# RANGKA KERJA KESELAMATAN TRANSAKSI BAGI PELANGGAN-PELAYAN BERASASKAN PERDAGANGAN ELEKTRONIK

(A SECURE TRANSACTION FRAMEWORK FOR CLIENT-SERVER BASED E-COMMERCE)

KETUA PROJEK:
ABDUL HANAN BIN ABDULLAH

RESEARCH VOTE NO:
72368

Jabatan Sistem Dan Komunikasi Komputer
Fakulti Sains Komputer Dan Sistem Maklumat
Universiti Teknologi Malaysia

2003

## ABSTRACT

Nowadays E-Commerce becomes more popular among online users for online trading. However, there are lots of intrusion issues on E-Commerce. Due to this main problem, this research project team "A Secure Transaction Framework for Client-Server Based E-Commerce" have been doing research specialized in three main components which are, website and server security, authorised data security on client-server and secure data transfer on network layer. Research on website and server security currently have been developing Web Document Integrity Detector (WebDID) to overcome intrusion in website and web server. WebDID is an alternative method to protect web server by detecting unauthorised changes in web and sending an intrusion alarm directly to web server administrator. Hashing function implementation have been using in this system as its cryptography algorithm. In authorised data security on client and server research, researcher mainly focused on data intrusions that can occur anytime without users awareness. Researchers have been developing a system using intrusion detector approach base on differential analysis. This method enable to differentiate between intruders and normal activities by counting the amount of coming system calls on each activities. Research in secure data transfer on network layer has been discussed in detail on the importance to have secure Internet protocol in E-Commerce environment. This particular research successfully produced a protocol to suit AES encryption algorithm in Internet Protocol Security (IPSec).

## ABSTRAK

E-Dagang kini kian popular dikalangan pengguna-pengguna atas talian untuk melakukan aktiviti-aktiviti perdagangan. Tetapi sejak kebelakangan ini kes-kes pencerobohan di dalam transaksi E-Dagang semakin membimbangkan. Oleh itu, projek penyelidikan "Rangka Kerja Keselamatan Transaksi bagi Pelayan-Pelanggan Berasaskan Perdagangan Elektronik" telah menjalankan beberapa penyelidikan untuk menyelesaikan permasalahan yang timbul di dalam persekitaran E-Dagang. Projek ini berkisar kepada tiga bidang utama iaitu keselamatan pelayan dan laman web, keselamatan data yang sah pada pelanggan mahupun pelayan dan keselamatan penghantaran data pada lapisan rangkaian. Penyelidikan mengenai keselamatan pelayan dan laman web menghasilkan satu sistem yang dikenali sebagai *Web Document Integrity Detector* (WebDID). WebDID merupakan satu kaedah alternatif untuk melindungi pelayan web dengan mengesan perubahan laman web dan memberi amaran pencerobohan kepada pentadbir pelayan web. Sistem ini telah mengimplementasikan algoritma kriptografi yang dikenali sebagai fungsi cincangan sebagai enjin utamanya. Penyelidikan tentang keselamatan data yang sah pada pelanggan mahupun pelayan berkisar kepada masalah pencerobohan data yang boleh berlaku ketika pengguna leka menggunakan aplikasi E-Dagang. Sistem yang dihasilkan menggunakan pendekatan sistem pengesan pencerobohan yang berasaskan analisis pembezaan. Kaedah ini mampu membezakan aktiviti penceroboh dengan aktiviti normal dengan mengira banyaknya jumlah *system call* yang dibangkitkan pada sesebuah aktiviti. Bagi penyelidikan keselamatan penghantaran data pada lapisan rangkaian pula, ia menjelaskan kepentingan mempunyai sebuah protokol keselamatan Internet (IPSec) yang dapat menyediakan kerahsiaan, ketulusan dan kesahihan pada maklumat yang dihantar di dalam persekitaran E-Dagang itu sendiri. Penyelidikan ini berjaya menghasilkan satu protokol pengurusan kekunci untuk membolehkan algoritma penyulitan AES diimplementasikan ke dalam IPSec.

# KANDUNGAN

## BAHAGIAN DUA



## BAHAGIAN TIGA

## BAHAGIAN LIMA

## BAB V PROTOKOL PENGURUSAN KEKUNCI KESELAMATAN INTERNET BAGI ALGORITMA PENYULITAN AES DALAM IPSEC

## BAHAGIAN ENAM



## BAHAGIAN TIGA BELAS

# SENARAI RAJAH

# SENARAI JADUAL

# BAB I

# PENDAHULUAN

Internet telah mengubah kebanyakan cara manusia berkomunikasi, berniaga, berhibur, belajar dan sebagainya. Jumlah penggunanya yang senantiasa meningkat menjadikan Internet sebagai satu medium perantaraan yang amat berpotensi untuk mempromosikan perniagaan, penjualan produk dan perkhidmatan, pertukaran surat-surat elektronik, dokumen serta penyebaran pelbagai jenis maklumat serta informasi.

Namun demikian, sejajar dengan perkembangan teknologi telekomunikasi terkini, Internet terdedah kepada pelbagai jenis ancaman penceroboh. Antaranya ialah penafian perkhidmatan, penipuan alamat IP, pengesanan dan kecurian paket data dan juga modifikasi pada data yang dihantar. Oleh itu, penyelidikan yang bertajuk "Rangka Kerja Keselamatan Transaksi Bagi Pelanggan-pelayan Berasaskan Perdagangan Elektronik" ini dijalankan bagi mendapatkan kaedah-kaedah pencegahan atau pemulihan yang dapat digunakan bagi menangani masalah tersebut. Projek ini dilaksanakan atas penajaan Kementerian Sains, Teknologi dan Alam Sekitar di bawah program IRPA (Intensification of Research in Priority Areas).

## 1.1 Matlamat Projek

Matlamat utama projek ini adalah untuk mendapatkan kaedah yang sesuai untuk menyediakan rangka kerja yang selamat bagi transaksi-transaksi di dalam aplikasi E-Dagang khususnya penglibatan antara pelanggan dan pelayan.

## 1.2 Objektif Projek

Berikut adalah objektif-objektif perlaksanaan projek ini:

(i) Menyelidik, menganalisis dan mengenalpasti isu-isu keselamatan semasa dan protokol pengangkutan di dalam aplikasi E-Dagang.

(ii) Menghasilkan satu rangka kerja untuk menyediakan transaksi yang selamat dan utuh berdasarkan penglibatan pelanggan dan pelayan di dalam E-Dagang.

(iii) Jika diberi masa yang mencukupi, satu model keselamatan yang mudah bagi menyediakan transaksi yang selamat di antara pelanggan dan pelayan di dalam E-Dagang akan dapat dihasilkan. Model ini melibatkan persekitaran antara pengguna-ke-perniagaan dan perniagaan-ke-perniagaan dan ianya dibangunkan untuk tujuan pengujian rangka kerja yang dihasilkan.

## 1.3 Skop Projek

Perlaksanaan projek ini mencakupi skop-skop berikut:

(i) Menggunakan sistem pengoperasian Linux sebagai pelantar pembinaan sistem

(ii) Pakej perisian firewall FWTK digunakan sebagai perisian utama pembinaan sistem firewall asas disamping IPFWADM dan IPCHAINS

(iii) Kriteria Pengiktirafan Produk ICSA FWPD digunakan sebagai garis panduan dalam pembinaan sistem asas dan modul-modul keselamatannya.

## 1.4 Bidang Penyelidikan

Penyelidikan ini dijalankan berkisar kepada tiga bahagian yang utama iaitu; keselamatan pelayan dan laman web, keselamatan data yang sah pada pelayan dan pelanggan serta keselamatan penghantaran data pada lapisan rangkaian.

### 1.4.1 Laman Web

Pada masa kini, penggunaan laman web menjadi semakin penting untuk berkomunikasi dan menyebarkan maklumat. Laman web atau World Wide Web (WWW) begitu sinonim dengan E-Dagang yang merupakan salah satu perkhidmatan dan aplikasi elektronik dan menggunakan Internet sebagai perantaraan. Capaian yang meluas dan efektif adalah

antara sebab-sebab mengapa laman web menjadi pilihan dalam menjalankan perniagaan.

Laman web ialah satu maklumat teks tinggi dan juga merupakan satu sistem komunikasi. Ia mengandungi teks, imej, suara dan video serta menggunakan jaringan komputer berinternet bersama komunikasi data. Operasinya bergantung kepada model pelayan-pelanggan (client-server). Pelayan web boleh mencapai pelbagai protokol maklumat media tinggi menggunakan skema pengalamatan. Teks tinggi web ditulis menggunakan *Hypertext Markup Language* (HTML) yang merupakan salah satu aplikasi *Standard Generalized Markup Language* (SGML). SGML adalah piawaian antarabangasa (ISO 8879) bagi pemprosesan teks maklumat. Secara ringkasnya,

| Web = Teks tinggi + Multimedia + Jaringan |
|---|

Ini di mana, teks tinggi adalah asas bagi pautan bersepadu, multimedia mempersembahkan data dan maklumat dalam pelbagai format dan bentuk sama ada dilihat atau didengar dan jaringan adalah capaian secara global.

### 1.4.2 Sistem Pengesanan Pencerobohan

Pengesanan pencerobohan adalah suatu seni daripada pengesanan ketidakpatutan, kesalahan ataupun aktiviti yang tidak lazim terjadi. Sistem pengesanan pencerobohan (*Intrusion Detection System*) yang beroperasi pada sebuah

host untuk mengesan aktiviti-aktiviti pencerobohan pada host tersebut dikenali sebagai Sistem Pengesanan Pencerobohan Berasaskan Hos (*Host-based Intrusion Detection System*), dan sistem pengesanan pencerobohan yang beroperasi pada jaringan dan aliran data disebut sebagai Sistem Pengesanan Pencerobohan Berasaskan Jaringan (*Network-based Intrusion Detection System*) (lehman 2000).

Pengesan pencerobohan berdasarkan hos melibatkan bahagian perisian pada sistem untuk dipantau. Pemuatan perisian menggunakan fail log dan sistem agen dalam melakukan ubah-suai sebagai sumber data. Sebaliknya, sistem pengesanan pencerobohan berdasarkan rangkaian memantau trafik pada segmen rangkaian sebagai sumber data. Pengesanan pencerobohan berdasarkan hos tidak hanya mencari data sumber pada trafik komunikasi di dalam atau di luar dari sebuah komputer tunggal, tetapi juga melakukan pemeriksaan pada keutuhan sistem fail dan melihat proses yang mencurigakan. Untuk memperoleh pencapaian yang lengkap pada sebuah lokasi pengesanan pencerobohan komputer, terdapat dua kelas utama daripada perisian pengesanan pencerobohan berasaskan host (Zirkle & Virginia, 2000).

Sebuah kaedah pengesanan pencerobohan baru berasaskan kepada analisis pembezaan (*Discriminant Analysis*) telah dikenal pasti oleh penyelidik Jepun dalam agen promosi teknologi dan maklumat (*Information-technology Promotion Agent*) (Midori, *et.al* 2001). Analisis pembezaan adalah sebuah teknik dalam statistikal yang digunakan untuk membezakan antara dua kelompok atau populasi yang saling bertindih. Mereka menyelidiki sebuah kaedah baru untuk mengesan pencerobohan berasaskan kepada “system call” semasa aktiviti seorang pengguna rangkaian pada

sebuah mesin hos. Kaedah ini berusaha untuk memisahkan pencerobohan dari aktiviti normal menggunakan analisis pembezaan, iaitu salah satu jenis analisis "multivariate" (*Multivariate Analysis*). Analisis "multivariate" adalah salah satu teknik yang digunakan untuk mencari pola yang saling berkaitan di antara beberapa pembolehubah secara berterusan.

### 1.4.3 IPsec (Internet Protocol Security)

IPsec adalah satu protocol keselamatan IP yang telah dibina oleh sekumpulan penyelidik dari IETF (Internet Engineering Task Force). IETF adalah satu organisasi yang bertanggungjawab menyelaraskan pelbagai aktiviti dalam Internet. IPSec menggunakan pelbagai algoritma kriptografi untuk menghasilkan proses penyulitan dan pengesahan yang berkesan. Ia berfungsi untuk menjaga ketelusan dan kerahsiaan dalam proses penghantaran data. Binaan IPSec direka secara khusus bagi menepati struktur binaan IPV4 dan IPV6. Ini akan memudahkan lagi proses aplikasi IPSec untuk kegunaan masa sekarang dan juga pada masa akan datang.

IPsec mempunyai beberapa kelebihan yang menjadikannya sesuai untuk diadaptasikan ke dalam pelbagai sistem. IPSec sangat fleksibel di mana ia membenarkan penggunaan pelbagai jenis algoritma kriptografi. Ia juga membenarkan pengubahan dilakukan pada polisi dan mekanisma yang digunakan. Ini adalah bertujuan untuk menyesuaikan keadaan persekitaran semasa dengan sistem yang dibina.

Ini bermakna, walaupun IPSec telah menetapkan algoritma-algoritma piawai yang mesti digunakan, namun senibinanya yang fleksibel membolehkan algoritma lain diimplemenkan di dalam IPSec. Oleh itu, menjadi tanggungjawab pereka bagi rekabentuk sistem yang menggunakan algoritma baru, menakrifkan beberapa rangka kerja, mekanisma dan polisi yang mesti dipatuhi, tetapi ia mesti dibuat berdasarkan ketetapan yang telah dinyatakan oleh IETF.

Penyelidikan berkenaan keselamatan pelayan dan laman web menghasilkan satu sistem yang dinamakan WebDID (Web Document Integrity Detector). Kajian berkenaan keselamatan data yang sah pada pelanggan dan pelayan menghasilkan sistem pengesanan pencerobohan berasaskan analisis perbezaan. Manakala penyelidikan keselamatan data pada lapisan rangkaian pula menjelaskan tentang kepentingan IPsec untuk menyediakan kerahsiaan, ketulusan dan kesahihan pada maklumat yang dihantar dalam persekitaran E-Dagang itu sendiri.

Secara keseluruhannya, tesis ini menceritakan tentang permasalahan keselamatan dunia Internet dalam persekitaran E-Dagang sehingga menjurus terlaksananya projek ini sebagai antara jalan penyelesaian yang boleh digunakan untuk menanganinya. Kajian latarbelakang, metodologi perlaksanaan projek, dan bidang-bidang kajian diterangkan pada setiap bab yang berkaitan.

# BAB II

# KAJIAN LITERATUR

Perkembangan teknologi maklumat yang berkembang pesat menyebabkan kita tidak boleh lagi memandang ke belakang sebagai panduan masa hadapan lagi. Semasa berhadapan dengan desakan pasaran yang dibawa oleh E-Dagang yang semakin tinggi persaingannya, syarikat tidak wajar terus dipandu oleh jejak-jejak sejarah atau masih terus dikekang oleh status quo masing-masing. Syarikat akan mendapati bahawa penyelesaian lama tidak lagi berkesan dengan masalah baru. Parameter perniagaan telah berubah, begitu juga dengan risiko dan kaedah pembayaran.

E-Dagang (E-Commerce) menjadi semakin kritikal dalam tiga dimensi yang saling berkaitan: interaksi pelanggan dengan peniaga, interaksi intra-peniaga, dan interaksi peniaga dengan peniaga. Dalam dimensi pelanggan dengan peniaga, E-Dagang membenarkan pelanggan mengimbasi produk yang hendak dipesan. Di sini konsep ekonomi digital diperlukan, iaitu pelanggan memesan sesuatu produk yang belum dibuat. Produk hanya akan dibuat sebaik sahaja menerima pesanan dan ciri-

ciri yang dikehendaki pelanggan. Dalam dimensi ini juga pelanggan mempunyai kawalan yang lebih besar terhadap proses pembuatan dan penghantaran.

E-Dagang juga merupakan pemangkin kepada perubahan dalam fungsian organisasi dalaman, seperti yang disaksikan oleh perkembangan mendadak intranet. Ia menghasilkan model organisasi yang dahulunya berasaskan organisasi arahan dan kawalan berhierarki kepada organisasi berasaskan maklumat. Kemunculan struktur berbentuk tekno organisasi melibatkan perubahan dalam tanggungjawab pengurusan, pengaliran komunikasi dan maklumat, dan struktur kumpulan kerja.

Ekonomi digital seperti yang ditunjukkan oleh E-Dagang turut memberi kesan kepada interaksi peniaga dengan peniaga. Perdagangan jenis baru ini melicinkan organisasi berbentuk rangkaian di mana firma-firma kecil bergantung kepada syarikat rakan kongsi untuk bekalan komponen dan pengedaran produk bagi memenuhi permintaan pelanggan secara lebih efektif.

Penggunaan intranet dan ekstranet menjadi kritikal dalam situasi sebegini. Dalam menggunakan teknologi intranet dan ekstranet ini, penyelesaian pengurusan perhubungan dipanggil pengurusan bersepadu atau pengurusan pengembangan rantaian bekalan. Konsep penguruan bersepadu ini merangkumi pengendalian rantaian perhubungan dengan pelanggan, pekerja, pembekal, rakan niaga, pengedar, dan juga pesaing.

