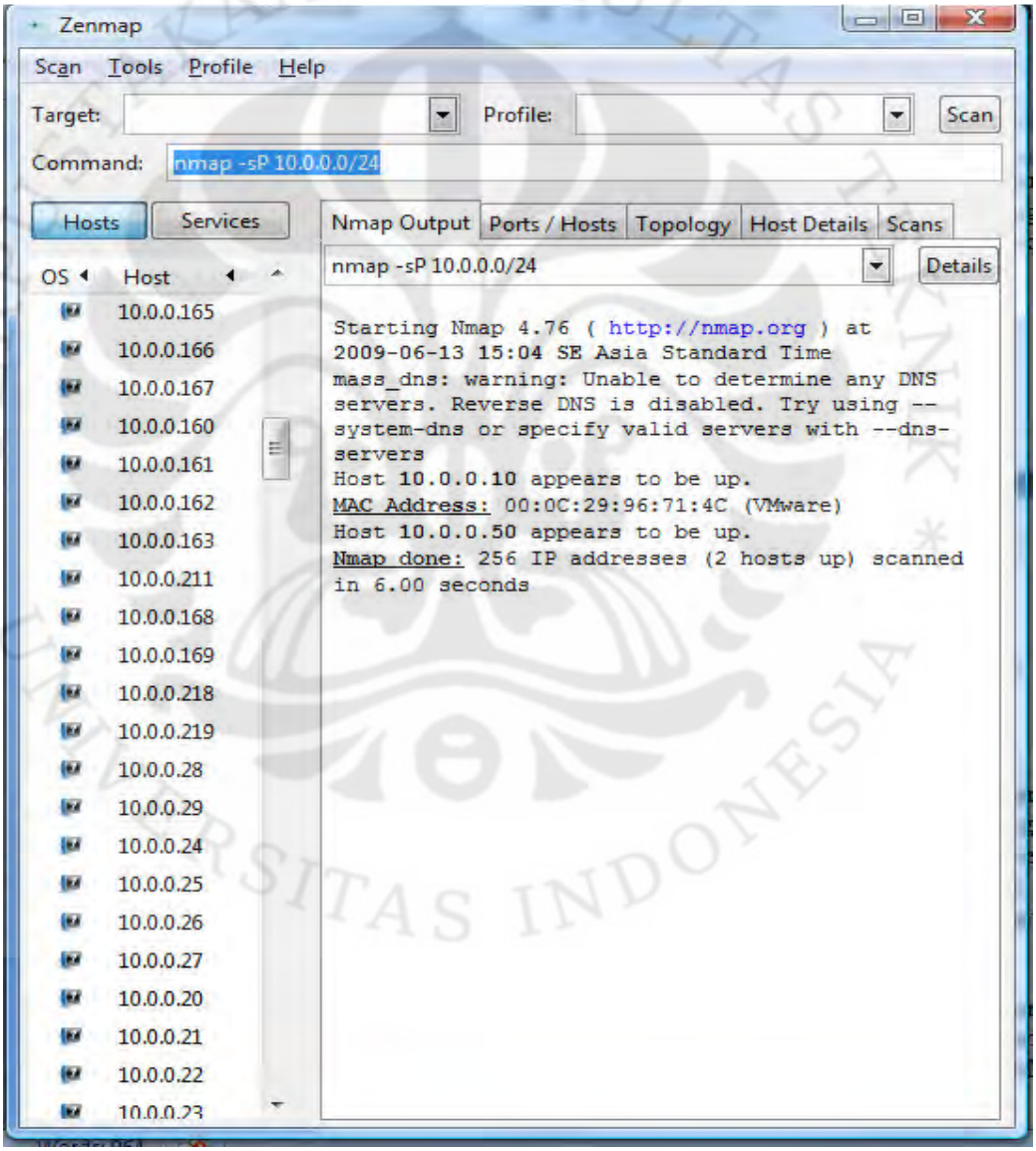
4.3.1 Skenario Tahap 1

Pada tahap pertama komputer penyerang memulai untuk merencanakan mengubah daftar gaji yang terletak di database dbkaryawan. Langkah pertama yang ia lakukan adalah mencari tahu terlebih dahulu IP Server HRD. Untuk memperoleh IP server HRD, penyerang menggunakan *tools* Nmap. *Tools* ini sangat berguna untuk melakukan *scan* terhadap ip maupun port-port pada suatu

LAN. Komputer penyerang menggunakan *tools* Nmap kemudian mengetikkan perintah :

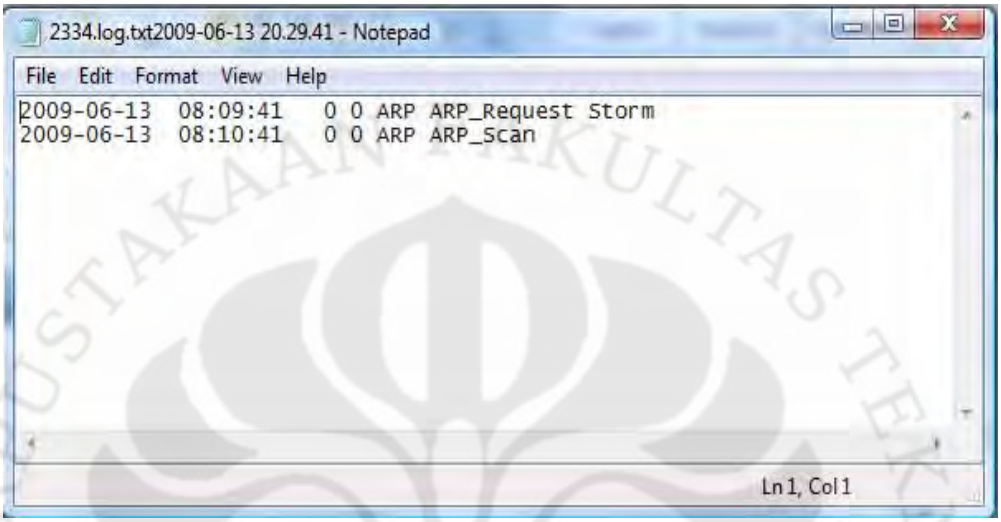
nmap -sP 10.0.0.0/24

Perintah tersebut bertujuan untuk melihat komputer-komputer apa saja yang terhubung pada segmen jaringan yang sama dengan komputer penyerang. Karena perusahaan X hanya memiliki satu LAN, maka perintah pada nmap dapat lebih mudah untuk dilakukan. Berikut ini merupakan output pada Nmap:



Gambar 4.1 *Output* dari Nmap Untuk Perintah Pencarian Host Dalam LAN

Dari gambar 4.1 terlihat bahwa selain IP penyerang terdapat satu IP lagi yang aktif, yaitu 10.0.0.10. Sedangkan gambar 4.2 merupakan output dari log yang dihasilkan IDS SAX2.



Gambar 4.2 *Output* dari Log Yang Dihasilkan IDS SAX2 Untuk Perintah Pencarian Host Dalam LAN

Dari log yang dihasilkan oleh IDS SAX2 terlihat ada dua macam threat yang muncul, yaitu ARP Request Storm dan ARP Scan. ARP Scan merupakan suatu perintah menggunakan protocol ARP untuk mencari dan melaporkan IP address apa saja yang ada pada jaringan lokal. ARP scan tidak memiliki source dan destination ip karena ARP bekerja pada link layer, dimana pada layer ini source and destination address berupa MAC address, bukan IP address. Hal tersebut menyebabkan SAX2 memberikan response pada field source and destination address bernilai 0, karena SAX2 tidak dapat mendeteksi MAC address. ARP Request Storm terjadi karena ARP scan dilakukan berulang kali sehingga menyebabkan flooding pada jaringan. ARP Request Storm mengindikasikan jumlah request paket-paket ARP yang muncul per detik atau ARP request/sec.

Setelah IDS SAX2 menghasilkan log tersebut, langkah selanjutnya adalah menjalankan halaman home.php dengan memasukkkan *input* sesuai tanggal dan waktu terjadinya kejadian tersebut, yaitu 13 juni 2009 pada *input* tanggal dan

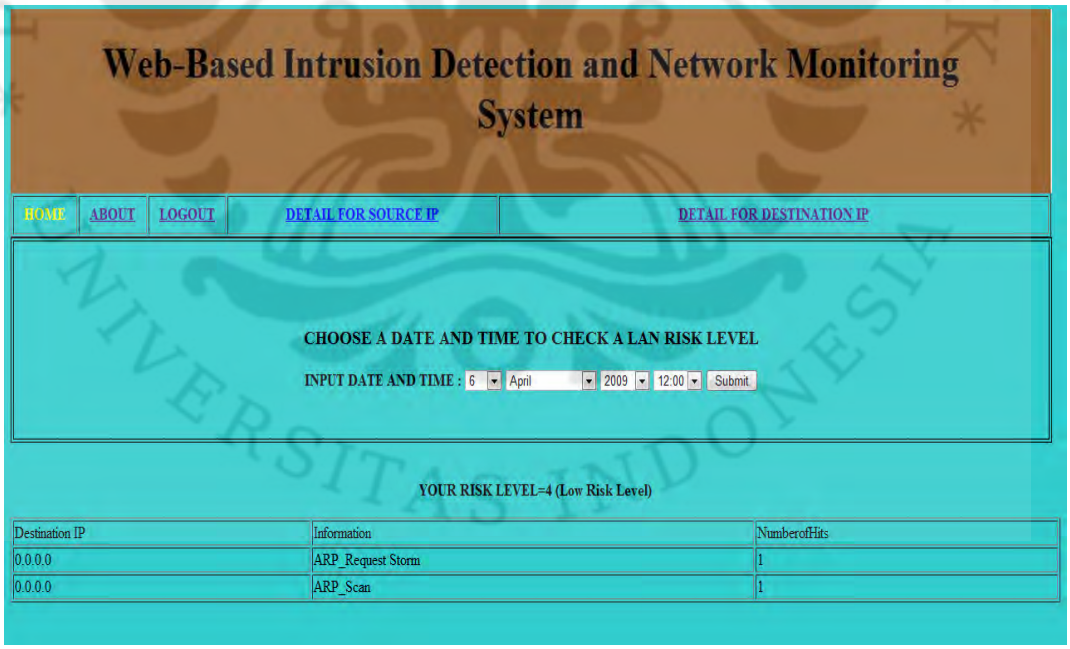
memilih jam 20:00 pada *input* jam, kemudian menekan tombol submit. Log yang dihasilkan IDS SAX2 kemudian akan masuk ke database skripsi_php tabel coba. Berikut ini adalah isi dari tabel coba tersebut:

	id	date	time	source_ip	destination_ip	source_port	destination_port	protocol	information
	1	2009-06-13	08:09:41	0	0	0	0	ARP	ARP_Request Storm
	2	2009-06-13	08:10:41	0	0	0	0	ARP	ARP_Scan
	3	0000-00-00	00:00:00	0	0				

Gambar 4.3 Isi Tabel Coba Database Database Skripsi_PHP

Terlihat pada gambar diatas, isi yang muncul pada database coba sama dengan isi dari log yang dihasilkan oleh IDS SAX2. Hal tersebut berarti *script* php pada halaman home.php telah berhasil melakukan perhitungan mulai pada tahap membaca log IDS SAX2 sampai memindahkan log tersebut kedalam database skripsi_php tabel coba.

Selanjutnya pada website halaman home.php akan dihasilkan average risk level pada nilai 4 atau bernilai Low.



Gambar 4.4 Output Halaman Home.PHP Untuk Uji Coba Pada Tahap 1

Setelah itu, akan dilakukan perhitungan manual average risk level sesuai yang tercantum pada tabel coba.

1. Pada baris 1, destination ip 0 memiliki nilai impact class (bernilai 5), kemudian dikalikan dengan nilai severity pada threat tersebut yang merupakan severity level 3 (bernilai 0,8). Sehingga untuk threat tersebut akan dihasilkan impact rating bernilai 4. Selanjutnya karena kejadian yang sama dengan threat tersebut hanya terjadi satu kali maka nilai risk levelnya adalah 4.
2. Pada baris 2, destination ip 0 memiliki nilai impact class (bernilai 5), kemudian dikalikan dengan nilai severity pada threat tersebut yang merupakan severity level 3 (bernilai 0,8). Sehingga untuk threat tersebut akan dihasilkan impact rating bernilai 4. Selanjutnya karena kejadian yang sama dengan threat tersebut hanya terjadi satu kali maka nilai risk levelnya adalah 4.
3. Nilai risk level rata-rata pada tabel coba adalah 4, sesuai perhitungan (4+4)/2=4

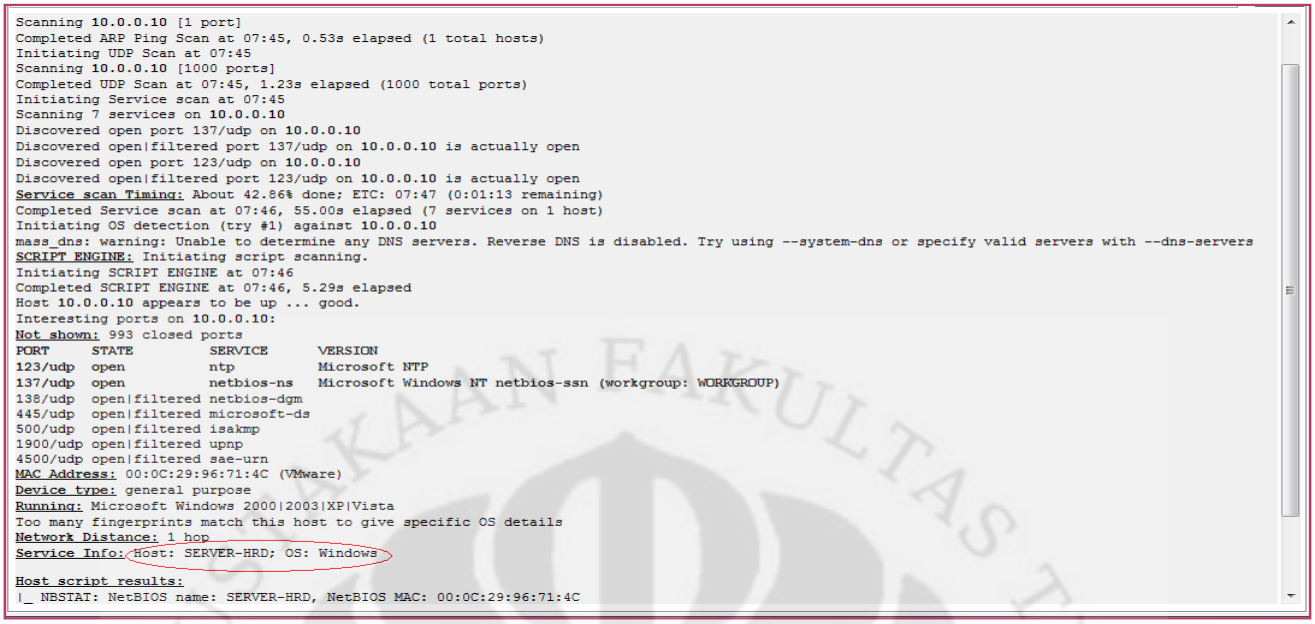
Baik perhitungan manual maupun hasil perhitungan menghasilkan nilai yang sama, yaitu average risk level pada nilai 4.

4.3.2 Skenario Tahap 2

Setelah penyerang mengetahui bahwa pada LAN yang sama dengan LAN komputer yang ia miliki terdapat satu komputer dengan ip 10.0.0.10, kemudian ia melakukan pengecekan apakah komputer tersebut merupakan server HRD. Untuk dapat memperoleh mengenai informasi mengenai suatu komputer, pada tools Nmap, ia melakukan inputan berikut ini:

```
nmap -PE -PA21,23,80,3389 -A -v -T4 10.0.0.10
```

Berikut ini merupakan output yang dihasilkan Nmap:



Gambar 4.5 *Output* Nmap Untuk Perintah *Scan* Port Host 10.0.0.10

Dari hasil output Nmap dapat dilihat bahwa komputer dengan IP 10.0.0.10 benar merupakan server-HRD.