Di dalam pelaksanaanya E-Dagang memberikan beberapa keuntungan yang dapat diperoleh dengan penerapannya dalam perniagaan, diantaranya:

- Masa yang diperlukan untuk melakukan proses terhadap dokumen lebih cepat. Setiap tahap pemrosesan dokumen diawali dari pencarian, pemprosesan, penyimpanan, penghantaran data dan sebagainya diserahkan kepada komputer.
- Tidak memerlukan ruang untuk penyimpanan dokumen yang besar. Sekumpulan fail atau dokumentasi dapat digantikan oleh sebuah komputer dengan kapasiti penyimpanan data yang besar.
- Transparansi pemprosesan dokumen dapat ditingkatkan sehingga mengurangi kesempatan adanya manipulasi data dan maklumat.
- Tingkat kesalahan pemprosesan dokumen dapat dikurangi karena pemprosesan dilakukan oleh komputer yang memiliki tingkat ketelitian yang cukup tinggi. Dengan demikian waktu pemprosesan akan lebih cepat.

Penurunan kepercayaan terhadap aplikasi-aplikasi perniagaan pada Internet menyebabkan para pelaksana perniagaan dan pelanggan, untuk tidak lagi menggunakan Internet pada masa sekarang dan kembali kepada metode tradisional untuk melakukan transaksi perniagaan. Kehilangan kepercayaan ini disebabkan oleh masalah yang sedia ada pada sistem p-elektronil dan serangan-serangan yang dilakukan para penceroboh terhadap laman web Internet dan penyalahgunaan terhadap data persendirian pelanggan. Misalnya, penceroboh menuntut sebuah tebusan dari sebuah laman web E-Dagang yang mana pihak peniaga tidak boleh mendedahkan maklumat kad kredit pelanggannya kepada umum. Konflik yang terjadi antara kemudahan yang ada pada Internet dengan keselamatan maklumat, adalah menjadi hal yang utama dalam aplikasi E-Dagang.

## 1.1 Masalah Sedia Ada Pada Sistem E-Dagang Konvensional

### 1.1.1 Kurang Fleksibel

Secara tradisional, sebuah sistem E-Dagang yang baik dapat di bagi menjadi tiga lapisan: sebuah lapisan persembahan yang memiliki antar muka dengan pelanggan, sebuah lapisan perniagaan yang menangani semua perniagaan logik dan sebuah lapisan data yang terlihat setelah aplikasi dihubungkan dengan data. Secara teori, ketiga lapisan ini harus bebas dan dapat digantikan tanpa dipengaruhi oleh satu sama lain. Walau bagaimanapun, banyak alat bantu yang boleh dipakai untuk membangun halaman web yang dinamis seperti JSP, PHP, dan ASP perniagaan logik yang bercampur baur dengan lapisan persembahan. Tidak ada kejelasan pemisahan antara kod perniagaan logik dan kod persembahan. Banyak aplikasi E-Dagang yang telah direka bentuk dibawah tekanan dan paksaan, iaitu TIM (*Time-to-Market*), dan keperluan seperti boleh digunakan kembali dan fleksibiliti yang rendah pada senarai keutamaan. Hasil ini patut dipertimbangkan dan seterusnya di buat selama urutan pemeliharaan untuk mempertemukan keperluan perniagaan yang sesungguhnya.

### 1.1.2 Tidak ada Kongsi Perkhidmatan

Banyak daripada sistem E-Dagang mulanya direka bentuk buat pelanggan untuk membeli atau melayani dengan membaca antara muka dari pelanggan, iaitu pada halaman web di dalam kebanyakan kes. Lapisan persembahan adalah kebanyakan lapisan yang biasa digunakan untuk berkongsi dengan dunia luar pada

berbagai aplikasi web. Meskipun terdapat ejen perisian pintar yang wujud untuk menggali maklumat dari halaman web, mereka juga memerlukan seorang manusia untuk memahami konteks daripada apa-apa yang ditampilkan. Itu tidak akan menjadi sebuah tugas yang mudah, sebagai contoh, dalam menulis sebuah program untuk memeriksa harga sebuah model televisyen tertentu dengan pembuat yang berbeza pada internet, kerana kebanyakan aplikasi web tidak di reka bentuk untuk menghadirkan sebuah bentuk program yang dapat dibaca dan aplikasi tidak dapat difahami secara mudah satu sama lain kecuali sebuah persetujuan mula yang wujud.

### 1.1.3 Tidak ada Keutuhan (*Integrity*)

Menurut sebuah studi yang dilakukan oleh Gartner, pengembang menghabiskan 68% masa mereka untuk mempererat aplikasi secara bersama. Secara nyata, kemampuan untuk menggabungkan dengan aplikasi lain sama ada dalam usaha niaga atau dengan sejawat perniagaan lain menjadi sebuah masalah penting di dalam dunia perniagaan yang dinamik. Peristiwa seperti menggabungan, menukar sejawat perniagaan, atau menubuhkannya kembali semua yang diperlukan di dalam mengintegrasikan aplikasi yang telah dibeli atau dibina baik secara moden atau peninggalan, yang telah ditulis dalam berbagai bahasa yang berbeza, berjalan pada platform dan lokasi yang berbeza pula. Sekarang masih terdapat beberapa mekanisme pembuatan aplikasi web untuk menggantikan maklumat Internet yang luas, tidak di katakan adanya keutuhan alamat yang akan datang dengan sejawat perniagaan dan pelanggan yang baru ataupun yang sedia ada.

## 1.2 Ancaman Terhadap E-Dagang

Model pelayan-pelanggan yang piawai memiliki tiga komponen utama aitu: sistem server, rangkaian, dan sistem klien. Pada masa lalu, sistem pelayan adalah sebuah kerangka utama yang menjalankan sistem operasi seperti MVS, VM, VMS atau Unix. Dan sekarang Windows NT dan Windows 2000 juga sudah digunakan sebagai sistem operasi pada beberapa server. Komponen rangkaian juga termasuk rangkaian bisnes dalaman antara bisnes dan pengguna melalui berbagai seperti ISP dan rangkaian dalaman bagi pengguna itu sendiri.

### 1.2.1 Komponen Keselamatan pada E-Dagang

Strategi keselamatan E-Dagang berkisar kepada dua pokok persoalan: iaitu perlindungan terhadap keutuhan rangkaian perniagaan dan sistem dalamannya; dan menyempurnakan keselamatan tranksaksi antara pengguna dan peniaga. Alat bantu perniagaan utama yang digunakan sebagai pelindung terhadap jaringan internal adalah *firewall*. *Firewall* adalah sebuah sistem perisian dan perkakasan yang hanya membolehkan pegguna luar dengan kriteria tertentu untuk mencapai sesebuah rangkaian yang peribadi (Kalakota dan Whinston, 1999). *Firewall* telah menjadi titik utama dalam pertahanan pada arkitek keselamatan perniagaan. Walau bagaimanapun, *firewall* mestilah menjadi sebuah bahagian pintar dari sebuah infrastruktur keselamatan dalam melakukan sesuatu perniagaan.

### 1.2.2 Virus

Virus adalah merupakan kebanyakan ancaman yang terjadi pada sisi sistem klien. Virus menjadi efektif disebabkan oleh kerana pembinaan yang tidak selamat terhadap sistem klien (PC/Mac). Untuk menjadikannya efektif, virus memerlukan "keistimewaan" yang ada pada sesebuah sistem. Amnya, skema capaian keistmewaan dapat dilihat pada sistem operasi Unix, VMS dan sistem operasi lainnya yang menghalang sesuatu viirus membuat kerosakan dan kehancuran pada seluruh sistem yang ada. Ianya hanya membuat kerosakan pada sebuah fail tertentu bagi seseorang pengguna.

### 1.2.3 Trojan Horse

BackOrrife, Netbus, BO2K merupakan beberapa alat bantu penggodam yang mengizinkan seorang pengguna jarak jauh untuk mengendalikan, menguji, dan melakukan beberapa pemantauan terhadap maklumat yang terdapat pada PC tujuan.

# BAB III

## PENGESAN KEUTUHAN BAGI LAMAN WEB

Kajian latar belakang yang dijalankan lebih tertumpu kepada proses pencarian maklumat untuk mengkaji ancaman kepada laman web, penelitian terhadap aplikasi yang telah ada di pasaran dan memahami algoritma cincang yang terkini. Algoritma cincang yang dikaji hanya RIPEMD-160 (Dobbertin *et, al*, 1996, Dobbertin *et,al,* 1997 dan Robshaw, 1996) dan SHA-1 (Burroes *et,al*, 1995) kerana menghasilkan nilai cincang 160-bit. Manakala aplikasi adalah Tripwire (Tripwire Inc, 2002), WebAlarm (E-Lock Corporation (2001) dan WebAgain (Lockstep System 2001). Maklumat-maklumat diperolehi daripada buku rujukan, artikel dan kertas kerja.

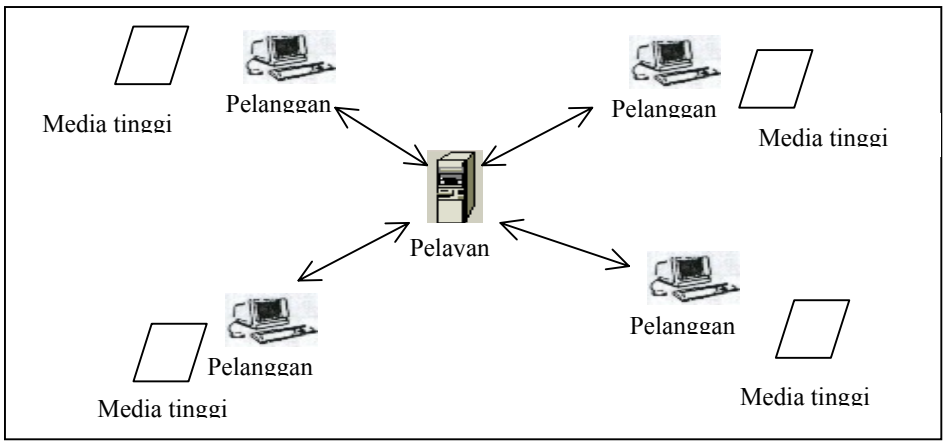
## 3.1 Laman Web

Pada masa kini, penggunaan laman web menjadi semakin penting untuk berkomunikasi dan menyebarkan maklumat. Laman web atau World Wide Web (WWW) begitu sinonim dengan e-perdagangan yang merupakan salah satu perkhidmatan dan aplikasi elektronik dan mengunakan internet sebagai perantaraan. Capaian yang meluas dan efektif adalah antara sebab-sebab mengapa laman web menjadi pilihan dalam menjalankan perniagaan.

Laman web ialah satu maklumat teks tinggi dan juga merupakan satu sistem komunikasi. Ia mengandungi teks, imej, suara dan video serta menggunakan jaringan komputer berinternet bersama komunikasi data. Operasinya bergantung kepada model pelayan pelanggan (client server). Pelayan web boleh mencapai pelbagai protokol maklumat media tinggi menggunakan skema pengalamatan. Teks tinggi web ditulis menggunakan *Hypertext Markup Language* (HTML) yang merupakan salah satu aplikasi *Standard Generalized Markup Language* (SGML). SGML adalah piawaian antarabangasa (ISO 8879) bagi pemprosesan teks maklumat. Secara ringkasnya,

| Web = Teks tinggi + Multimedia + Jaringan |
|---|

Dimana, teks tinggi adalah asas bagi pautan bersepadu, multimedia mempersembahkan data dan maklumat dalam pelbagai format dan bentuk sama ada dilihat atau didengar dan jaringan adalah capaian secara global. Rajah 2.1 adalah gambaran organisasi web.




**Rajah 3.1: Organisasi web**

### 3.1.1 Ancaman Terhadap Laman Web

Pelayan web digunakan untuk memaparkan maklumat dan ribuan pengguna akan mencapainya melalui Internet. Walaubagaiamanapun, transaksi yang dijalankan di dalam internet terdedah kepada pelbagai jenis masalah keselamatan seperti pencerobohan. Masalah ini mudah dan cepat disebarkan kerana WWW merupakan salah satu sistem berpersekitaran teragih (Soh *et,al*, 1998).

Kes pencerobohan dianggap serius, memandangkan jumlah kes pencerobohan laman web meningkat dari hari ke sehari, dimana secara purata 10 kes pencerobohan berlaku dalam sehari (Hollander, 1999).Berdasarkan laporan *CSI/FBI Computer Crime and Security Survey* bagi tahun 2002, jumlah pencerobohan laman web pada tahun 2001 adalah sebanyak 30,388 kes berbanding 7,629 pada tahun 2000 yang meliputi semua jenis domain. Sebuah laman web iaitu attrition.org memantau aktiviti pencerobohan dan memaparkan senarai laman web yang telah diceroboh.

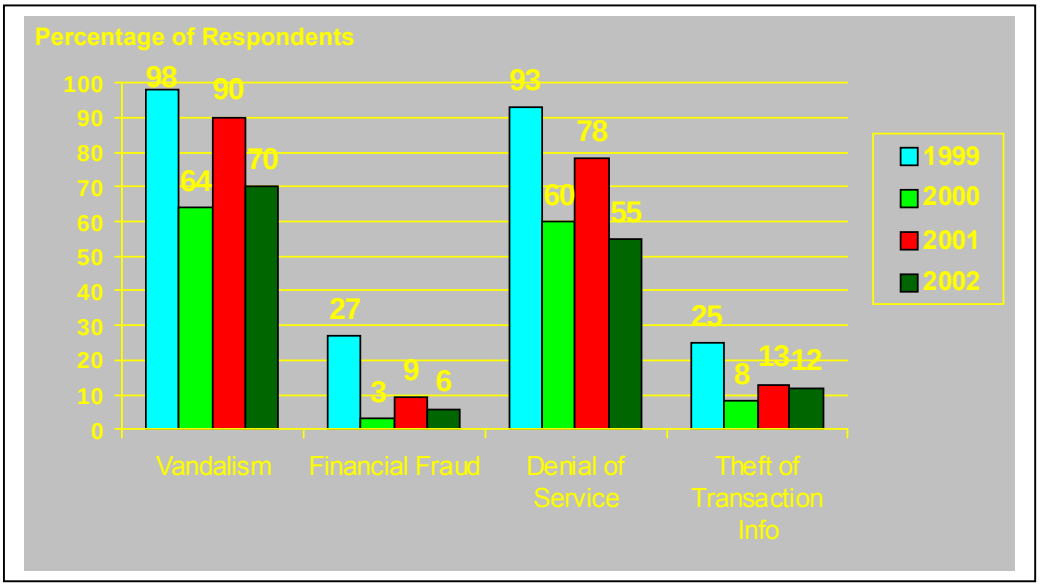
Pencerobohan akan memberi risiko kepada pelayan web (Soh *et,al*, 1998) dimana :

i. Fail-fail sulit dan rahsia yang disimpan di pelayan berkemungkinan jatuh ke tangan pihak yang tidak berhak.

ii. Berlaku pemintasan terhadap pelayan untuk memperolehi maklumat sulit dan rahsia seperti nombor kad kredit.

iii. Maklumat mesin berjarak jauh seperti fail /etc/passwd.akan cuba dibebanturun oleh pengondam dan digunakan untuk memasuki sistem.

iv. Wujud pepijat bagi membenarkan orang luar menjalankan arahan pada sistem jarak jauh.




**Rajah 3.2: Jenis-jenis ancaman ke atas laman web**

Laman web yang telah dirosakkan pula akan mendedahkan pengguna kepada maklumat yang tidak benar atau salah sehingga ia dikesan dan dibaikpulih. Masalah pencerobohan laman web ini akan memberi kesan kepada organisai seperti institusi perbankan (Hollander, 1999).