IDS SAX 2 kemudian merespon serangan yang dilakukan oleh IP 10.0.0.50. Berikut ini adalah log yang dihasilkan oleh IDS SAX 2 (Lampiran C):

Dari log tersebut ada beberapa macam *threat* yang muncul, yaitu:

1. ICMP_Port Unreachable (source IP=10.0.0.10, destination IP=10.0.0.50)

   IP 10.0.0.10 mengirimkan pesan *error* kepada komputer penyerang bahwa *request* yang dikirimkan sebelumnya tidak tersedia atau tidak dapat direspon oleh IP 10.0.0.10. Pesan tersebut muncul karena saat penyerang melakukan perintah slow comprehensive scan melalui Nmap, *tools* tersebut akan mengirimkan perintah-perintah tertentu kepada IP tersebut agar informasi mengenai IP tujuan dapat diketahui, biasanya IP penyerang akan mengirim perintah-perintah pada port-port IP tujuan. Akibat dari perintah tersebut komputer yang diserang akan melakukan respon berupa ICMP_Port Unreachable.

2. UDP_Green_Trojan (source ip=10.0.0.50, destination ip=10.0.0.10)

Threat UDP Green Trojan muncul akibat IDS SAX2 mendeteksi adanya kemungkinan komputer penyerang mencoba untuk mengontrol komputer dengan IP 10.0.0.10 dengan menggunakan trojan jenis green trojan.

3. UDP_SQL_Ping (source ip=10.0.0.50, destination ip=10.0.0.10)

Salah satu perintah lainnya yang dilakukan *tools* Nmap adalah mengecek apakah SQL ter-*install* pada komputer tujuan. Ketika Nmap melakukan hal tersebut, IDS SAX2 mendeteksi perintah tersebut dengan mengeluarkan pesan UDP SQL Ping

4. ICMP_Net Unreachable (source ip=10.0.0.50, destinationip=10.0.0.10)
5. ICMP_Host Unreachable (source ip=10.0.0.50, destination ip=10.0.0.10)
6. UDP_Port scan

Threat UDP port Scan muncul karena pada perintah slow comprehensive scan melalui Nmap, tools tersebut akan melakukan scan terhadap port UDP yang dimiliki oleh IP 10.0.0.10

7. UDP_DOS ISAKMP invalid identification payload attempt (source ip=10.0.0.50, destination ip=10.0.0.10)

Perintah ini akan muncul untuk memperingatkan user bahwa source ip=10.0.0.50 memiliki kemungkinan besar untuk mengeksploitasi komputer dengan ip 10.0.0.10 melalui tindakan Denial Of Service. Biasanya pengekspoitasian dilakukan melalui payload yang dimiliki suatu tools tertentu (hal ini akan dijelaskan pada bagian 4.3.3)

8. ICMP_Ping Unusual Length (source ip=10.0.0.50, destination ip=10.0.0.10)

Paket-paket ping biasanya memiliki panjang kurang dari100 bytes. Apabila jumlah tersebut melebihi, maka ada kemungkinan ada sebuah software yang yang berasal dari source ip yang akan mengirimkan data-

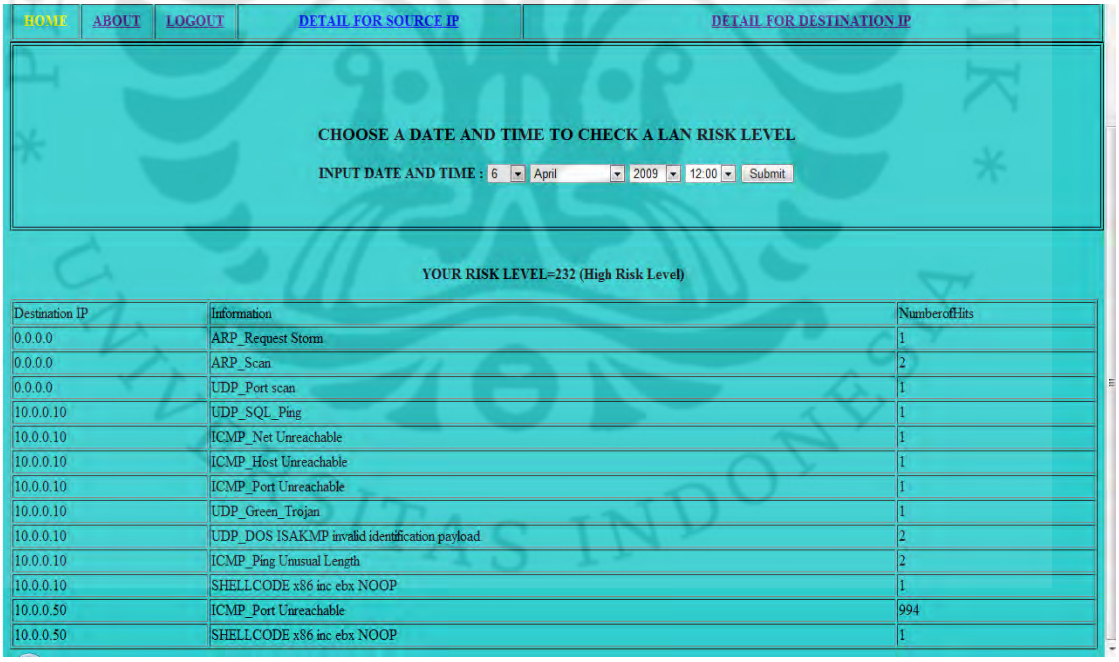
data untuk dapat menyerang destination ip melalui *backdoor* (*Backdoor* akan dijelaskan lebih lanjut pada bagian 4.3.3)

9. SHELLCODE x86 inc ebx NOOP (source ip=10.0.0.50, destination ip=10.0.0.10)

   Pesan ini akan muncul apabila ada kemungkinan akan dilakukan ekseskusi *shellcode* pada destination IP oleh source IP untuk mengubah data-data pada destination ip

10. ARP_Scan

Langkah selanjutnya adalah melihat lagi output yang dihasilkan website, pada halaman home.php. Nilai average risk level yang tercantum merupakan hasil penjumlahan nilai-nilai threat pada langkah pertama ditambah dengan jumlah nilai-nilai threat pada tahap kedua. Berikut ini merupakan output yang dihasilkan pada halaman tersebut:



Destination IP	Information	Numberofhits
0.0.0.0	ARP_Request Storm	1
0.0.0.0	ARP_Scan	2
0.0.0.0	UDP_Port scan	1
10.0.0.10	UDP_SQL_Ping	1
10.0.0.10	ICMP_Net Unreachable	1
10.0.0.10	ICMP_Host Unreachable	1
10.0.0.10	ICMP_Port Unreachable	1
10.0.0.10	UDP_Green_Trojan	1
10.0.0.10	UDP_DOS ISAKMP invalid identification payload	2
10.0.0.10	ICMP_Ping Unusual Length	2
10.0.0.10	SHELLCODE x86 inc ebx NOOP	1
10.0.0.50	ICMP_Port Unreachable	994
10.0.0.50	SHELLCODE x86 inc ebx NOOP	1

Gambar 4.6 Output Halaman Home.PHP Untuk Uji Coba Pada Tahap 2

Langkah selanjutnya adalah melakukan perhitungan manual dari tiap-tiap threat tersebut sehingga dapat diketahui nilai IT Risk Level rata-rata dari semua threat tersebut:

- ICMP_Port Unreachable.