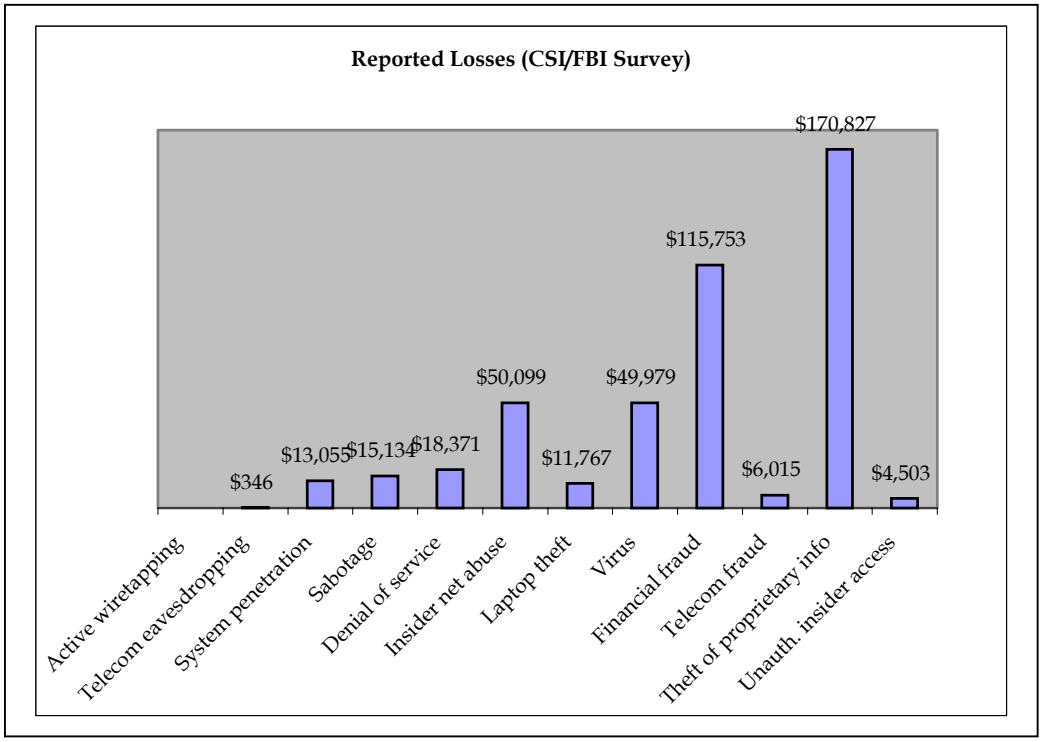
i. Pihak yang tidak bertanggungjawab akan memperolehi maklumat penting seperti nombor kad kredit untuk melakukan aktiviti jenayah yang lain.

ii. Pencerobohan web memberi ancaman kepada perniagaan yang dilakukan secara atas talian. Ini akan memberi kesan kepada reputasi dan keolehan sesebuah organisasi.

iii. Pengondam melakukan pencerobohan bertujuan untuk mendapatkan publisiti dengan menguji tahap keselamatan yang digunakan.

iv. Kemusnahan akibat perncerobohan ini tidak seimbang. Kemusnahan boleh berlaku daripada kehilangan kepercayaan pengguna sehingga kehilangan pendapatan. Rajah 3.3 menunjukkan jumlah kerugian yang dialami oleh pengguna mengikut jenis ancaman. Pengguna hilang kepercayaan akan keselamatan perniagaan elektronik.

v. Peruncit elektronik (*E-retailer*) pula akan hilang pertimbangan perlindungan jika pengguna merasai sesuatu perniagaan elektronik itu tidak selamat.




**Rajah 3.3: Graf Jumlah Kerugian Mengikut Jenis Ancaman**

### 3.1.2 Kelemahan Penyelesaian Sedia Ada

Terdapat tiga penyelesaian yang sering digunakan untuk melindungi pelayan dan mengesan pencerobohan. Walaubagaimanapun, penyelesaian ini tidak memadai kerana pencerobohan tetap berlaku. Atas sebab-sebab kelemahan setiap penyelesaian, organisasi mencari alternatif lain untuk mempertingkatkan keselamatan pelayan web.

*i. Firewall*

Firewall merupakan perkakasan keselamatan yang utama pada masa kini, tetapi ia tidak dapat mencegah pencerobohan laman web. Firewall

berfungsi untuk mencapai paket komunikasi yang masuk dan menghalang masuk jika paket didapati berniat jahat. Bagi membolehkan pelayan web beroperasi, pot 80 sentiasa dibuka bagi membolehkan protocol HTTP menggunakannya. Pada kebiasaannya, firewall tidak mengimbas paket HTTP yang masuk. Oleh sebab itu, paket yang berniat jahat tidak dapat dikenalpasti dan komunikasi adalah sah. Maka, ancaman HTTP tidak dapat dikesan oleh firewall.

ii. *Network-based Intrusion Detection System* (NIDS)

NIDS dikatakan boleh mengesan beberapa ancaman berasaskan HTTP. Walaubagaimanapun, NIDS lambat mengesan ancaman. Ini kerana NIDS hanya mendengar paket dalam talian dan tidak menghalang permindahan paket. Dalam kes ini, paket akan sampai di destinasinya and diproses sebelum ia diterjemahkan oleh NIDS.

iii. *Host-based Intrusion Detection* (HIDS)

HIDS turut mempunyai masalah yang sama dengan NIDS. Kelewatan mengesan ancaman dalam semua kes secara virtual sehingga ancaman telah dilakukan. HIDS mengimbas log dari masa ke semasa dan membina tempoh masa jika tiada pemprosesan. Pada masa ini, pencerobohan cuba dilakukan tanpa dikesan, walaupun mereka telah dilog oleh komponen sistem.

### 3.1.3 Cara Pencerobahan Laman Web Dilakukan

Pengodam melakukan pencerobohan laman web sama ada dengan tujuan atau untuk mencuba-cuba sistem keselamatan yang digunakan. Pengondam mengambil kesempatan atas kelemahan *firewall* yang menghalang hanya pencerobohan ke atas sistem pengoperasian pelayan. Ada sesetengah sistem pengoperasian pelayan melaksanakan perisian pelayan web seperti Internet Information Server (IIS), Apache, WebSphere dan sebagainya. Pelayan seperti ini memaksa *Firewall* membenarkan pengguna untuk mencapai pot 80 dan menggunakan aplikasi berasaskan web. Melalui cara ini, pengondam mampu memperolehi maklumat yang berguna daripada perisian pelayan web, maka firewall tidak lagi berfungsi.

Oleh sebab itu, pengondam selalunya akan menggunakan cara pencerobohan yang melibatkan aplikasi berbanding sistem pengoperaasian. Kajian mendapati terdapat beberapa cara atau teknik yang digunakan oleh pengodam untuk memecah masuk ke dalam sistem dan laman web (Sturat *et,al*, 2003).

**i. Pencerobohan URL**

Pencerobohan URL merupakan teknik pencerobohan melalui URL. Pengondam mengubah satu URL supaya ia tidak dapat dikesan oleh pelayan web dan mengelirukan. URL adalah singkatan daripada *Uniform Resource Locator* yang merupakan satu mesej yang dihantar kepada pelyaan web. Ia mengandungi maklumat berkaitan laman web atau sumber yang diminta oleh pengimbas web *(web browser)*.

Sebagai contoh, pengguna hendak melihat senarai filem secara atas talian. Pengguna akan menaip “http://www.apple.com/trailers/” pada pengimbas (*browser*) atau mengikut pautan ke lokasi. Lokasi ini adalah URL yang mungkin mengandungi data seperti nama pengguna atau ID pengguna. Maklumat ini kebiasaannya akan muncul dalam URL selepas tanda tanya.

Contoh lain seperti perniagaan secara atas talian yang menggunakan laman “hhtp://www.somesite.com”. URL bagi laman ini mengandungi maklumat harga bagi barangan. Menghantar URL bersama-sama maklumat harga dilakukan untuk mengurangkan permintaan daripada aplikasi bagi memperolehi harga barangan daripada pangkalan data. Disamping itu, pengimbas pengguna dapat menghantar semula harga tersebut kepada pelayan setiap kali pengguna mencapai laman tersebut.


**Rajah 3.4: Penggodam Mengubah URL**

Walaubagaimanapun, pengondam akan cuba mengubah bahagian URL yang mengandungi harga. Pengondam kaan menukar harga barangan daripada dua ratus ringgit kepada dua ringgit. Sesetengah pelayan tidak menyedari kerana URL tersebut mempunyai set dan bilangan aksara yang sama dan tidak nampak kecacatannya. Rajah 3.4 menunjukkan contoh pengondam mengubah URL.

**ii. Rampasan Sidang (Session Hijacking)**

Rampasan sidang adalah teknik kedua yang biasa digunakan oleh pengondam untuk menceroboh laman web. Ia merupakan teknik penipuan identiti secara atas talian.

Internet merupakan contoh sistem tanpa penyambungan yang memecahkan komunikasi antara pihak lain kepada transmisi *inbound* dan *outbound* . Sebagai contoh, pengguna beban naik satu laman web yang nmengandungi dua imej iatu “Imej A” dan Imej B”. Pengimbas web akan membina dua penyambungan ke pelayan dimana imej akan diletakkan., satu penyambungan untuk satu imej. Pelayan tidak mengetahui kedua-dua imej tersebut adalah daripada pengimbas yang sama.

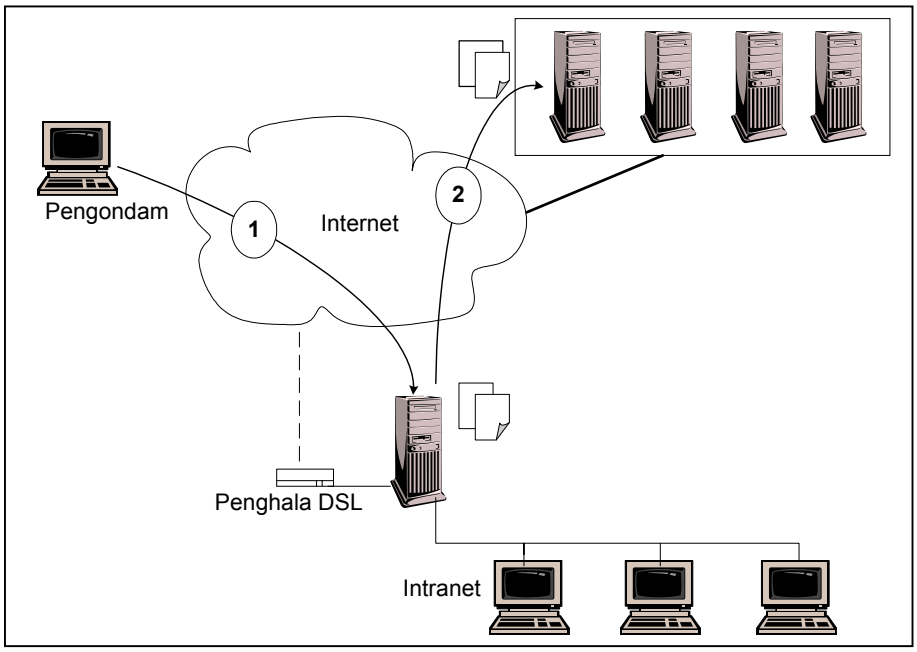
Untuk memautkan permintaan pada sebelah pelayan secara berasingan, kebanyakkan pelayan akan menanam (*embed*) sidang ID sebagai sebahagian daripada URL, sebahagian daripada cookie, atau menerusi pelbagai metod lain. Sidang ID adalah sebagi tanda (tags) kepada permintaan pengguna.

Dengan cara ini, pelayan web dapat mengenalpasti sama ada permintaan daripada penguna tertentu adalah berbentuk multiple, simultaneous or successive.

### iii. **Contengan Siber (*Cyber Graffiti*)**

Teknik contengan siber pula merupakan teknik pencerobohan yang dilakukan ke atas laman web. Pengondam akan cuba mengexploitasi pembinaan (prefabricated) untuk mengawal pentadbiran sistem yang menjadi sasaran dan menggantikan laman web dengan laman web versi mereka.

Pengodam akan menceroboh pelayan proxy sesebuah organisasi menerusi internet dan meletakkan laman web mereka pada direktori hentian (*staging*) web. Pengubahsuaian laman web pada kawasan hentian akan merosakkan laman web setelah replikasi dilaksana secara automatik. Rajah 3.5 menunjukkan bagaimana contengan siber dilakukan.




**Rajah 3.5: Contengan Siber**

### 3.1.4 Penyelesaian

Untuk mengatasi kelemahan pada kaedah yang sedia ada, satu kaedah yang dikenali sebagai Penilaian Keutuhan (*integrity assessment*) dicadangkan sebagai alternatif tambahan terutama kepada pelayan web. Kaedah ini digunakan untuk memastikan keutuhan sesuatu dokumen tidak diubahsuai. Ia menyimpan kod cincang bagi laman web yang mewakili kandungan web. Dari masa ke semasa, sistem akan membuat perbandingan kod cincang laman web semasa dengan kod cincang yang telah disimpan. Jika laman web telah diubah, kod cincang tidak sama. Selepas pengesanan, sistem keutuhan kan memberitahu pentadbir, dan ada yang akan menggantikan dengan laman web yang asal.

Walaubagaimanapun, kaedah ini memerlukan kekerapan dalam melakukan perbandingan kod cincang. Jika tidak pengesanan ancaman dibuat dalam masa yang panjang. Dalam jangka masa ini, pengguna akan dipaparkan dengan laman web yang tidak benar. Kaedah ini hanya sesuai digunakan hanya untuk laman web yang statik. Laman web yang dinamik yang sentiasa berubah memerlukan kekerapan yang lebih berbanding laman web statik.

Terdapat beberapa aplikasi penilaian keutuhan yang telah ada di pasaran yang juga menyerupai projek ini. Kajian telah dilakukan dan mendapati cara perlaksanaan setiap aplikasi hampir serupa, tetapi cara pembangunan sistem berbeza terutama pada bahasa pengaturcaraan, fungsi cincangan dan pangkalan data yang digunakan. Perisian-perisian tersebut adalah WebAlarm (E-Lock, 2001), Tripwire (Tripwire, 2002) dan WebAgain (Lockstep System, 2001).

### 3.1.4.1 Fungsi Cincang

Cincang dalam konteks keselamatan merupakan satu nilai "rumusan" atau "tag" yang dijanakan daripada satu mesej dengan menggunakan peraturan-peraturan matematik atau algoritma. Manakala cerna mesej merupakan satu jujukan nombor khas yang dikira daripada data input pengguna dengan mengunakan algoritma cincang (Mohd Aizaini *et,al*, 2000).

Nilai cincang dihasilkan menggunakan fungsi *H*,

$$h = H(M)$$

dimana h mewakili pembolehubah panjang mesej *dan H(M)* mewakili panjang tetap nilai cincang. Fungsi cincangan itu sendiri bukanlah suatu rahsia. Oleh itu, nilai cincang perlulah dilindungi. Untuk mendapatkan hasil yang baik, nilai cincang, *H* perlulah memenuhi  kriteria (Stalling, 1996) berikut:

i) Diimplemen pada pelbagai saiz mesej.

ii) Menghasilkan nilai cincang yang tetap.

iii) Mudah untuk dikira.

iv) Bergantung kepada mesej.

v) Sukar untuk terbalikan proses cincangan.

- **Simbol Logik**

Di bawah ini adalah simbol-simbol logik yang digunakan dalam fungsi yang digunakan bagi algoritma cincang.

| | | |
|---|---|---|
| X ^ Y | = | logik "and" bagi X dan Y |
| X / Y | = | logik "inclusive-or" bagi X dan Y |
| X ⊕ Y | = | logik "exclusive-or" bagi X dan Y |
| ~X | = | "complement" bagi X |

- **Pelampiran mesej**

Mesej atau fail data dijadikan input bagi algoritma cincangan untuk menghasilkan cerna mesej. Mesej dan fail data adalah berbentuk rentetan bit.

Panjang mesej ialah bilangan bit di dalam mesej. Tujuan pelampiran mesej adalah untuk menjadikan panjang mesej dalam gandaan 512. Pelampiran dilakukan ke atas mesej yang mempunyai panjang kurang daripada $2^{64}$ dan sebelum proses cincangan (Burroes *et,al*, 1995). Langkah-langkah pelampiran adalah seperti berikut:

i) Pelampiran "1" di sebelah kanan.

Sebagai contoh, mesej asal "01010000" menjadi "01010000 1".

ii) Pelampiran "0".

Bilangan "0" bergantung kepada panjang mesej asal. 64 bit terakhir daripada blok 512-bit dikhaskan untuk panjang mesej asal. Sebagai contoh,

01100001 01100010 01100011 01100100 01100101.

Setelah langkah i) dilaksanakan, mesej akan menjadi

01100001 01100010 01100011 01100100 01100101 1.

Oleh kerana panjang mesej ialah 40 dan bilangan bit di atas adalah 41, maka 407 "0" dilampirkan dan menjadikan jumlah bit ialah 448.

Mesej akan menjadi seperti dibawah ( bentuk perenambelasan):

61626364 65800000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000

iii) Pelampiran panjang mesej asal.

Bilangan bit atau panjang mesej asal diwakili dengan dua kata. Jika panjang mesej asal adalah kurang daripada $2^{23}$, maka kata pertama adalah sifar. Dua kata tersebut dilampir pada mesej.Contoh seperti di ii), panjang mesej adalah 40 ( ditukar perenambelasan) dan dilampirkan menjadi

61626364 65800000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000028

- **Algoritma Cincang**

Algoritma cincangan adalah tanda untuk melindungi data daripada pengubahsuaian (Pfleeger, 1997). Ia digunakan untuk mengira nilai cincang bagi sesebuah data. Nilai cincang akan sama jika dilakukan pada data yang sama. Oleh itu, sebarang perubahan dapat dijejaki dengan menbandingkan nilai cincangan yang diperolehi.

Di bahagian seterusnya, algoritma cincang RIPEMD-160 (Dobbertin *et,al*, 1996, Dobbertin *et,al*, 1997 dan Robshaw, 1996) akan diterangkan secara terperinci, memandangkan projek ini telah memilih RIPEMD-160 sebagai enjin utama. RIPEMD-160 dipilih kerana proses mampatannya yang menggunakan dua baris

selari yang merupakan kekuatan pada algoritma ini. Manakala, SHA-1 (Burroes *et,al*, 1995) diterangkan secara ringkas sebagai perbandingan algoritma yang juga menghasilkan nilai cincang 160-bit. Walaubagaimanapun, proses mampatan pada SHA-1 dijalankan pada satu baris sahaja.