Threat tersebut memiliki destination ip=10.0.0.50 (impact class=5) dan memiliki severity 2 (0,6), sehingga menghasilkan impact rating sebesar 3. Jumlah yang sama untuk threat tersebut sebanyak 994 kali sehingga menghasilkan nilai risk level sebesar 3x994=2982

Selain itu terdapat satu threat yang sama namun source ip dan destination ip yang muncul berkebalikan yaitu source ip=10.0.0.50 dan destination ip 10.0.0.10. untuk threat tersebut akan dihasilkan risk level sebesar 2x0,6x1=1,2

- UDP_Green_Trojan

Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 2 (0,6), sehingga menghasilkan impact rating sebesar 1,2. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,2

- UDP_SQL_Ping

Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 2 (0,6), sehingga menghasilkan impact rating sebesar 1,2. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,2

- ICMP_Net Unreachable

Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 2 (0,6), sehingga menghasilkan impact rating sebesar 1,2. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,2

- ICMP_Host Unreachable

Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 2 (0,6), sehingga menghasilkan impact rating sebesar 1,2. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,2

- UDP_Port scan

Threat tersebut memiliki destination ip=0 (impact class=5) dan memiliki severity 3 (0,8), sehingga menghasilkan impact rating sebesar 4. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 4

- UDP_DOS ISAKMP

  Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 3 (0,8), sehingga menghasilkan impact rating sebesar 1,6. Jumlah yang sama untuk threat tersebut sebanyak 2 kali sehingga menghasilkan nilai risk level sebesar 3,2

- ICMP_Ping Unusual Length

  Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 3 (0,8), sehingga menghasilkan impact rating sebesar 1,6. Jumlah yang sama untuk threat tersebut sebanyak 2 kali sehingga menghasilkan nilai risk level sebesar 3,2

- SHELLCODE x86 inc ebx

  Threat tersebut memiliki destination ip=10.0.0.10 (impact class=2) dan memiliki severity 3 (0,8), sehingga menghasilkan impact rating sebesar 1,6. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,6

  Selain itu terdapat satu threat yang sama namun source ip dan destination ip yang muncul berkebalikan yaitu source ip=10.0.0.10 dan destination ip 10.0.0.50. untuk threat tersebut akan dihasilkan risk level sebesar 5x0,8x1=4

- ARP_Scan

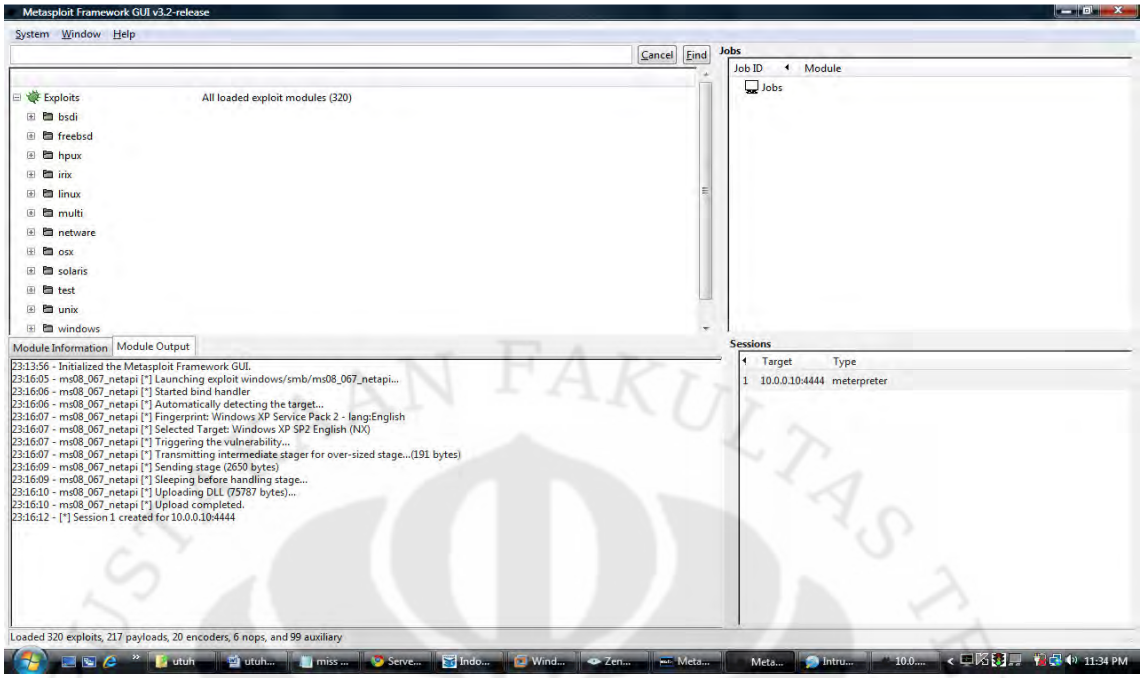
  Threat tersebut memiliki destination ip=0 (impact class=2) dan memiliki severity 3 (0,8), sehingga menghasilkan impact rating sebesar 1,6. Jumlah yang sama untuk threat tersebut sebanyak 1 kali sehingga menghasilkan nilai risk level sebesar 1,6

Dari kesepuluh jenis threat diatas dan ditambah dengan dua data pada tahap dua, maka akan didapatkan nilai rata-rata risk level sebesar 231.8 ≈ 232 atau

berarti risk level untuk semua threat tersebut bernilai High. Kedua perhitungan, baik dari website maupun manual memberikan hasil yang sama. Hal tersebut berarti website " Web-Based Intrusion Detection and Network Risk Monitoring" telah berjalan dengan baik.

4.3.3 Skenario Tahap 3

Setelah informasi mengenai IP address server HRD serta port-port apa saja yang terbuka pada server HRD berhasil diketahui, langkah selanjutnya adalah mengambil alih akses server HRD tersebut. Dengan berhasilnya suatu komputer mengambil alih suatu komputer lain, maka seorang penyerang dapat dengan mudah mengambil data, merusak data ataupun mengubah data dari suatu komputer. Langkah yang dipergunakan pada tahap ini adalah dengan melakukan eksploitasi terhadap *vulnerability* yang dimiliki pada komputer yang akan diserang. Eksploitasi terhadap *vulnerability* dapat dilakukan apabila diketahui service-service apa saja yang sedang dijalankan pada komputer yang akan diserang tersebut, biasanya service-service tersebut berkaitan erat port-port pada komputer. Dari hasil scan Nmap pada skenario tahap 2, dapat diketahui bahwa ada beberapa port yang terbuka pada server HRD, yaitu port 123, 137, 138, 445, 500, 1900 dan 4500. Pada contoh ini port yang akan dimanfaatkan untuk dieksploitasi adalah port 445. Port 445 merupakan port untuk layanan SMB file sharing. Biasanya port ini akan aktif apabila layanan komputer untuk dapat melakukan sharing file antar komputer dalam satu LAN sedang diaktifkan. Untuk melakukan eksploitasi digunakan tools yang bernama metasploit[7]. Metasploit merupakan suatu *tools* yang berisi kumpulan jenis eksploitasi yang memiliki tujuan untuk dapat melakukan koneksi secara *remote* dan merusak komputer yang akan diserang. Gambar 4.7 merupakan tampilan pada menu utama Metasploit v3.2.



Gambar 4.7 Tampilan Menu Utama *Tools* Metasploit v3.2

Pada menu pertama kumpulan exploit digolongkan berdasarkan jenis OS. Karena server HRD menggunakan OS Windows, maka menu Windows dipilih. Selanjutnya, masuk pada menu Windows, akan muncul pilihan menu berdasarkan vulnerability yang dimiliki komputer yang akan diserang. Karena pada contoh ini port yang akan dieksploitasi adalah port SMB File Sharing, maka menu SMB yang dipilih. Pada menu SMB tercantum beberapa macam serangan yang akan dilakukan, penyerang memutuskan untuk memilih jenis serangan yang memiliki kode tahun terbaru, yaitu ms08_067_netapi/Microsoft Server Service Relative Path Stack Corruption. Setelah pilihan tersebut dipilih, user diharuskan melakukan 3 langkah proses input data mengenai serangan yang akan dilakukan, yaitu:

1. Select Your Target

   Pada pilihan ini user harus melakukan input data mengenai jenis OS dari komputer yang akan diserang. Penyerang kemudian memilih automatic targeting untuk memudahkan proses ini.

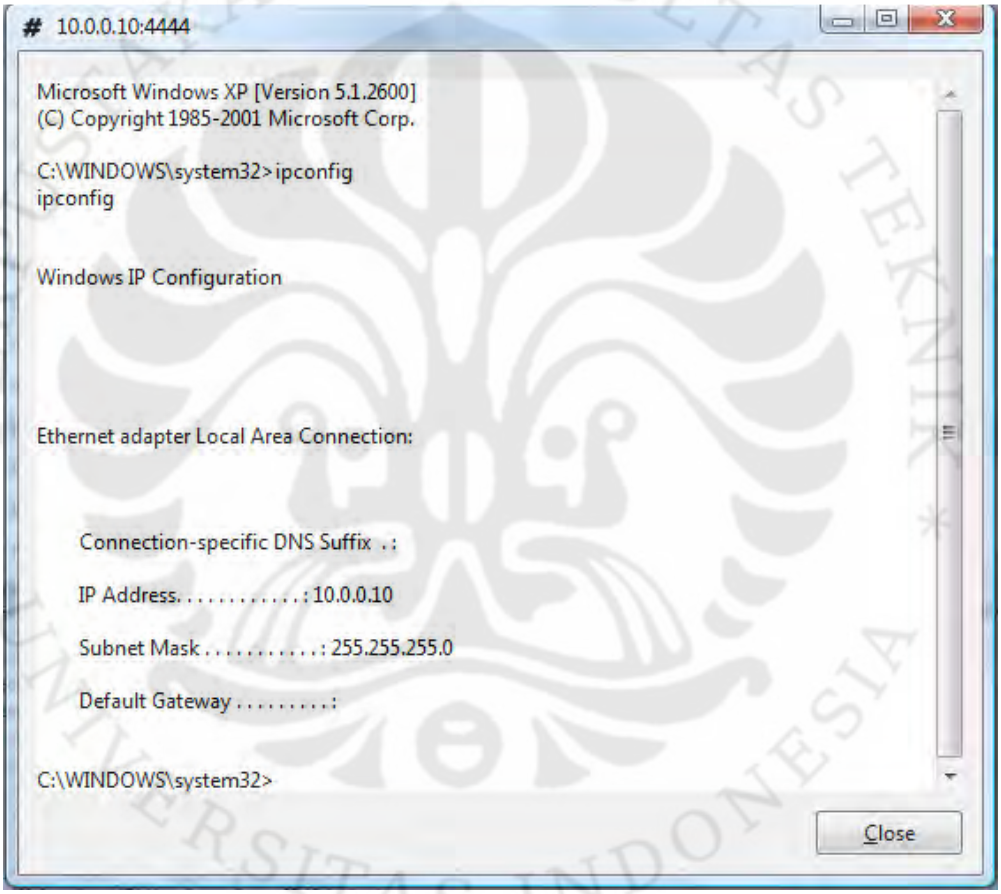
2. Select Your Payload

Payload adalah kode yang disertakan dalam proses penyerangan yang akan dieksekusi pada komputer target

3. Select Your Options

Pada langkah ini user diharuskan melakukan proses input untuk IP address komputer serangan dan port yang akan dieksploitasi

Setelah ketiga langkah tersebut dilakukan, maka komputer 10.0.0.50 berhasil masuk ke komputer 10.0.0.10 , melalui command prompt.



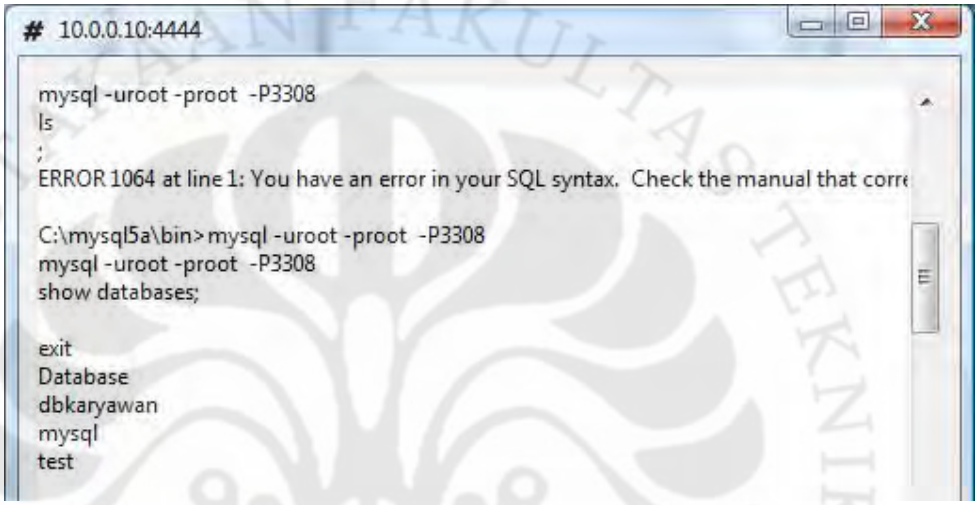
Gambar 4.8 Komputer 10.0.0.50 Berhasil Masuk Ke Komputer 10.0.0.10 Menggunakan Metasploit

Setelah berhasil masuk ke server HRD, penyerang kemudian mencoba untuk mencari lokasi dimanakah folder mysql terletak. Berikut ini adalah langkah-

langkah yang dilakukannya untuk menemukan directory mysql sampai mengubah daftar gaji karyawan:

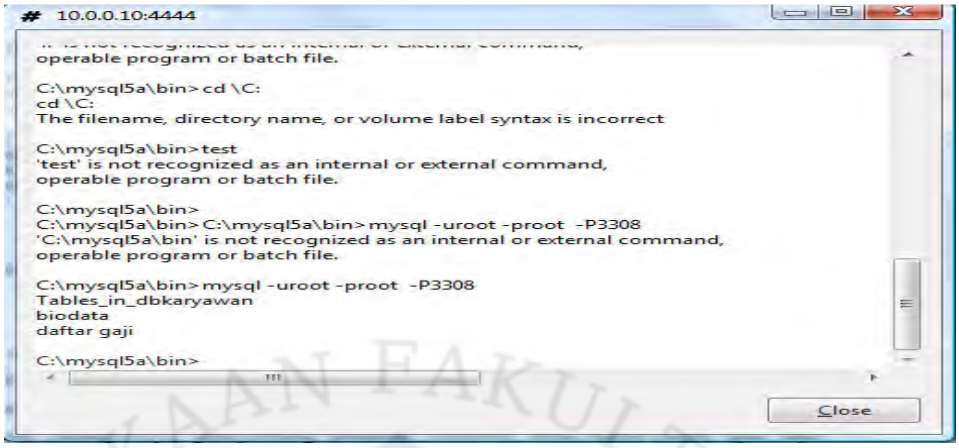
1. Mencari list folder-folder apa saja yang ada di directory C:\

2. Ternyata pada directory tersebut ada sebuah folder yang bernama mysql5a, yang merupakan sebuah folder untuk database. Kemudian penyerang akan melihat database apa saja yang terdapat pada mysql5a tersebut



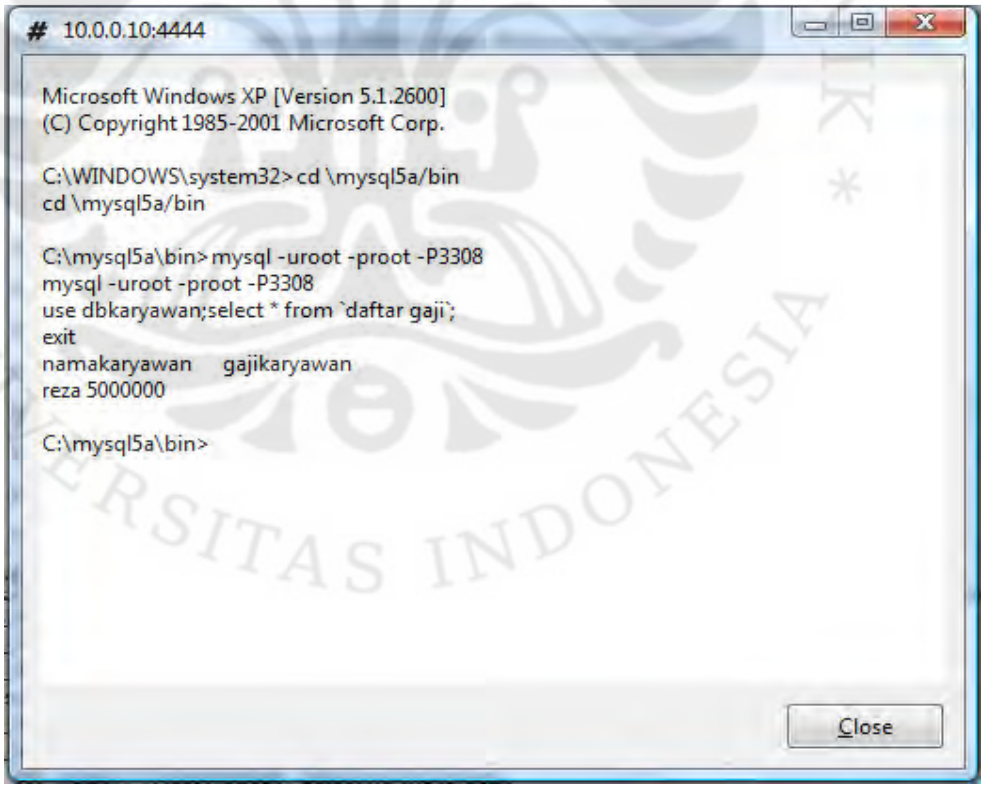
Gambar 4.9 Tampilan Metasploit Yang Menampilkan Database Yang Terdapat Pada Folder Mysql5a