- **RIPEMD-160**

RIPEMD-160 merupakan salah satu fungsi cincang kriptografi yang direkabentuk oleh Hans Dobbertin, Antoon Boselears dan Bart Preneel dan dibangunkan oleh kumpulan kerja EU projek RIPE (*RACE Integrity Primitive Evaluation*, 1998-1992). RIPEMD-160 adalah versi terbaru (strengthened version) bagi RIPEMD yang menghasilkan nilai cincang 160-bit (Dobbertin *et,al*, 1996). Ia juga digunakan sebagai penggantian selamat bagi fungsi cincangan 128-bit seperti MD5 dan RIPEMD. Penggantian ini mempunyai dua sebab:

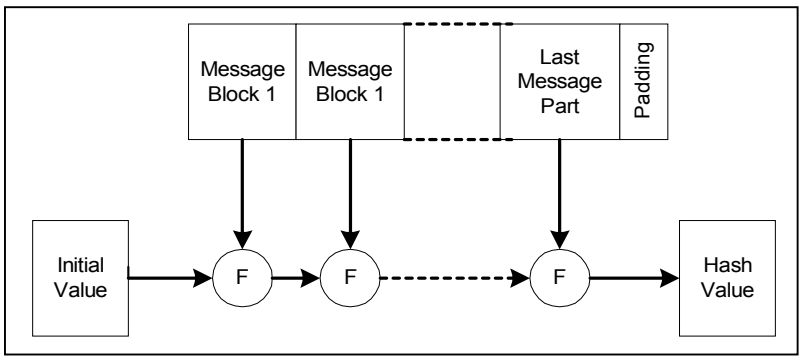
i) Cincangan 128-bit tidak lagi menyediakan perlindungan yang baik. Serangan pencarian pelanggaran *brute-force* pada cincangan 128-bit memerlukan $2^{64}$ atau $2.10^{19}$ penilaian fungsi. Pada tahun 1994, Paul van Oorschot dan Mike Wiener telah menunjukkan *brute-force* dapat dilaksanakan dalam masa kurang daripada sebulan dan telah menelan belanja sebanyak $10 juta pelaburan. (" Parallel collision search with application to hash function and discrete logarithms" *2nd ACM Conference on Computer and Communications Security*, ACM Press, 1994,pp 210-218).

ii) Hans Dobbertin memperolehi pelanggaran bagi versi RIPEMD yang menghadkan 2 daripada 3 pusingan sahaja. Pada musim luruh tahun 1995, Hans telah menggunakan teknik yang hampir sama dan memperolehi pelanggaran bagi semua pusingan MD4. Pada musim bunga tahun 1996, Hans memperolehi pelanggaran bagi fungsi mampatan MD5 dan seterusnya pelanggaran bagi MD5.

Ron Rivest juga telah mengemukakan satu bukti kriptanalitik bagi membuktikan bahawa MD4 tidak lagi sesuai digunakan dalam aplikasi. Selain itu, pelanggaran pada MD5 juga belum dapat diselesaikan. RSA *Laboratories* pula mencadangkan RIPEMD-160 sebagai satu penggantian alternatif bagi MD4. MD5 dan RIPEMD disamping SHA-1 (Robshaw, 1996).

Secara umum, fungsi cincang merupakan satu proses pengulangan fungsi mampatan ke atas sesuatu mesej. Fungsi mampatan ini memerlukan panjang pembolehubah sesuatu rentetan sebagai input dan menghasilkan satu rententan pendek yang mempunyai panjang tetap. Dalam proses ini, panjang satu mesej secara rawak akan dibahagikan kepada beberapa blok. Saiz blok ini bergantung kepada jenis fungsi mampatan. Sebagai contoh RIPEMD-160, mesej dibahagikan kepada blok bersaiz 513 bits setiap satu. Setiap blok akan dibahagi pula kepada 16 rentetan bersaiz 4 bait (32-bit kata). Kemudian , mesej dilampirkan dengan bit tambahan untuk menghasilkan mesej bersaiz 512 bit iaitu dua kali ganda saiz blok. Setiap blok akan diproses secara berjujukan dimana output daripada blok akan menjadi input kepada blok yang seterusnya. Output daripada blok yang terakhir akan menjadi nilai

terakhir bagi nilai cincang mesej tersebut (Dobbertin *et,al*, 1996). Rajah 3.6 menunjukkan proses mampatan dilakukan.



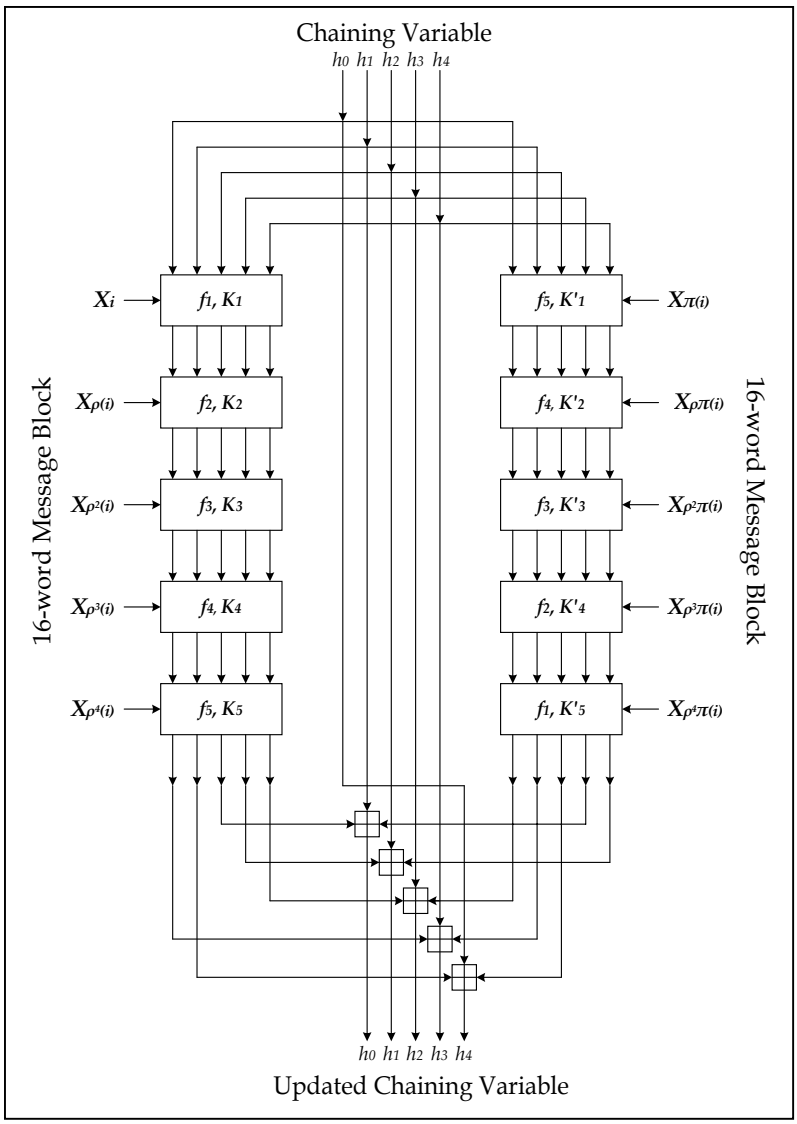

**Rajah 3.6: Struktur Umum Fungsi Cincang; F ialah fungsi mampatan**

Ini adalah fungsi logik $f_0$, $f_1$,..., $f_{79}$ yang digunakan oleh RIPEMD-160. Setiap f, 0 <= t <= 79 beroperasi pada tiga 32-bit kata iaitu x, y dan z. Nilai konstan dan fungsi yang digunakan adalah seperti berikut :

**Jadual 3.1 : Nilai Konstan Dan Fungsi Primitif RIPEMD-160**

| No. Langkah | Nilai Konstan | Fungsi logik | Nilai fungsi |
|---|---|---|---|
| 0 <= j <= 15 | K = 00000000 | f(j, x, y, z) | x XOR y XOR z |
| 16 <= j <= 31 | K = 5A827999 | f(j, x, y, z) | (x AND y) OR (NOT(x) AND z) |
| 32 <= j <= 47 | K = 6ED9EBA1 | f(j, x, y, z) | (x OR NOT(y)) XOR z |
| 48 <= j <= 63 | K = 8F1BBCDC | f(j, x, y, z) | (x AND z) OR (y AND NOT(z)) |
| 64 <= j <= 79 | K = A953FD4E | f(j, x, y, z) | x XOR (y OR NOT(z)) |

B. Preneel et.al di dalam kertas kerja mereka (Dobbertin *et,al*, 1997), mengajak penyelidik bersama-sama untuk mengkaji RIPEMD-160 dari segi keselamatannya. Mereke turut menyatakan bahawa RIPEMD-160 dicadangkan untuk menyediakan (*intended to provide*) keselamatan di peringkat tinggi (*high security level*) bagi 10 tahun akan datang dan mungkin lebih.




**Rajah 3.7: Fungsi Mampatan RIPEMD-160**

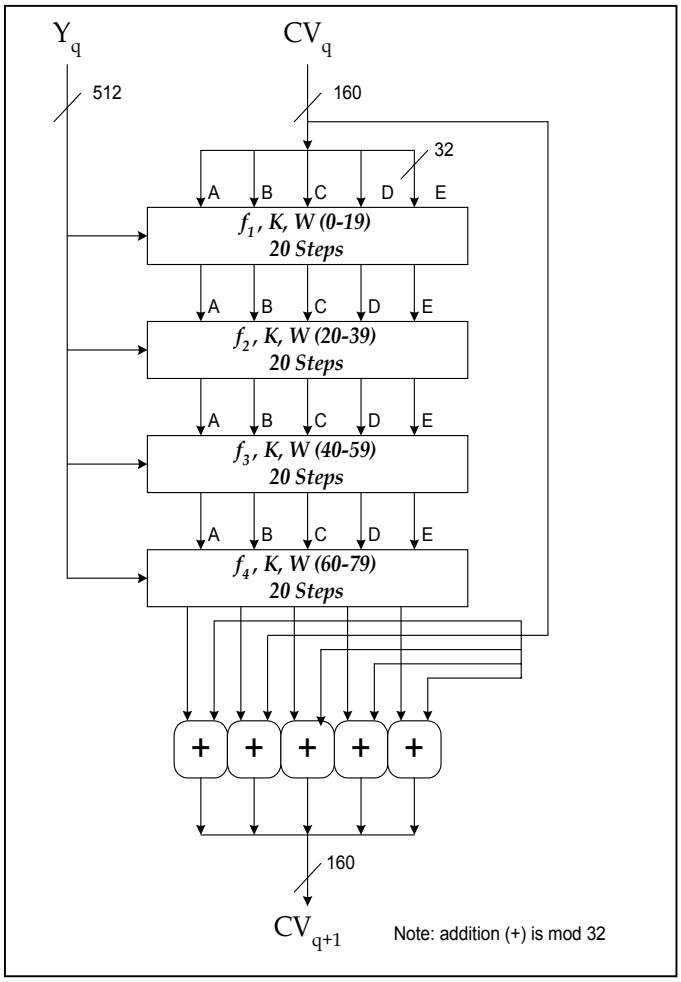
Walaubagaimanapun, kecanggihan teknologi dan peralatan memungkinkan satu baris daripada RIPEMD-160 diancam pada tahun hadapan. Walaupun demikian, dengan adanya penyatuan dua baris selari di dalam aliran proses RIPEMD-160, ia dapat menghalang keseluruhan algorithma daripada diancam. Ancaman tersebut mungkin memberi kesan kepada salah satu baris dan mungkin meningkat kepada tiga pusingan daripada dua baris selari. Rajah 3.7 menunjukkan fungsi mampatan RIPEMD-160 yang menggunakan dua baris selari berbanding SHA-1 pada Rajah 3.6.

### 3.1.4.2 SHA-1 (Secure Hash Function)

Secure Hash Algorithm dibangunkan oleh *National Institute of Standard and Technology* dan dikenali sebagai *federal information processing* dalam tahun 1993. SHA-1 sering digunakan sebagai sokongan dalam tandatangan tetapi juga digunakan di dalam semua aplikasi keselamatan yang melibatkan nilai cincangan. Contoh aplikasi, *Digital Signature Standard*. SHA-1 menggunakan panjang mesej kurang daripada $2^{64}$ bit dan menghasilkan 160-bit output yang dipanggil sebagai cerna mesej. SHA-1 direkabentuk untuk menyukarkan pencarian mesej asal kepada cerna mesej dan tidak ada dua mesej yang akan mempunyai cerna mesej yang sama.

Ini adalah fungsi logik $f_0$, $f_1$,..., $f_{79}$ yang digunakan oleh SHA-1. Setiap $f_t$, 0 <= t <= 79 beroperasi pada tiga 32-bit kata iaitu B, C, D dan menghasilkan satu 32-bit kata sebagai output. $f_t$(B,C,D) ditakrifkan seperti di dalam Jadual 3.1. Jujukan kata konstan K(0), k(1), …, K(79) digunakan di dalam SHA-1 dalam bentuk

perenambelasan dan dimuatkan di dalam penimbal. Nilai konstan yang digunakan disenaraikan seperti Jadual 3.2.




**Rajah 3.8: Fungsi mampatan SHA-1**

**Jadual 3.2 : Nilai konstan Dan Fungsi Primitif SHA-1**

| No. Langkah | Nilai konstan | Fungsi logik | Nilai fungsi |
|---|---|---|---|
| 0 <= t <= 19 | $K_t$ = 5A827999 | $F_t$(B,C,D) | (B AND C) OR ((NOT B) AND D) |
| 20 <= t <= 39 | $K_t$ = 6ED9EBA1 | $F_t$(B,C,D) | B XOR C XOR D |
| 40 <= t <= 59 | $K_t$ = 8F1BBCDC | $F_t$(B,C,D) | (B AND C) OR (B AND D) OR (C AND D) |
| 60 <= t <= 79 | $K_t$ = CA62C1D6 | $F_t$(B,C,D) | B XOR C XOR D |

### 3.1.4.3 Hasil Penelitian

Setelah penelitian dan kajian dilakukan didapati bahawa sistem tanda masa banyak diimplemen berasaskan internet dan amat penting terutama kepada perniagaan yang dilakukan secara atas talian atau e-dagang. Ini kerana ia dapat melindungi dan memastikan data atau maklumat yang dihantar benar-benar utuh tanpa pengubahsuaian daripada pihak-pihak tertentu.

Oleh sebab itu, sistem tanda masa juga sesuai jika diimplemenkan ke atas laman web yang merupakan entiti yang amat penting bagi sesebuah organisasi atau persendirian. Walaubagaimanapun, ia tidak akan melibatkan kepercayaan pihak ketiga tetapi akan menggunakan skema pautan. Persekitaran yang akan diimplemenkan adalah intranet yang tidak menggunakan menggunakan internet.

Terdapat beberapa algoritma cincang yang telah wujud dan perbandingan telah dilakukan. Perbezaan antara algoritma cincang yang diperolehi adalah seperti di dalam Jadual 3.3.

**Jadual 3.3 : Perbezaan Antara Algoritma Cincangan**

| | MD5 | SHA | RIPEMD | TIGER |
|---|---|---|---|---|
| Panjang cerna | 128 bit | 160 bit | 160 bit | 192 bit |
| Unit pemprosess | 512 bit | 512 bit | 512 bit | |
| Bilangan langkah | 64 (4 daripada 16 pusingan) | 80 | 64(5 daripada 16 pusingan) | |
| Maksimum panjang mesej | Infiniti | $2^{64}$ bit | | |
| Logik primitif | 4 | 3 | 3 | |
| Konstan tambahan | 64 | 4 | 4 | |

Mengikut pemerhatian Stalling(1996) terdapat perbezaan antara algoritma SHA-1 dan MD5 dari segi :

i) Keselamatan.

Panjang cernaan SHA ialah 32 bit berbanding cernaan MD5. Algoritma SHA lebih kuat. Dengan menggunakan teknik *brute-force*, kesukaran menghasilkan cernaan mesej adalah $2^{128}$ operasi bagi MD5 dan $2^{160}$ operasi bagi SHA.

ii) Kepantasan.

Oleh kerana kedua-dua algoritma bergantung kepada penambahan modulo $2^{23}$ , kedua-dua algoritma ini dilaksanakan dalam senibina 32-bit. Tetapi SHA melibatkan pertambahan langkah iaitu 80 berbanding 64 dan diproses menggunakan penimbal 160-bit. MD5 pula menggunakan penimbal 128-bit. Oleh sebab itu, SHA dijalankan 25 peratus lebih lambat berbanding MD5 pada perkakasan yang sama.

iii) Kepadatan dan kemudahan

Kedua-dua algoritma mudah untuk digambarkan dan mudah untuk diimplemen tanpa menggunakan aturcara yang besar atau jadual penggantian. Walaubagaimanapun, SHA menggunakan struktur satu langkah manakala MD5 menggunakan struktur 4 langkah. Tambahan pula, manipulasi penimbal perkataan adalah sama bagi semua langkah SHA, manakala setiap langkah MD5 memerlukan penyusunan perkataan. Oleh itu, SHA adalah lebih baik.

Perbandingan kepantasan antara algoritma yang telah dilakukan oleh Stalling (1996) boleh dilihat pada Jadual 3.4. Ia dikodkan dalam bahasa himpunan dan dilaksanakan menggunakan prosesor 90 MHz Pentium dalam model flat ingatan 32-bit.

**Jadual 3.4 : Perbandingan Kepantasan Antara Algoritma**

| Algoritma | Kitaran | Mbit/saat. | Mbait/saat | Pencapaian |
|---|---|---|---|---|
| MD5 | 337 | 136.7 | 17.09 | 0.72 |
| SHA-1 | 837 | 55.1 | 6.88 | 0.29 |
| RIPEMD-160 | 1013 | 45.5 | 5.68 | 0.24 |

Daripada perbandingan yang telah dibuat dua algoritma yang yang terpantas dan selamat akan dipilih untuk diimplemen di dalam projek ini.

## 3.2 Metodologi

Metodologi didefinasikan sebagai kombinasi konsep dan proses untuk dijadikan panduan yang praktikal dalam pembangunan sistem (Blaha *et,al*, 1998). Terdapat beberapa metodologi boleh digunakan dalam sesebuah pembangunan sistem seperti Model Air Terjun, Model Spiral, Model Prototaip dan Model Berorientasikan Objek.

Metodologi merupakan satu kaedah yang menyeluruh yang melibatkan beberapa langkah dalam memberikan panduan untuk pembangunan sistem. Ia juga mempengaruhi kualiti sesuatu sistem (Jeffrey *et,al* 1996). Ini akan menjadikan sesebuah pembangunan sistem lebih teratur dan tersusun. Di dalam pembangunan projek ini, metodologi prototaip (Jeffrey *et,al* 1996, Robshaw, 1996 dan Penny, 1996) yang berasaskan kitar hayat *Rapid Application Development* (RAD) telah

digunakan. Ini kerana jangka masa pembangunan projek adalah pendek, sistem yang akan dibangunkan tidak terlalu besar dan tidak melibatkan pangkalan data yang besar.

### 3.2.1 *Rapid Application Development* (RAD)

Di dalam teknik pembangunan sistem, kitar hayat tradisional (life cycle) yang biasa digunakan adalah *System Development Life Cycle* (SDLC). Terdapat satu kitar hayat yang boleh dijadikan alternatif daripada kitar hayat tradisonal iaitu *Rapid Application Development* (RAD) (Jeffrey et,al, 1996). Semua fasa yang terdapat di dalam kitar hayat tradisional terdapat di dalam RAD, tetapi setiap fasa dijalankan pada jangka masa yang pendek. RAD biasa digunakan bagi pembangunan sistem yang sangat memerlukan penglibatan pengguna seperti Joint Application Development (JAD). prototaip, integrasi perkakasan CASE (*Computer-Aided Software Engineering*) dan penjana kod yang memerlukan sistem dibangunkan dan diimplemen dalam jangka masa yang singkat.

Oleh sebab itu, RAD dapat membantu mempercepatkan proses pembangunan sistem, disamping dapat menjimatkan kos dan menghasilkan sistem yang berkualiti tinggi. RAD mengandungi empat fasa utama iaitu perancangan, analisis, rekabentuk dan implementasi yang dinamakan sebagai Perancangan Keperluan (requirement planning), Rekabentuk Pengguna (user design), Pembangunan (construction) dan Cutover. Melalui RAD, pembangunan sistem dapat dijalankan dalam masa enam bulan berbanding 24 bulan menggunakan SDLC traditional.

**Jadual 3.5: Empat Fasa Dalam Kitar Hayat RAD**

| | |
|---|---|
| Perancangan Keperluan | Projek mengenalpasti dan memilih keperluan serta analisa |
| Rekabentuk Pengguna | Membangunkan prototaip bagi menggantikan fasa analisa dan rekabentuk logikal |
| Pembinaan | Membangunan komponen bukan program seperti manual pengguna dan bahan latihan.. |
| *Cutover* | Menghantar sistem baru kepada pengguna akhir. |

### 3.2.2 Model Prototaip

Model prototaip merupakan salah satu metodologi yang menggunakan kitar hayat RAD. Prototaip adalah teknik yang dipinjam daripada bidang kejuruteraan (Jeffrey et,al, 1996). Pada tahun 1970-an, kebanyakkan ahli akademik dan professional dalam bidang sistem maklumat telah memandang rendah tentang keupayaan model prototaip yang sering kali dikaitan dengan perkataan “quick and dirty”. “Quick”dikaitankan dengan kekurangan dalam mendefinasi keperluan bagi menjalankan fasa rekabentuk dan pembangunan, manakala “dirty” dikaitan dengan penghasilan sistem yang tidak berdokumentasi dan sukar untuk diselenggarakan.

Walaubagaimana pun, pada awal tahun 1980-an, beberapa kajian telah dibuat dan mendapati bahawa model prototaip banyak memberikan kebaikan iaitu:

i. Memendekkan tempoh pembangunan system

ii. Kebolehgunaan (usability) yang lebih

iii. Mempertingkatkan komunikasi antara pembangun dan pengguna

iv. Mengurangkan kesan tarikh tamat (deadline)

Kaedah prototaip telah dipinda untuk mengatasi kelemahan yang ada seperti perancangan yang tidak teratur, tiada kawalan, proses yang tidak terurus, kesukaran mengintegrasikan sistem prototaip dengan sistem lain dan penghasilan rekabentuk sistem yang kurang pertalian berbanding kaedah lain (Sandra, 1996).

- **Kaedah dalam model prototaip**

Model prototaip mempunyai dua kaedah yang sering digunakan pada masa kini iaitu keperluan model protaip dan evolusi model prototaip (Sandra, 1996). Di dalam projek ini, kaedah kedua iaitu evolusi model prototaip digunakan bagi menjimatkan masa pembangunan sistem. Kedua-dua kaedah akan dijelaskan seperti di bawah :

i. Keperluan model prototaip

Menggunakan prototaip bagi menetapkan keperluan untuk sesuatu cadangan sistem. Kemudian, menunjukkan contoh sistem kepada pengguna. Ini membolehkan pengguna :

a. memahami fungsi dan kegunaan sistem

b. memastikan pembangunan sistem mengikut kehendak pengguna dan
c. memastikan sistem berjalan dengan baik.

Daripada maklum balas pengguna, prototaip sistem akan dibaikpulih atau dibina baru dan akan ditunjukkan sekali lagi kepada pengguna. Jika terdapat masalah atau keperluan baru, proses ini akan diulangi sehingga pengguna berpuas hati dengan sistem yang akan digunakan. Setelah dipersetujui, prototaip sistem akan dibuang manakala spesifikasi rekabentuk akan digunakan untuk membangunan sistem sebenar menggunakan teknik dan perkakasan lain.

ii. Evolusi model prototaip

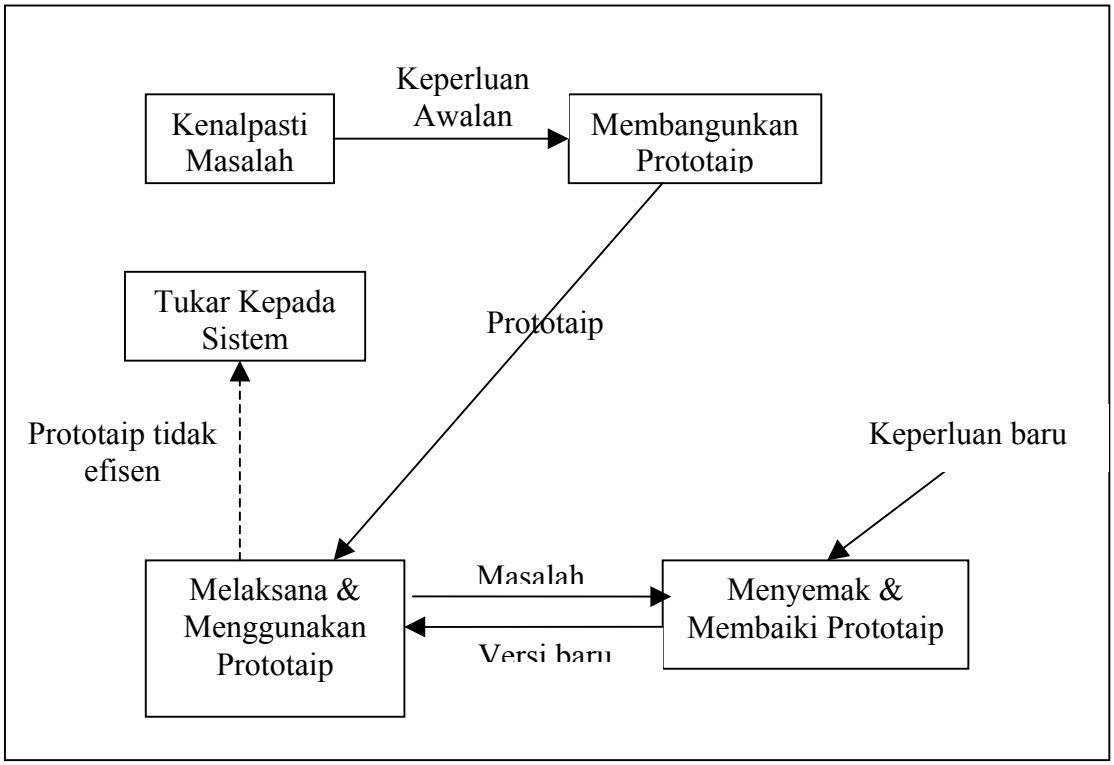
Keadah ini melaksanakan sepenuhnya metodologi pembangunan sistem. Proses yang dijalankan adalah sama seperti kaedah pertama tetapi protaip sistem tidak dibuang. Sebaliknya, teknik dan perkakasan yang sama akan digunakan semula untuk membangunkan sistem sebenar.

Proses-proses yang dijelaskan dalam kedua-dua kaedah dalam model prototaip boleh digambarkan seperti Rajah 3.5 (Jeffrey et,al, 1996). Proses di dalam membangunkan prototaip boleh dibahagikan kepada tiga peringkat [Penny, 1996] iaitu:

i. Peringkat pertama hanya melibatkan antaramuka pengguna seperti skrin. Segala proses seperti pengiraan dan manipulasi data menggunakan program dan data palsu.

ii. Pada peringkat kedua, program fungsi dan data sebenar akan digunakan untuk menggantikan program dan data palsu yang digunakan pada pringkat pertama..

iii. Pada peringkat ketiga, prototaip yang telah berfungsi sepenuhnya akan dikembangkan dengan menambah beberapa ciri tambahan seperti documentasi pengguna, skrin bantuan, keselamatan dan prosidur backup, dan kebolehan menguruskan sejumlah besar data transaksi. Pertambahan ini akan menghasilkan satu sistem akhir yang lengkap.




**Rajah 3.9: Metodologi prototaip**

- **Kebaikan dan Keburukan Model Prototaip**

Pembangunan prototaip semasa fasa analisa dan rekabentuk dapat menjelaskan keperluan pengguna dan dapat memastikan sistem akhir yang dibina memenuhi keprluan tersebut. Oleh itu, model prototaip telah memberikan beberapa kebaikan (Penny, 1996) iaitu:

i. Penglibatan pengguna secara terus akan menjadikan sistem lebih lengkap, tepat dan mempunyai antaramuka pengguna yang ramah pnegguna.
ii. Prototaip dapat memenuhi kehendak pengguna terutama keperluan yang diabaikan pada permulaannya.
iii. Pengguna lebih yakin menerima sesuatu sistem kerana telah mencubanya terlebih dahulu.
iv. Pengguna lebih mudah mengendalikan sistem kerana mereka telah memberi bantuan dalam pembangunan sistem tersebut.
v. Prototaip dapat mengurangkan kos dan masa pembangunan projek. Ini kerana “prototaip sangat effectif untuk meminimakan keperluan bagi pembanguan yang sederhana dan pertukaran skop”.
vi. Sistem yang dibangunkan prototaip semasa mengumpulkan data adalah lebih murah untuk diselenggara. Ini kerana ia dapat mengurangkan ketidakpastian, ralat dan pengabaian.
vii. Prototaip yang dikembangkan menjadi sistem akhir selalunya murah untuk diselanggara kerana kod dibangunkan menggunakan bahasa generasi keempat (4GL) berbanding bahasa generasi ketiga.

Walaubagaimanapun, terdapat juga beberapa kekangan di dalam projek ini yang juga merupakan keburukan (Penny, 1996) model prototaip iaitu:

i. Kemungkinan pengguna tidak mahu meluangkan masa dan usaha dalam menghasilkan prototaip yang dikehendaki.

ii. Prototaip yang menjadi sistem akhir berkemungkinan mempunyai perlaksanaan yang lambat, memerlukan ingatan yang laju dan menghalang bahagian sistem yang lain dilaksana.

iii. Prototaip selalunya tidak teratur bagi sistem yang memerlukan pemprosesan menggunakan algoritma komplek.

### 3.2.3 Keperluan Perisian Dan Perkakasan Serta Justifikasi Keperluannya

Untuk membangunkan sistem ini, beberapa pemerhatian telah dilakukan terhadap keperluan perkakasan dan perisian. Setelah diteliti, spesifikasi yang diperlukan adalah seperti berikut:

i) Perisian

a) Sistem pengoperasian Windows

b) Microsoft Access 2000 (pangkalan data)

c) Borland Jbulider 4 untuk Java

d) Microsoft Office ( dokumentasi)

e) Microsoft Project 2000 (carta gantt)

ii) Perkakasan

a) Komputer 366 Mhz

b) Ingatan utama 32-bit dan 64 Mb

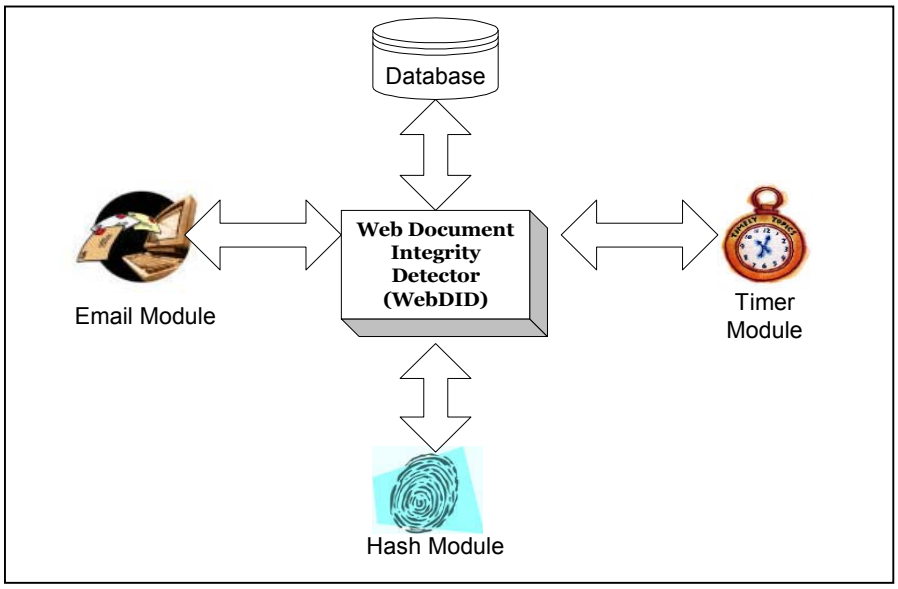
c) Minimum 4.2 Gb cakera keras

d) Skrin paparan SVGA

e) Peranti tambahan ( papan kekunci dan tetikus)

## 3.3 Rekabentuk

Bahagian ini membincangkan rekabentuk sistem secara menyeluruh “Web Document Integrity Detector (WebDID)”yang telah dibangunkan untuk mengesahkan keutuhan laman web. Java adalah bahasa pengaturcara yang digunakan dalam membangunkan WebDID.




**Rajah 3.10: WebDID**

Ia mengandungi beberapa bahagian seperti yang ditunjukkan dalam Rajah 3.10. Bahagian-bahagian tersebut adalah:

i. Bahagian cincangan yang mengandungi satu fungsi cincang iaitu RIPEMD-160. Bahagian ini akan menjana nilai cincang bagi setiap laman web dan membuat perbandingan bagi mengesahkan keutuhan laman web.

ii. Bahagian pemasa yang akan menetapkan tempoh masa untuk melakukan pengesahan keutuhan lamn web.

iii. Bahagian e-mel yang akan mencipta dan menghantar e-mel kepada pengguna sebagai amaran setelah pengubahsuaian laman web dikesan.

iv. Bahagian pangkalan data yang akan menyimpan dan mencapai beberapa maklumat seperti nama fail web, direktori dan nilai cincang.