Pada mysql5a terdapat 7 buah database. Karena database yang akan diubah meliputi data karyawan, maka penyerang kemudian masuk menuju database dbkaryawan. Langkah selanjutnya penyerang melihat terlebih dulu tabel-tabel apa saja yang terdapat pada database dbkaryawan



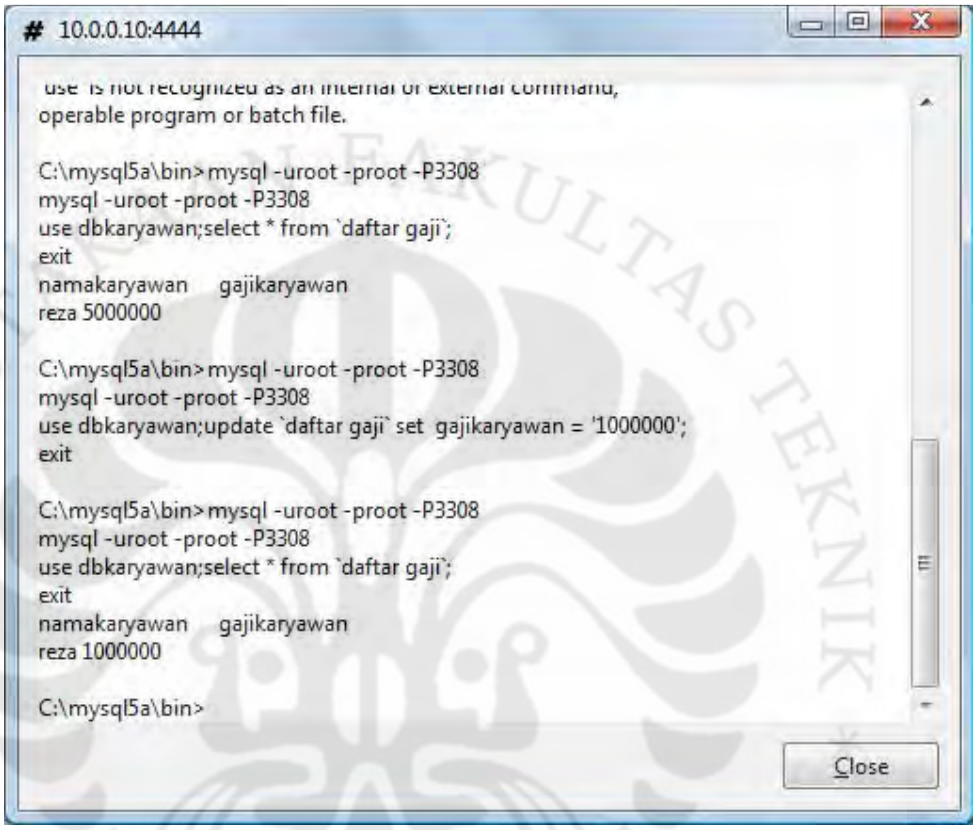
Gambar 4.10 Tampilan Metasploit Yang Menampilkan Tabel-Tabel Yang Terdapat Pada Database dbkaryawan

3. Tujuan utama dari serangan ini adalah mengubah daftar gaji seorang karyawan, sehingga langkah berikutnya yang dilakukan penyerang adalah melihat list yang terdapat di tabel gajikaryawan



Gambar 4.11 Tampilan Metasploit Yang Menampilkan Entry-Entry Yang Terdapat Pada Tabel Daftar Gaji

4. Penyerang sekarang telah mengetahui list yang terdapat di tabel gajikaryawan, kemudian dia akan mengubah salah satu kolom gaji yang terletak di tabel tersebut



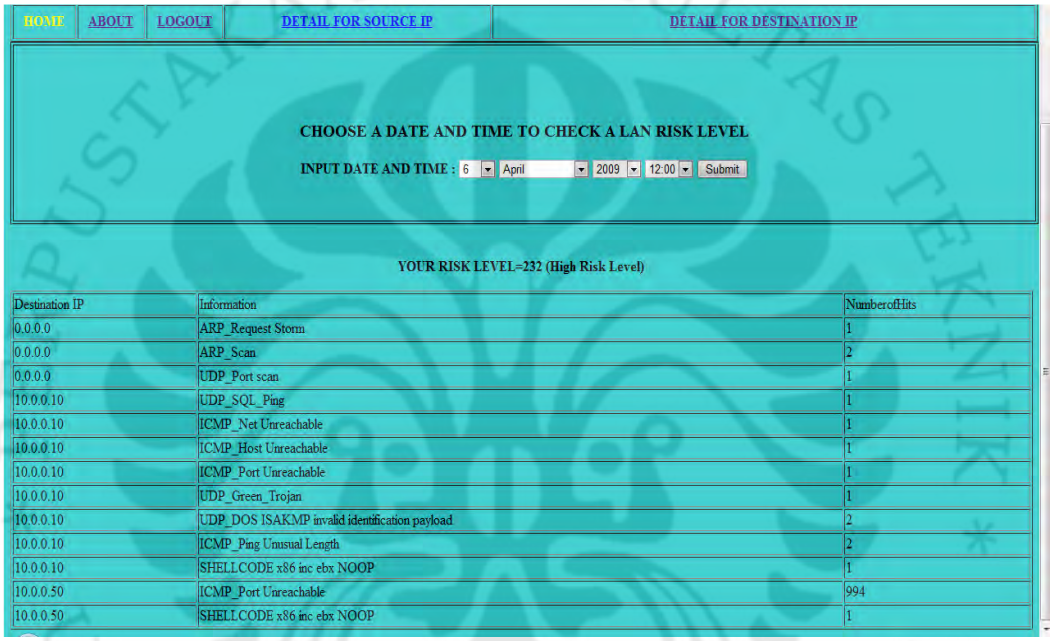
Gambar 4.12 Tampilan Metasploit Yang Mengubah Entry Gaji

Penyerang telah berhasil mengubah salah satu entry, yaitu gaji reza yang sebelumnya 5000000, sekarang menjadi 1000000.

Setelah serangan berhasil dilakukan, langkah selanjutnya adalah melihat output yang dihasilkan pada website halaman home.php

Ternyata pada nilai average risk level yang dihasilkan pada langkah ketiga tidak mengalami perubahan dibandingkan dengan nilai average risk level pada langkah kedua, yang berarti serangan yang terjadi pada langkah ketiga tidak berhasil dideteksi oleh IDS SAX2. Hal tersebut disebabkan dari jenis dari IDS

SAX2 sendiri. IDS SAX2 merupakan network based IDS , yaitu IDS yang dapat mendeteksi serangan dalam lingkup satu jaringan lokal, namun tidak dapat mendeteksi secara rinci serangan yang terjadi pada suatu host. Jenis IDS yang dapat mendeteksi secara rinci serangan untuk suatu host disebut host based IDS. Host based IDS tidak digunakan pada implementasi website karena IDS tersebut tidak dapat melakukan deteksi untuk satu jaringan lokal. Langkah antisipasi yang dapat dilakukan oleh pengguna IDS network based adalah langsung merespon secara cepat terhadap threat yang dikeluarkan IDS tersebut.