### 3.3.1 Rekabentuk WebDID Secara Keseluruhan

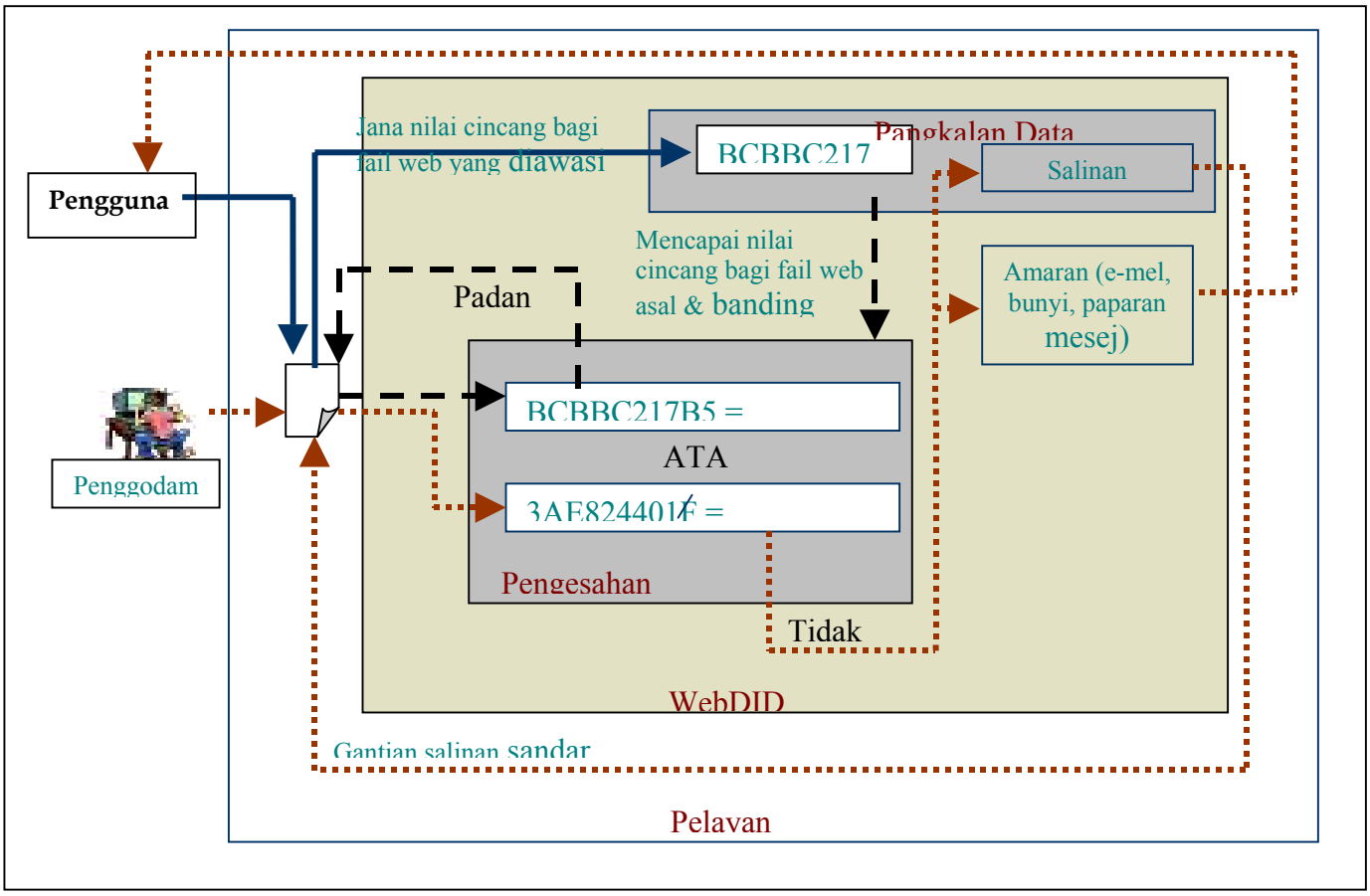
Penyelidikan dan pembangunan WebDID ini bertujuan untuk memastikan keutuhan laman web tidak terjejas. Keutuhan merupakan salah satu daripada komponen penting dalam keselamatan data selain daripada kerahsiaan dan kebolehharapan. Keutuhan bermaksud laman web hanya boleh diubahsuai oleh orang yang berhak dan dengan cara yang betul. Bagi mencapai matlamat ini, WebDID dibangunkan untuk mengawai dan mengesan laman web yang telah diubah dalam masa nyata (*real-time)*. Laman web tersebut termasuk fail web jenis HTML, ASP, PHP dan CGI.

WebDID mengaplikasikan pengunaan teknik kriptografi yang dikenali sebagai cincangan sebagai enjin utama. Ia digunakan untuk mengesahkan keutuhan laman web. Semasa sistem mengesan perubahan data dalam laman web, sistem akan melakukan langkah pemulihan dengan menggantikan laman web yang telah diubah

dengan salinan asal laman web. Kemudian, secara automatik sistem akan memberi amaran kepada pengguna melalui e-mel disamping bunyian, dan paparan mesej amaran di skrin komputer pelayan.

### 3.3.2 Jujukan Proses WebDID

Bahagian ini akan menunjukkan jujukan proses WebDID dalam melakukan pengesahan keutuhan laman web. Jujukan proses WebDD boleh dibahagikan kepada tiga bahagian iaitu sebelum, semasa dan selepas pengesahan keutuhan. Rajah 3.11 menggambarkan jujukan proses WebDID.




**Rajah 3.11: Aliran penjanaan tanda masa**

### 3.3.3 Sebelum pengesahan

Bagi membolehkan pentabdir sistem menggunakan WebDID, satu katalaluan diperlukan dan perlu dirahsiakan. WebDID akan melaksanakan fungsi cincang RIPEMD-160 untuk menjana nilai cincang bagi katalaluan tersebut dan disimpan di dalam pangkalan data. Ini bertujuan untuk menjaga kerahsiaan katalaluan tersebut.

Proses pengesahan keutuhan dimulakan dengan pentadbir sistem mengenalpasti dan memilih fail web yang hendak diawasi daripada diceroboh. Setelah itu, WebDID akan menjana nilai cincang bagi setiap fail web dengan menlaksanakan fungsi cincangan RIPEMD-160. Nama fail web dan nilai cincangan ini akan direkodkan ke dalam pangkalan data yang selamat. Dalam masa yang sama, sistem akan membuat satu salinan fail web dan diletakkan di satu direktori sandar yang ditentukan oleh pentadbir sistem.

Pentabdir sistem juga dikehendaki menetapkan kekerapan pengesahan keutuhan yang akan dilakukan dan menentukan alamat-alamat e-mel yang perlu dituju bagi membolehkan sistem menghantar pemberitahuan atau amaran pencerobohan, sekiranya berlaku. Nilai kekerapan dan alamat-alamat ini akan direkodkan di dalam pangkalan data.

### 3.3.4 Semasa pengesahan

WebDID akan melaksanakan proses pengesahan keutuhan fail web secara automatik mengikut kekerapan yang telah ditetapkan. Proses pengesahan keutuhan dimulakan dengan mencapai senarai fail web daripada pangkalan data dan seterusnya mencapai fail web tersebut yang terkini. Setelah itu, sistem akan melaksanakan fungsi cincangan pada fail web terkini untuk menjana nilai cincang. Nilai cincang terkini akan dibandingkan dengan nilai cincang fail web yang dicapai daripada pangkalan data.

Jika nilai cincang terkini bagi fail web tersebut adalah sama dengan yang dicapai daripada pangkalan data, ini menunjukkan bahawa keutuhan kandungan laman web tersebut tidak terjejas. Sebaliknya, jika nilai cincang fail web tersebut tidak sama, maka kandungan laman web tersebut telah dicerobohi atau telah diubah. Ini kerana segala perubahan yang berlaku pada laman web akan menghasilkan nilai cincang yang berbeza daripada nilai cincang yang sebenar yang disimpan di dalam pangkalan data. Dengan cara ini, sistem dapat mengesan bahawa pelayan web telah dicerobohi dan telah mengubah kandungan laman web.

### 3.3.5 Selepas pengesahan

Setelah pencerobohan dikesan, WebDID akan mencipta satu e-mel yang mengandungi fail web yang telah dicerobohi beserta masa pengesahan akan dihantar kepada pentadbir sistem sebagai pemberitahuan atau amaran tentang pencerobohan yang telah berlaku. Selain itu, sistem turut memberi amaran melalui bunyi dan paparan mesej di skrin komputer pelayan web.

Sebagai tindakbalas daripada pencerobohan, fail web yang telah dicerobohi akan dialihkan ke satu direktori lain. Proses penggantian fail akan berlaku di dalam direktori semasa dimana salinan fail web akan menggantikan fail web yang telah dicerobohi. Ini merupakan satu langkah pencegahan pencerobohan supaya pengguna tidak menyedari akan berlakunya pencerobahan tersebut dan masih boleh melayari laman web bagi sesebuah organisasi. Fail web yang dicerobohi yang telah disimpan

akan dianalisa oleh pentadbir sistem untuk menentukan langkah-langkah yang perlu diambil seterusnya.

### 3.3.6 Rekabentuk Pangkalan Data

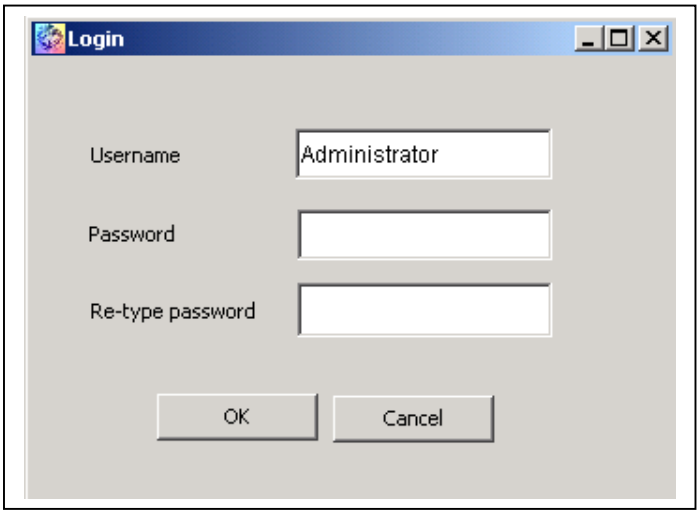
Bagi keseluruhan sistem yang dibangunkan, segala data disimpan di dalam sebuah pangkalan data Microsoft Access 2000. Pangkalan data bagi sistem ini adalah bersaiz kecil. Ini kerana pangkalan data ini hanya menyimpan beberapa maklumat penting seperti katalaluan, masa, alamat email, nama fail dan direktori serta nilai cincangan. Maklumat-maklumat ini akan dicapai oleh sistem semasa perlaksanaan proses pengesahan keutuhan laman web dan setelah pengubahsuaian laman web dikesan.

## 3.4 Hasil Pembangunan

Bahagian ini akan menunjukkan hasil pembangunan sistem yang telah dibangunkan. Pembangunan “Web Document Integrity Detector”ini diharapkan dapat mengesan dan menjaga keutuhan laman web daripada ancaman penceroboh. Input yang diperlukan oleh sistem adalah fail laman web, nama pengguna, kata laluan, masa dan alamat e-mel. Penerangan medan-medan ini boleh dilihat di dalam bahagian seterusnya.

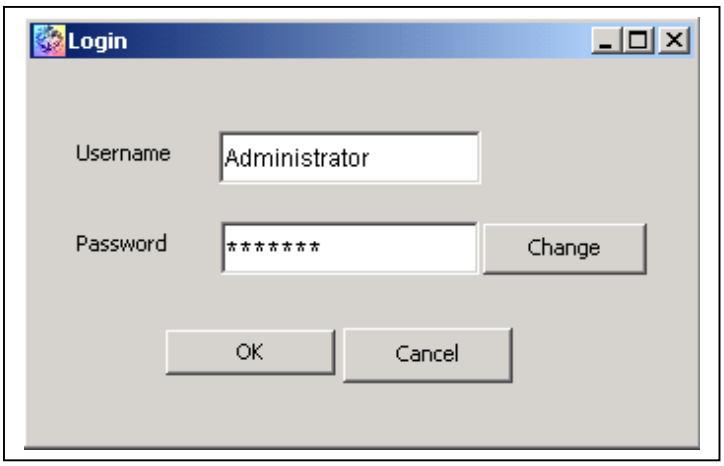
### 3.4.1 Skrin Login

Sebelum pengguna memasuki sistem dan memperolehi skrin utama, pengguna dikehendaki memberikan katalaluan yang betul. Katalaluan ini merupakan salah satu ciri keselamatan yang digunakan bagi memastikan hanya pengguna tertentu sahaja yang boleh mengendalikan sistem ini sebagai contoh pengurus pelayan web. Bagi pengguna yang baru pertama kali menggunakan sistem ini, skrin Login seperti Rajah 3.12 akan dipaparkan yang mengkehendaki pengguna memasukkan katalaluan dan mengesahkannya dengan menaip semula katalaluan tersebut. Katalaluan ini akan disimpan di dalam pangkalan data bagi membolehkan pengguna menggunakan sistem ini dilain masa. Katalaluan yang disimpan adalah selamat kerana katalaluan tersebut telah ditukarkan kepada nilai cincang dalam bentuk perenambelasan sebelum disimpan.




**Rajah 3.12: Skrin *Login* (Untuk Pengguna Baru)**

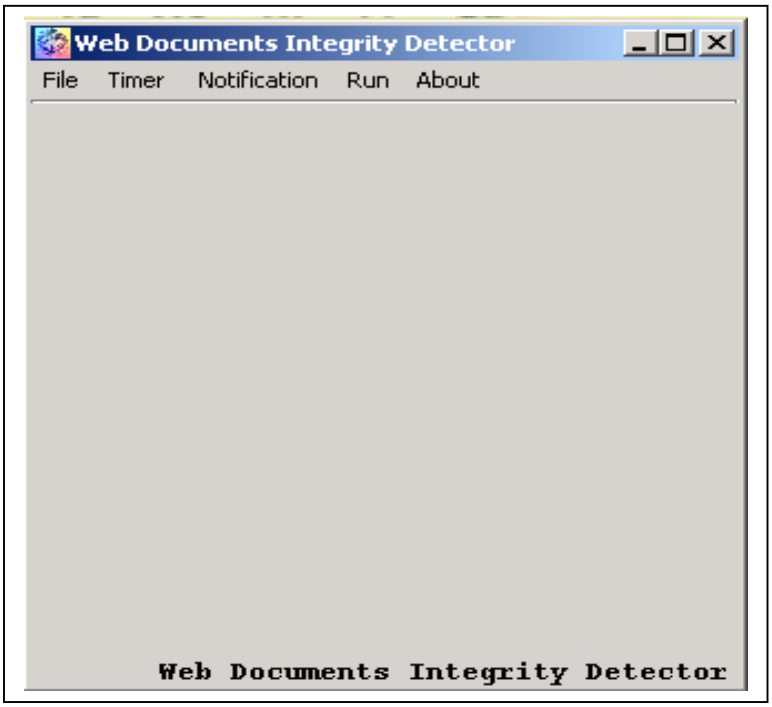
Penggunaan sistem seterusnya akan memaparkan skrin Login seperti Rajah 3.13. Pada skrin ini, pengguna dibenarkan membuat pertukaran katalaluan yang memerlukan katalaluan baru. Katalaluan lama juga diperlukan bagi memastikan perubahan katalaluan dilakukan oleh pengguna yang sah.




**Rajah 3.13: Skrin *Login***

### 3.4.2 Skrin Utama Web Document Integrity Detector

Skrin utama *Web Document Integrity Detector* seperti Rajah 3.14 mempunyai empat senarai menu iaitu *File, Timer, Notification, Configure, Run* dan *About*. Sebelum pengguna boleh menggunakan sistem ini, pengguna perlu melakukan konfigurasi seperti pemilihan fail web, tempoh masa untuk proses pengesahan, alamat e-mel untuk pemberitahuan jika berlaku pengubahsuaian. Pengguna hanya perlukan memilih tiga menu iaitu *File*, *Timer* dan *Notification*.

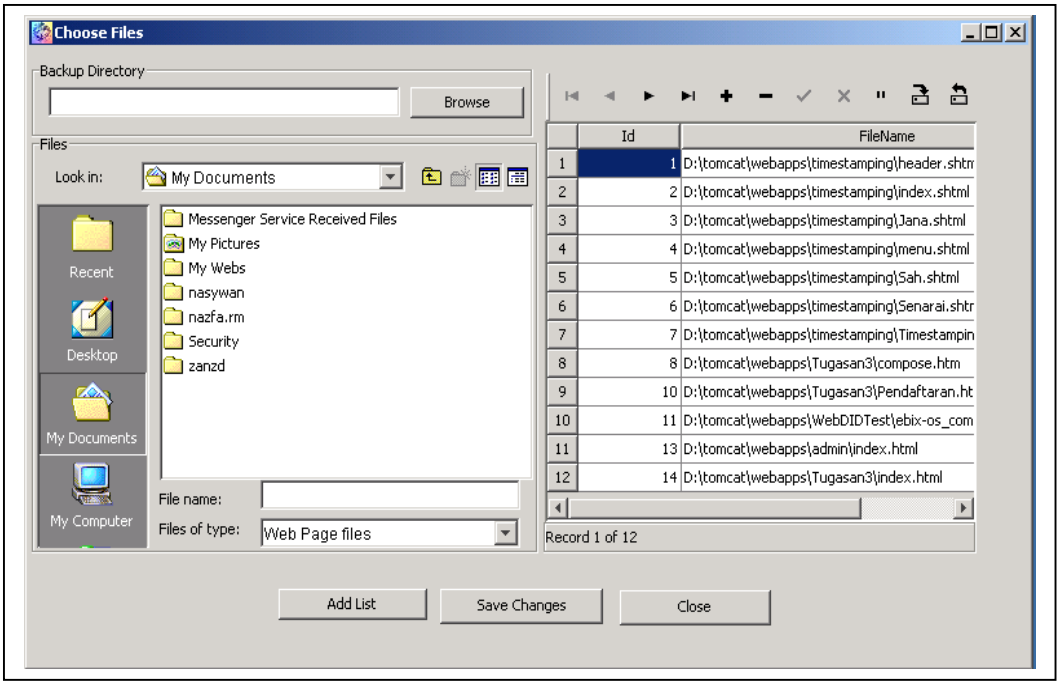



**Rajah 3.14: Skrin utama**

Menu *File* digunakan untuk mengistiharkan nama-nama fail web yang hendak diawasi. Pada menu ini, skrin *Choose Files* akan dipaparkan. Menu *Timer* digunakan untuk menetapkan tempoh masa proses pengesahan yang akan memaparkan skrin *Setup Timer*. Manakala, Menu *Notification* digunakan untuk menetapkan alamat e-mel bagi membolehkan sistem menghantar amaran berbentuk e-mel kepada pengguna. Setelah konfigurasi dilaksanakan, proses pengesahan keutuhan boleh dilarikan dengan memilih Menu *Run* yang akan memaparkan skrin *Checking Integrity*.

### 3.4.3 Skrin Choose Files

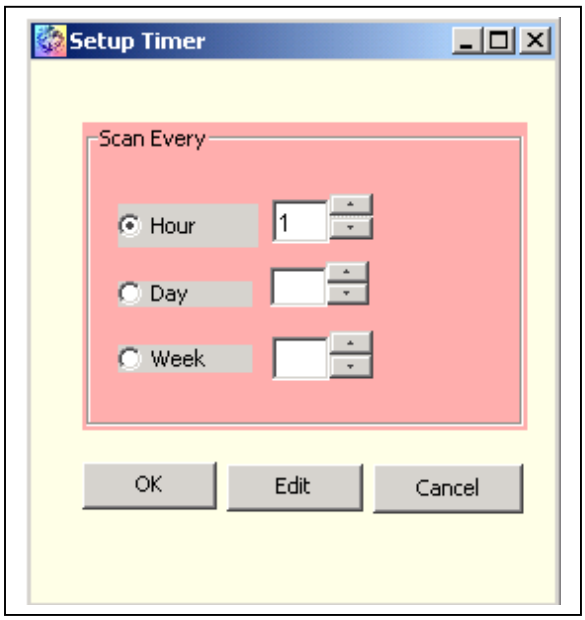
Skrin *Choose Files* seperti Rajah 3.15 akan memaparkan senarai fail web yang akan diawasi keutuhannya. Skrin ini memerlukan dua input iaitu nama fail web dan nama direktori. Pertama sekali, pengguna perlu menetapkan direktori sandar (*backup*) untuk meletakkan fail sandar yang akan digunakan semasa proses pemulihan (*recovery)*. Kemudian, pengguna perlu memilih fail-fail daripada direktori yang dipaparkan dan menekan butang <Add List> untuk dimasukkan dalam senarai. Butang <Save List> perlu ditekan apabila pengguna selesai memilih. Sistem akan menyimpan nama fail berserta direktori di dalam pangkalan data. Dalam masa yang sama, sistem akan membuat satu fail sandar untuk disimpan di dalam direktori sandar.




**Rajah 3.15: Skrin *Choose Files***

### 3.4.4 Skrin Setup Timer

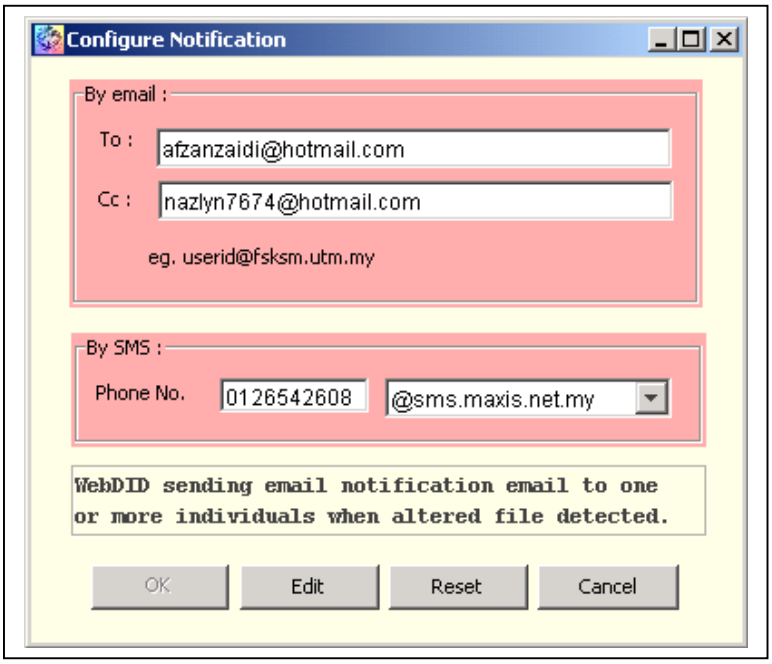
Rajah 3.16 merupakan skrin yang akan dipaparkan apabila pengguna hendak menetapkan masa persediaan (*setup time*) untuk proses pengesahan keutuhan dilakukan. Pengguna mempunyai tiga pilihan sama ada proses pegesahan dilakukan mengikut jam. hari atau minggu. Pengguna hanya perlu menekan salah satu butang <Radio> dan memilih bilangan jam atau hari atau minggu mengikut pilihan. Pengguna juga dibenarkan mengubahsuai masa ini dengan menggunakan skrin yang sama. Masa ini akan direkodkan di dalam pangkalan data. Ia akan dicapai sebelum proses pengesahan keutuhan laman web bermula.




**Rajah 3.16: Skrin Setup Timer**

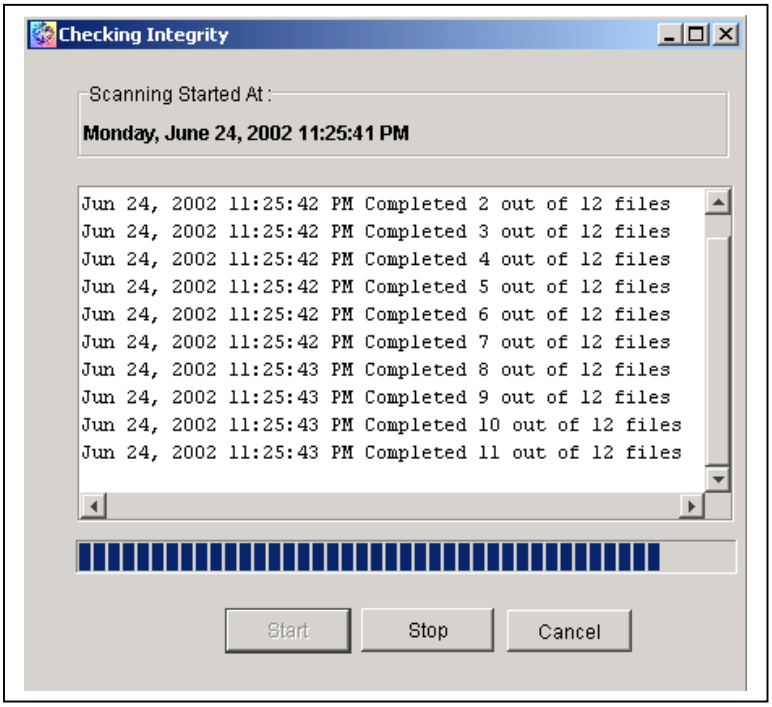
### 3.4.5 Skrin Configure Notification

Sistem memerlukan pengguna mengistiharkan alamat-alamat e-mel dan akan direkodkan di dalam pangkalan data. Ini boleh dilakukan pada skrin *Configure Notification* seperti yang ditunjukkan pada Rajah 3.17. Alamat e-mel ini akan digunakan oleh sistem untuk menghantar amaran dalam bentuk e-mel dan mesej pendek di telefon bimbit. Amaran ini hanya berlaku jika sistem mengesan pengubahsuaian telah berlaku pada laman web.




**Rajah 3.17 : Skrin *Configure Notification***

### 3.4.6 Skrin Checking Integrity

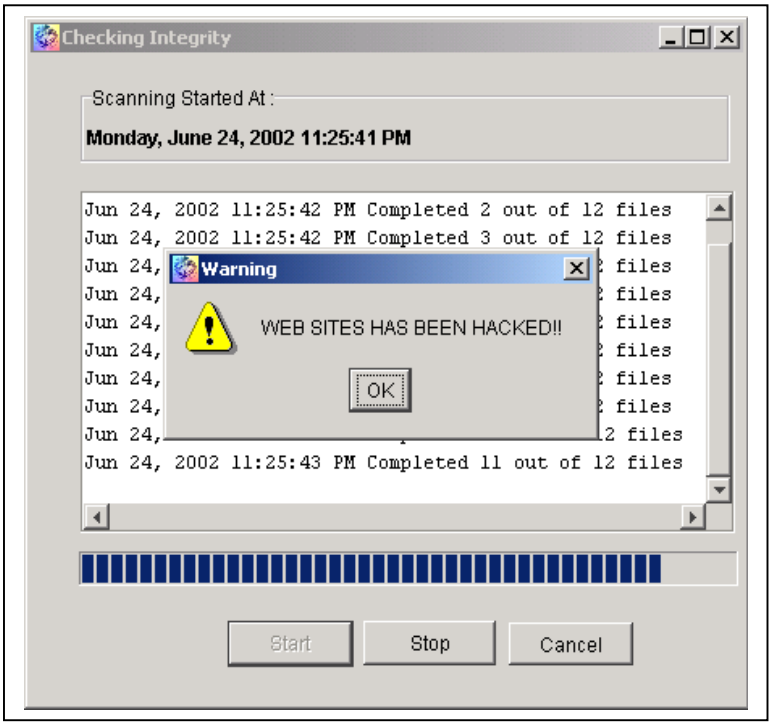



**Rajah 3.18 : Skrin *Checking Integrity***

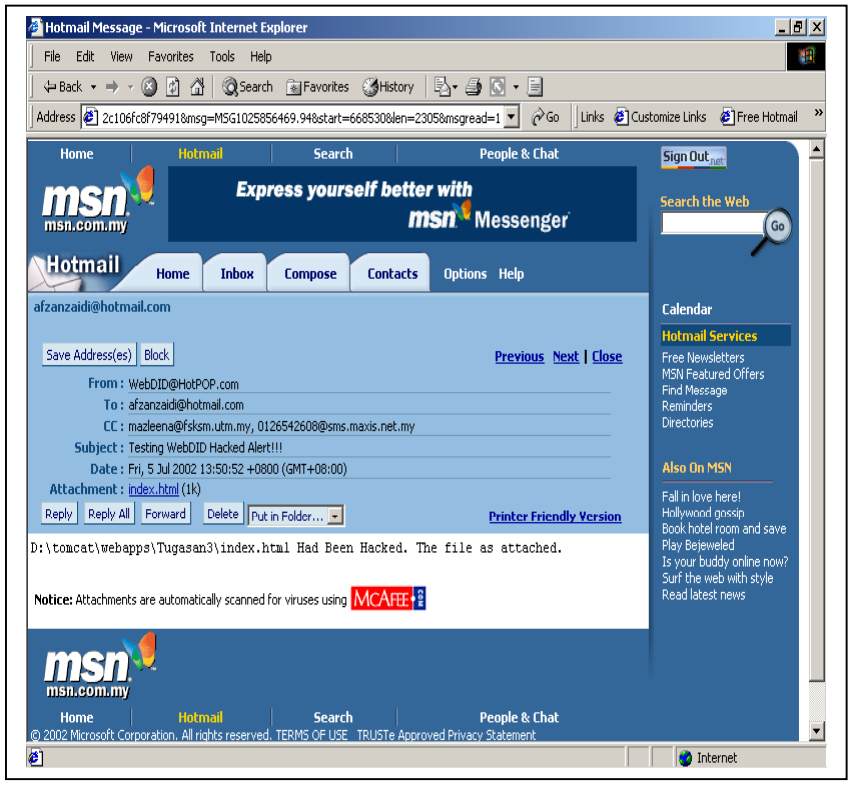
Proses pengesahan keutuhan boleh dilarikan dengan menekan butang <Start> yang terdapat pada skrin *Checking Intgerity* seperti yang ditunjukkan pada Rajah 3.18. Sebelum skrin ini dipaparkan, sistem telah mencapai rekod masa, senarai fail web dan nilai cincangan daripada pangkalan data. Semasa proses pengesahan keutuhan dilaksanakan, skrin ini akan memaparkan senarai bilangan fail yang telah diproses berserta waktu pengesahan.

Dalam masa yang sama, sistem akan melakukan proses mendapatkan nilai cincang bagi fail web semasa dan dibandingkan dengan nilai cincang yang diperolehi daripada pangkalan data. Kedua-dua nilai cincang tersebut mesti mempunyai nilai

yang sama bagi mengesahkan ia tidah diubah. Jika tiada pengubahsuaian dikesan, proses pengesahan yang seterusnya akan dilakukan seperti yang telah ditetapkan oleh pengguna. Jika sistem mengesan salah satu fail web telah diubahsuai, maka amaran berbentuk e-mel, mesej ringkas, bunyi dan paparan mesej pada skrin komputer akan dilakukan. Rajah 3.19 adalah contoh mesej amaran yang dipaparkan pada skrin komputer. Manakala Rajah 3.20 adalah contoh amaran melalui e-mel.




**Rajah 3.19 : Paparan Mesej Amaran Pada Skrin Komputer**




**Rajah 3.20 : Contoh Amaran Melalui E-mel**

### 3.4.7 Penerangan Kod Pengaturcaraan

Seksyen ini akan menerangkan teknik pengaturcaraan yang digunakan. Kod pengaturcaraan yang berkenaan juga ditunjukkan sebagai rujukan.

### 3.4.7.1 Algoritma Cincangan

Untuk melaksanakan fungsi cincangan, satu *library* keselamatan iatu Cryptix 3.2 digunakan. Ia boleh diperolehi daripada internet. Cryptix mengandungi semua algoritma sama ada cerna mesej atau sifer serta dikodkan dalam Java. Langkah-langkah untuk *install* Cryptix adalah seperti berikut:

i) *Unzip* fail Cryptix3.2

ii) Dapatkan directori src

iii) Compile aturcara dengan arahan

javac -d . cryptix\provider\Install.java

iv) Run dengan arahan java cryptix.provider.Install

*v)* Pernyataan "*# Added by Cryptix-Java installation program: security.provider.2=cryptix.provider.Cryptix"* akan terpapar dan ditambah pada fail lib/security/java.security di dalam JDK secara automatik.

Contoh keratan aturcara untuk melakukan cincangan menggunakan algoritma SHA-1 adalah seperti dibawah :

```
import cryptix.util.core.Hex;
import cryptix.provider.md.*;

…
...
```

```
MessageDigest sha = MessageDigest.getInstance("SHA-1");
while (inDataStream.available() != 0) {
        b = (byte) inDataStream.read();
        sha.update(b);
}

byte[] hash = sha.digest();
w = cryptix.util.core.Hex.dumpString(hash);
…
```

Fungsi algoritma cincangan SHA-1 dipanggil terlebih dahulu. Kemudian kandungan laman web dibaca satu-persatu dan ditukarkan ke dalam bentuk bait. Bait-bait tersebut dicerna dan diumpukan ke dalam sebuah tatasusunan. Kandungan tatasusunan yang berbentuk bait ditukar ke dalam bentuk perenambelasan yang nerupakan nilai cincangan.

### 3.4.7.2 Capaian Pangkala Data (DBMS)

Capaian servlet ke DBMS diperlukan dalam projek ini untuk mendapatkan dan menyimpan fail tanda masa. Kod aturcara seperti di bawah diperlukan untuk menjalinkan hubungan tersebut.

```
...
```

*String url = "jdbc:odbc:NamaPangkalanData";*

*Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");*

*Connection con = DriverManager.getConnection(url, user, password);*

**…**

**Url adalah nama pangkalan data yang dibina untuk sistem. Ia boleh dibina dengan menambah sumber data pada *ODBC Data Source* yang berada di dalam *Control Panel*. Nama pengguna dan katalaluan boleh disetkan.**

### 3.4.7.3 Hantar Amaran Melalui E-mel

*Library* Javamail 1.2 dan *JavaBeans Activation Framework* (JAF) diperlukan untuk membolehkan maklumat atau sijil tanda masa dihantar kepada pengguna melalui emel. Kedua-duanya boleh diperolehi di internet. Javamail 1.2 API menyediakan satu kelas abstrak untuk membina sistem mel. Manakala API menyediakan platfom tidak bergantungan dan rangka kerja protokol tidak bergantungan untuk membina mel dan aplikasi mesej berasaskan teknologi Java. Dibawah ini adalah keratan aturcara untuk menghantar emel.

i. Istihar kelas–kelas yang diperlukan.