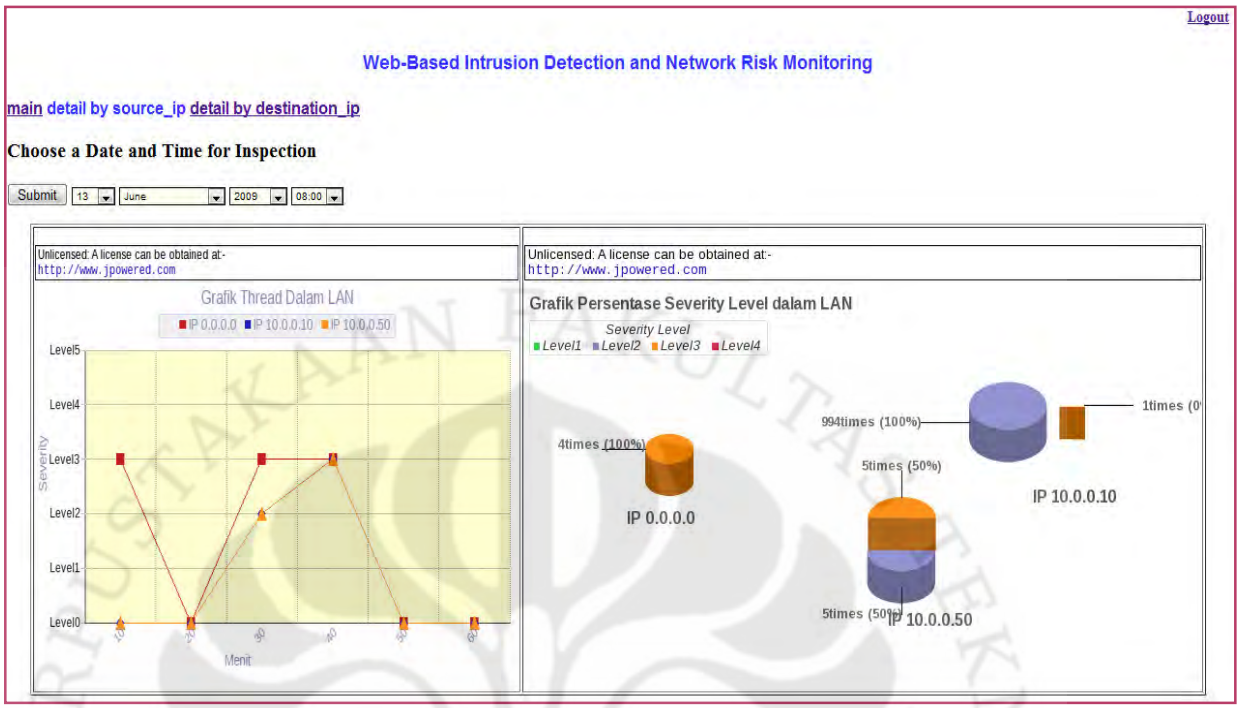


Destination IP	Information	NumberofHits
0.0.0.0	ARP_Request Storm	1
0.0.0.0	ARP_Scan	2
0.0.0.0	UDP_Port scan	1
10.0.0.10	UDP_SQL_Ping	1
10.0.0.10	ICMP_Net Unreachable	1
10.0.0.10	ICMP_Host Unreachable	1
10.0.0.10	ICMP_Port Unreachable	1
10.0.0.10	UDP_Green_Trojan	1
10.0.0.10	UDP_DOS ISAKMP invalid identification payload	2
10.0.0.10	ICMP_Ping Unusual Length	2
10.0.0.10	SHELLCODE x86 inc ebx NOOP	1
10.0.0.50	ICMP_Port Unreachable	994
10.0.0.50	SHELLCODE x86 inc ebx NOOP	1

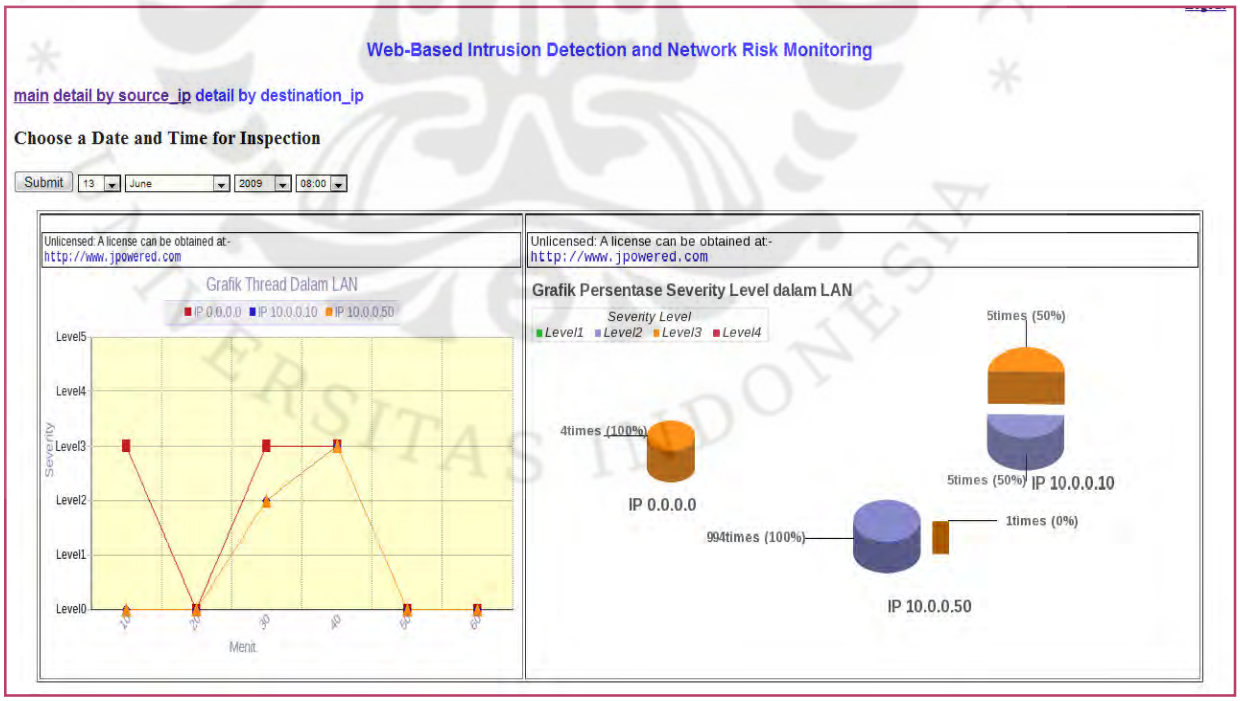
Gambar 4.13 Output Halaman Home.PHP Untuk Uji Coba Pada Tahap 3

Contohnya pada langkah kedua studi kasus ini, IDS SAX2 telah mengeluarkan pesan ”SHELLCODE x86 inc ebx NOOP”, yang berarti akan adanya eksekusi shell code pada IP address tujuan. Ketika administrator telah melihat pesan tersebut seharusnya ia sadar akan adanya bahaya yang akan dialami server HRD, sehingga ia harus melakukan langkah-langkah perlindungan agar data-data yang terdapat pada server HRD tetap aman dari kerusakan dan kecurian.

4.4 Output Menu Detail By Source dan Detail By Destination IP Address



Gambar 4.14 Output Menu Detail By Source IP Untuk UjiCoba Pada Perusahaan X



Gambar 4.15 Output Menu Detail By Destination IP Untuk UjiCoba Pada Perusahaan X

Gambar 4.14 dan 4.15 merupakan output yang dihasilkan pada menu detail by source dan detail by destination IP Address saat dilakukan ujicoba pada perusahaan X. Output-output tersebut dihasilkan saat ketiga tahapan yang telah dijelaskna sebelumnya selesai dilakukan. Dengan adanya grafik-grafik, maka administrator dapat terbantu dalam menyimpulkan apakah yang sebenarnya terjadi pada jaringan yang sedang dimonitor oleh IDS SAX2. Hal itu disebabkan dengan adanya visualisasi yang dilakukan oleh kedua grafik, yaitu Grafik Threat dalam LAN dan Grafik Persentase Severity Level dalam LAN. Grafik Threat dalam LAN memperlihatkan threat-threat tertinggi untuk tiap-tiap segmen waktu 10 menit pada tiap-tiap IP address sesuai menu yang dipilih. Contohnya pada menu detail by destination IP address dapat disimpulkan scan jaringan terjadi sangat sering pada kurun waktu pukul 08.00 sampai 09.00. Hal tersebut diperoleh karena IP 0.0.0.0 yang menunjukkan ketidakmampuan SAX2 melihat MAC address terjadi 3 kali dalam segmen waktu tersebut. Hal tersebut diperjelas pada Grafik Persentase Severity Level dalam LAN dimana untuk IP 0.0.0.0 terjadi masalah sebanyak 4 kali yang semuanya berupa severity ber-level 3.

BAB 5
KESIMPULAN

Setelah melakukan pengujian dari program yang telah dijalankan, maka dapat didapatkan beberapa hal sebagai berikut:

1. Pada skenario ujicoba tahap 1, yaitu pencarian *IP Address* pada suatu LAN, menghasilkan nilai kuantitatif Risk Level sebesar 4 (Low Risk Level).
2. Pada skenario ujicoba tahap 2, yaitu pencarian informasi meliputi port dan nama komputer untuk suatu komputer, menghasilkan nilai kuantitatif Risk Level sebesar 232 (High Risk Level).
3. Pada skenario ujicoba tahap 3, yaitu pengambilalihan suatu komputer target, menghasilkan nilai kuantitatif Risk Level sebesar 232 (High Risk Level).
4. Perbedaan cara ujicoba pada tahap 2 dan tahap 3 terletak dari telah atau belumnya penyerang masuk mengambil alih komputer tujuan. Pada tahap tiga komputer penyerang telah berhasil masuk ke komputer tujuan dan mengubah database server.
5. Nilai Risk Level yang dihasilkan pada ujicoba tahap 2 dan tahap 3 adalah sama. Hal ini disebabkan karena jenis IDS yang digunakan pada skripsi ini adalah network-based IDS, yang tidak dapat melakukan pendeteksian secara rinci pada suatu host/komputer.

DAFTAR ACUAN

[1] G. Stoneburner, A. Goguen and A. Feringa, “Risk Management Guide for Information Technology Systems”, NIST.

[2]“IntrusionDetectionSystem”,http://en.wikipedia.org/wiki/Intrusion_detection_system. Diakses pada 18 Desember 2008.

[3]”UnifiedModelingLanguage”,http://en.wikipedia.org/wiki/Unified_Modeling_Language. Diakses pada 20 Maret 2009.

[4]”The Software Development Life Cycle-For Small To Medium Database”. Document ID:REF-0-02. Version:1.0d.

[5]”TheSecurityRiskManagementGuide”,http://www.microsoft.com/DOWNLOADS/details.aspx?familyid=C782B6D3-28C5-4DDA-A168-3E4422645459&displaylang=en. Diakses pada 15 April 2009.

[6]”Nmap-Zenmap”, http://nmap.org/download.htm. Diakses pada 3 Mei 2009.

[7]”Metasploit”,http**://**www.metasploit.com/. Diakses pada 25 Mei 2009

[8] R.A. Kemmerer and G. Vigna, "Intrusion Detection: A Brief History and Overview", Security and Privacy 2002.

DAFTAR PUSTAKA

"PHP:Basicsyntax".ThePHPGroup.http://www.php.net/manual/en/language.basic-syntax.php

R.A. Kemmerer and G. Vigna, "Intrusion Detection: A Brief History and Overview", Security and Privacy 2002.

“IntrusionDetectionSystem- SAX2”,http://www.ids-sax2.com/.

”TheSecurityRiskManagementGuide”,http://www.windowsecurity.com/articles/Microsoft-Security-Risk-Management-Guide.html -.

R. Bace and P. Mell, "Intrusion Detection System", NIST

“Visual Paradigm For UML”http:// www.visual-paradigm.com/.

JPoweredGraph: Advanced Graphs and Charts for PHP,http://www.jpowered.com/php-scripts/adv-graph-chart/index.htm

”Nmap-Zenmap”, http://nmap.org/download.htm.

”Metasploit”,http://www.metasploit.com/.

“PHP Programming, a comprehensive guide to programming in php” wikipediaopen-contenttextbook
http://en.wikibooks.org/wiki/PHP_Programming/

"EmbeddingPHPinHTML".O'Reilly.2001-05-03.
http://www.onlamp.com/pub/a/php/2001/05/03/php_foundations.html

“Aplikasi Web dengan PHP dan MySQL” (Kasman Peranginangin, Penerbit ANDI Yogyakarta 2007, ISBN 979-763-526-0)

LAMPIRAN A

Contoh Tabel Rencana Implementasi Yang Aman Pada IT Risk Management

(1) Risk (Vulnerability/ Threat Pair)	(2) Risk Level	(3) Recommended Controls	(4) Action Priority	(5) Selected Planned Controls	(6) Required Resources	(7) Responsible Team/Persons	(8) Start Date/ End Date	(9) Maintenance Requirement/ Comments
Unauthorized users can telnet to XYZ server and browse sensitive company files with the *guest* ID	High	• Disallow inbound telnet • Disallow "world" access to sensitive company files • Disable the *guest* ID or assign difficult-to-guess password to the *guest* ID	High	• Disallow inbound telnet • Disallow "world" access to sensitive company files • Disabled the *guest* ID	10 hours to reconfigure and test the system	John Doe, XYZ server system administrator; Jim Smith, company firewall administrator	9-1-2001 to 9-2-2001	• Perform periodic system security review and testing to ensure adequate security is provided for the XYZ server

(1) The risks (vulnerability/threat pairs) are output from the risk assessment process
(2) The associated risk level of each identified risk (vulnerability/threat pair) is the output from the risk assessment process
(3) Recommended controls are output from the risk assessment process
(4) Action priority is determined based on the risk levels and available resources (e.g., funds, people, technology)
(5) Planned controls selected from the recommended controls for implementation
(6) Resources required for implementing the selected planned controls
(7) List of team(s) and persons who will be responsible for implementing the new or enhanced controls
(8) Start date and projected end date for implementing the new or enhanced controls
(9) Maintenance requirement for the new or enhanced controls after implementation.

LAMPIRAN B

Langkah-Langkah Perhitungan Pada Microsoft-Security Risk Management Guide

1. Menentukan Impact Rating Dari Masing-Masing Ancaman Pada Setiap Aset Perusahaan

Impact class	Impact Class Value (V)
HBI	10
MBI	5
LBI	2

Exposure Factor Rating	Exposure Factor (EF)	Impact Rating (V * EF)	Impact Range	Summary Level Comparison
5	100%		7 - 10	High
4	80%		4 - 6	Medium
3	60%		0 - 3	Low
2	40%			
1	20%			

Exposure Rating	Confidentiality or Integrity of Asset
5	Severe or complete damage to asset, e.g. externally visible and affects business profitability or success
4	Serious but not complete damage to asset, e.g. affects business profitability or success, may be externally visible
3	Moderate damage or loss, e.g. affects internal business practices, causes increase in operational costs or reduction of revenue
2	Low damage or loss, e.g. affects internal business practices, cannot measure increase in costs
1	Minor or no change in asset

2. Menentukan Risk Level Dari Setiap Perhitungan Impact Rating

Impact rating * Probability rating = Risk Level				
Impact Rating Ranges		*	Probability Ranges	
High	10 -- 7		10 -- 7	
Medium	6 -- 4		6 -- 3	
Low	3 -- 0		3 -- 0	

	H	10	0	10	20	30	40	50	60	70	80	90	100
		9	0	9	18	27	36	45	54	63	72	81	90
		8	0	8	16	24	32	40	48	56	64	72	80
		7	0	7	14	21	28	35	42	49	56	63	70
Impact		6	0	6	12	18	24	30	36	42	48	54	60
	M	5	0	5	10	15	20	25	30	35	40	45	50
		4	0	4	8	12	16	20	24	28	32	36	40
		3	0	3	6	9	12	15	18	21	24	27	30
		2	0	2	4	6	8	10	12	14	16	18	20
	L	1	0	1	2	3	4	5	6	7	8	9	10
			0	1	2	3	4	5	6	7	8	9	10
			L					M					H
								Probability					

Overall Risk	Risk level
41-100	High
20-40	Medium
0-19	Low