*import javax.mail.**;

```
import javax.mail.internet.*;
import javax.activation.*;
```

ii. Dapatkan *system properties*

```
java.util.Properties prop = System.getProperties();
```

iii. Setup pelayan emel yang digunakan untuk menghantar emel kepada pengguna. Alamat e-mel pelayan mestilah mempunyai capaian POP dan penghantaran.

```
prop.put("mail.smtp.host",smtpHost);
Session ses = Session.getInstance(prop,null);
Store store = ses.getStore("pop3");
store.connect(popHost, username, password);
```

iv. Mengistiharkan kandungan mesej seperti alamat penghantar, alamat penerima, tajuk mesej, tarikh dan masa serta teks.

```
Message message = new MimeMessage(ses);
Address fromAddress = new InternetAddress(from);
message.setFrom(fromAddress);
 message.setSubject(subject);
```

```
Address[] toAddress = InternetAddress.parse(to);
 message.setRecipients(Message.RecipientType.TO,toAddress);
message.setSentDate(new java.util.Date());
 message.setText(text);
```

v. Hantar mel

```
Transport.send(message);
```

### 3.4.7.4 Amaran Melalui Bunyi

Bagi membolehkan amaran bunyi digunakan, dua kelas perlu diistiharkan iaitu:

```
import javax.sound.*;
import javax.sound.sampled.*;
```

Bunyi dikategori sebagai audio dan kelas seperti AudioFormat, AudioInputStream,AudioSystem, Clip, DataLine digunakan yang boleh didapati di dalam kelas *javax.sound.sampled.** . Contoh keratan kod aturcara adalah seperti di bawah dimana ia dimulakan dengan mengistihar nama fail audio, dan kemudian mencapai dan membuka fail audio tersebut.

```
AudioInputStream audioInputStream = null;
File clipFile = new File(Filename);
audioInputStream = AudioSystem.getAudioInputStream(clipFile);
```

```
if (audioInputStream != null)
{
        AudioFormat format = audioInputStream.getFormat();
        DataLine.Info info = new DataLine.Info(Clip.class, format);
        m_clip = (Clip) AudioSystem.getLine(info);
        m_clip.open(audioInputStream);
}
```

# BAB IV

# PENGESANAN PENCEROBOHAN BERASASKAN ANALISIS PERBEZAAN

## 4.1 Pengenalan

Kehadiran Internet memberikan banyak fasiliti dan kemudahan kepada pengguna rangkaian komputer. Kita lihat bahawa Internet telah menggantikan aktiviti-akitiviti yang dulunya dilakukan secara konvensional di mana ia telah menjadi cepat tanpa batasan jarak, ruang dan waktu. Pelbagai kesempatan dan peluang adalah tersedia dengan tidak terbatas; tetapi kemudahan dan fasiliti yang ada mempunyai risiko-risiko dan peluang bagi para penceroboh komputer untuk melakukan aktiviti jahatnya.

Adalah penting untuk merekabentuk sebuah mekanisma keselamatan bagi mencegah capaian tidak sah pada data dan sumber yang ada pada sistem komputer. Banyak kaedah telah diperkenalkan untuk mengesan pencerobohan; misalnya, kaedah pola yang bersesuaian (*Pattern Matching*) untuk mengetahui ciri-ciri penceroboh, (Kumar, 1995), dan pendekatan statistikal (*Statistical Approach*) yang mengesan penyimpangan-

penyimpangan dari aktiviti normal. Kaedah baru untuk mengesan pencerobohan berdasarkan kepada jumlah *system call* semasa aktiviti rangkaian seorang pengguna pada sesebuah hos (Midori, 2002).

Secara amnya, kaedah pengesanan pencerobohan dapat dipecahkan kepada tiga jenis iaitu kaedah pengesanan pencerobohan salah gunaan (*Misuse Intrusion Detection Method*) dan kaedah pengesanan pencerobohan tak lazim (*Anomaly intrusion Detection Method*). Kaedah pengesanan pencerobohan salah guna ialah menjejak pencerobohan dengan sampel data yang saling bersesuaian untuk mengetahui pola pencerobohan, dan kaedah pengesanan pencerobohan tak lazim adalah dengan menganalisis keganjilan aktiviti-aktiviti normal pada peringkat pengguna atau peringkat sistem (Sundaram, 1996).

Pada kaedah pengesanan pencerobohan tak lazim, biasanya digunakan sebuah pendekatan statistikal dan rangkaian neural. Kaedah ini juga boleh digunakan untuk mengesan pencerobohan yang belum dikenal pasti. Walau bagaimanapun, kaedah ini memerlukan peruntukan yang besar bagi mesin hos (Spafford, 1995) yang mana harus menyediakan kapasiti yang cukup besar untuk merekod semua aktiviti pengguna dan membuat ciri-ciri pengguna berdasarkan pengukuran yang telah didefinisikan untuk mengesan pencerobohan (Denning, 1987).

Pengesanan pencerobohan adalah suatu seni daripada pengesanan ketidakpatutan, kesalahan ataupun aktiviti yang tak lazim terjadi. Sistem pengesanan pencerobohan (*Intrusion Detection System*) yang beroperasi pada sebuah hos untuk

mengesan aktiviti-aktiviti pencerobohan pada hos tersebut dikenal sebagai Sistem Pengesanan Pencerobohan Berdasarkan Hos (*Host-based Intrusion Detection System*), dan sistem pengesanan pencerobohan yang beroperasi pada jaringan dan aliran data disebut sebagai Sistem Pengesanan Pencerobohan Berdasarkan Jaringan (*Network-based Intrusion Detection System*) (Lehman, 2000).

Pengesan pencerobohan yang berdasarkan hos melibatkan bahagian perisian pada sistem untuk dipantau. Pemuatan perisian menggunakan fail log dan sistem agen dalam melakukan ubah-suai sebagai sumber data. Sebaliknya, sistem pengesanan pencerobohan berdasarkan rangkaian memantau trafik pada segmen rangkaian sebagai sumber data. Pengesanan pencerobohan berdasarkan hos tidak hanya mencari data sumber pada trafik komunikasi di dalam atau di luar dari sebuah komputer tunggal, tetapi juga melakukan pemeriksaan pada keutuhan sistem fail dan melihat proses yang mencurigakan. Untuk memperoleh pencapaian yang lengkap pada sebuah lokasi pengesanan pencerobohan komputer, terdapat dua kelas utama daripada perisian pengesanan pencerobohan berasaskan hos (Zirkle dan Virginia, 2000).

Sebuah kaedah pengesanan pencerobohan baru berasaskan kepada analisis pembezaan (*Discriminant Analysis*) telah dikenal pasti oleh penyelidik Jepun dalam agen promosi teknologi dan maklumat (*Information-technology Promotion Agent*) [Midori *et.al*, 2001]. Analisis pembezaan adalah sebuah teknik dalam statistik yang digunakan untuk membezakan antara dua kelompok atau populasi yang saling bertindih. Mereka menyelidiki sebuah kaedah baru untuk mengesan pencerobohan berasaskan kepada *system call* semasa aktiviti seseorang pengguna rangkaian pada sebuah mesin

hos. Kaedah ini berusaha untuk memisahkan pencerobohan dari aktiviti normal menggunakan analisis pembezaan, iaitu salah satu jenis analisis *multivariate* (*Multivariate Analysis*). Analisis *multivariate* adalah salah satu teknik yang digunakan untuk mencari pola yang saling berkaitan di antara beberapa pembolehubah secara berterusan.

Melihat pada masalah yang ada pada sistem pengesanan pencerobohan berasaskan hos di atas, pengarang mencuba mengembangkan kaedah analisis pembezaan untuk dapat membezakan antara aktiviti normal dan aktiviti pencerobohan pada suatu mesin hos. Dengan menggunakan analisis ini juga diharapkan dapat mengurangkan kos di dalam sistem pemprosesan ataupun sistem penyimpanan di dalam merekod aktiviti yang dilakukan oleh seorang pengguna.

Penyelidikan ini difokuskan kepada pengesanan pencerobohan berasaskan hos. Pada sistem ini dijalankan beberapa buah program aplikasi yang akhirnya pada setiap program tersebut akan menghasilkan sebuah fail log yang dijadikan sebagai sumber data dalam penyelidikan. Di dalam pengesanan pencerobohan ini akan dilakukan tumpuan kepada jumlah proses yang terjadi, masa yang diperlukan untuk menghasilkan *system call*, ciri-ciri ataupun jumlah *system call* itu sendiri. Juga hubung-kait setiap *system call* untuk menentukan samada sebuah aktiviti normal atau aktiviti pencerobohan.

## 4.2 Implementasi Sistem Pengesanan Pencerobohan

Untuk melakukan pengesanan pencerobohan yang terjadi pada suatu sistem komputer, maka perlu dilakukan percubaan-percubaan yang merujuk kepada kaedah penyelidikan yang telah di bahas pada bab sebelumnya. Percubaan yang dilakukan tersebut dimaksudkan untuk menguji sampai sejauh mana sistem yang telah direka bentuk boleh mengesan pencerobohan yang terjadi.

### 4.2.1 Mengira Jumlah *System Call*

Di dalam penyelidikan ini, kita akan membezakan antara aktiviti normal dan aktiviti pencerobohan pada sistem pengoperasian Linux (RedHat Linux 7.3 kernel 2.4.18-3) dengan menggunakan kaedah-kaedah pembezaan. Iaitu sebuah kaedah baru untuk memisahkan pencerobohan dari aktiviti normal. Di dalam percubaan ini, kita telah menetapkan aktiviti-aktiviti harian yang dilakukan pada program aplikasi sebagai sebuah aktiviti normal dan pemilihan daripada aktiviti tersebut dilakukan secara sembarang. Setelah aktiviti tersebut dijalankan pada beberapa program aplikasi, seterusnya kita akan memperoleh sistem fail log (D. Endle, 1998). Dan aktiviti-aktiviti pencerobohan dilakukan dengan cara melakukan cracking ke dalam akaun yang ada pada sistem. Program cracking ini boleh diperoleh dengan cara mendownload nya dari beberapa lokasi penyerang yang terdapat pada Internet. Kita merekodkan hampir semua daripada fail log *system call* ketika menjalankan program aplikasi sebagai aktiviti normal dan

menjalankan cracking tool sebagai aktiviti pencerobohan. Adapun contoh daripada fail log tersebut adalah:

```
Localhost.localdomain LinuxAudit event,chmod(), Tuesday,Feb
11,16:12:10,2003,user,
root(0),root(0),root(0),root(0) process,2460,in.telnetdpath,/dev/pts/11
attributes,rw---return,0, sequence,68380

Localhost.localdomain LinuxAudit event,chown32(),Tuesday,Feb
11,16:12:10,2003,user,
root(0),root(0),root(0),root(0) process,2460,in.telnetdpath,/dev/pts/11
attributes,rw-rw-rw return,0, sequence,68382
```

Kita mengira jumlah *system call* yang ada di dalam fail log. Kadang-kadang terdapat beberapa *system call* yang tidak diingini: kita tidak merekodkan beberapa *system call* tertentu yang terjadi secara berterusan dan *system call* tersebut tidak berkaitan dengan daemon rangkaian (misalannya ioctl). Sebagai contoh kita akan melihat sekumpulan *system call* yang dihasilkan ketika dilakukan aktiviti sambungan pada aplikasi sendmail (Jadual 4.1).

**Jadual 4.1 *System Call* Aplikasi Sendmail**

| Jumlah Peristiwa | Nama Peristiwa | System Call | Kejadian |
|---|---|---|---|
| 30 | AUE_EXECVE | execve | 1 |
| 72 | AUE_OPEN_R | open, read | 16 |
| 210 | AUE_MMAP | mmap | 8 |
| 112 | AUE_CLOSE | close | 7 |
| 8 | AUE_CHDIR | chdir | 2 |
| 214 | AUE_MUNMAP | munmap | 2 |

"AUE_XXX" menerangkan sebuah nama dari pada peristiwa ubah suai. Baris yang paling depan menandakan sebuah ubah suai jumlah peristiwa yang diikuti oleh ubah suai nama peristiwa . Dan nombor yang paling belakang menunjukkan jumlah kejadian selama terjadi sambungan.

Pada contoh di atas, *system call* execve terjadi sebanyak satu kali, *system call* open dan read terjadi 16 kali, mmap terjadi 8 kali dan seterusnya. Dengan adanya data ini, kita mengira jumlah *system call* yang terjadi pada setiap contoh, dan memisahkan setiap contoh ke dalam aktiviti normal dan aktiviti pencerobohan menggunakan analisis pembezaan. Kita menggunakan perintah "strace" yang sedia ada pada sistem operasi Linux yang boleh merekodkan *system call* dari sebuah program aplikasi yang tengah berjalan. Kita mengumpulkan contoh data dalam penyelidikan ini menggunakan perintah tersebut. Dan kita pun telah memperoleh fail log dengan menjalankan 4 program cracking dan menjalankannya 26 aktiviti normal.

### 4.2.2 Membuat Pembolehubah Keterangan (making explanatory variabel)

Seksyen ini mencuba menyelidiki anggapan bahawa contoh-contoh yang digunakan dalam percubaan mempunyai pembahagian secara *multivariate* normal. Jika kita hanya mencadangkan untuk memisahkan kelompok pencerobohan dan aktiviti normal, adalah tidak perlu untuk membuat perumpamaan secara normal pada kedua kelompok tersebut. Walau bagaimanapun, jika kita menginginkan untuk mengklasifikasikan aktiviti baru atau aktiviti yang belum diketahui ke dalam setiap

kelompok pencerobohan atau normal aktiviti, kita pertama kali harus menguji kenormalan dari pada kedua kelompok tersebut. Oleh kerana itu, kita telah melakukan percubaan pada kedua kelompok tersebut dan ianya terbagi secara normal.

Jika kita boleh memilih *system call* sebagai pembolehubah keterangan, maka proses tersebut adalah mudah. Walau bagaimanapun juga, penyebaran *system call* tidak mengikuti pembahagian secara *multivariate* normal. Oleh sebab itu, kita tak boleh mengklasifikasikan contoh yang belum diketahui. Untuk mengatasi masalah ini, kita harus mengubah contoh tersebut kepada pembahagian normal yang terdekat. Kita menyelenggarakan sebuah analisis komponen utama (*principal component analysis*) daripada *system call* untuk mengizinkan mereka mempunyai sifat pembahagian secara normal. Hasil daripada komponen utama diambil sebagai pembolehubah keterangan untuk digunakan pada analisis pembezaan.

Analisis komponen utama mengambil sebahagian kecil jumlah pembolehubah pada struktur variance-covariance. Tujuan daripada komponen utama adalah untuk penurunan data. Boleh dikatakan analisis komponen memungkinkan untuk mengeluarkan semula jumlah pembolehubah dari kelompok *system call* pada kelompok yang baru, seperti jumlah komponen yang kurang dari jumlah *system call* itu sendiri. Jika skor keutamaan baru terbahagi secara normal, memungkinkan untuk pengklasifikasian ke dalam kelompok normal atau kelompok pencerobohan. Sebuah komponen utama sesuai untuk sebuah eigenvector daripada matriks covariance atau matrix hubung kait. Ketika semua unit pembolehubah keterangan sama, matriks

covariance yang digunakan untuk perkiraan. Jika berbeza, maka matriks hubung kait yang digunakan.

Kita menggambarkan sebuah hasil analisis komponen utama yang mana pembolehubah keterangan adalah *system call* pada aktiviti pencerobohan dan aktiviti normal di dalam UNIX. Secara amnya, jika sejumlah besar pembolehubah penjelas telah dipilih, maka kumulatif nisbah dari sumbangan cenderung tidak mencapai 100% dengan cepat. Guna mengelakkan masalah ini, kita memilih *system call* khusus yang erat kaitannya dengan pencerobohan sebagai pembolehubah penjelas. Kita telah memilih 10 *system call* dari 105 sytem call yang terjadi di dalam contoh yang dibuat di antara 250 *system call* yang sedia ada pada sistem operasi Linux. Seperti; open, read, write, getpgrp, setgid 32 dan seterusnya. Kita menggantikan *system call* yang telah kita ramalkan berkaitan erat dengan pencerobohan dengan beberapa buah pembolehubah (Jadual 4.2) untuk memudahkan pemrosesan.

**Jadual 4.2 Pembolehubah *System Call***

| ***System call*** | open | read | write | getgrp | Setgid32 | Setuid32 | munmap | Chown32 | execve | umask |
|---|---|---|---|---|---|---|---|---|---|---|
| **P. Ubah** | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | X9 | X10 |

Setelah kita menentukan pembolehubah untuk setiap *system call* yang akan pakai pada pemprosesan secara statistik, kita melakukan sebanyak 26 kali aktiviti normal dan 4 kali aktiviti pencerobohan ke dalam sistem yang sedang berjalan. Dengan

menggunakan perintah strace yang sedia ada pada Linux, maka kita akan peroleh jumlah semua *system call* yang telah di rekod oleh perintah tersebut (Jadual 4.3). Di sini kita hanya menetapkan 10 *system call* sahaja yang dilakukan pemprosesan seperti yang telah disebutkan di atas.

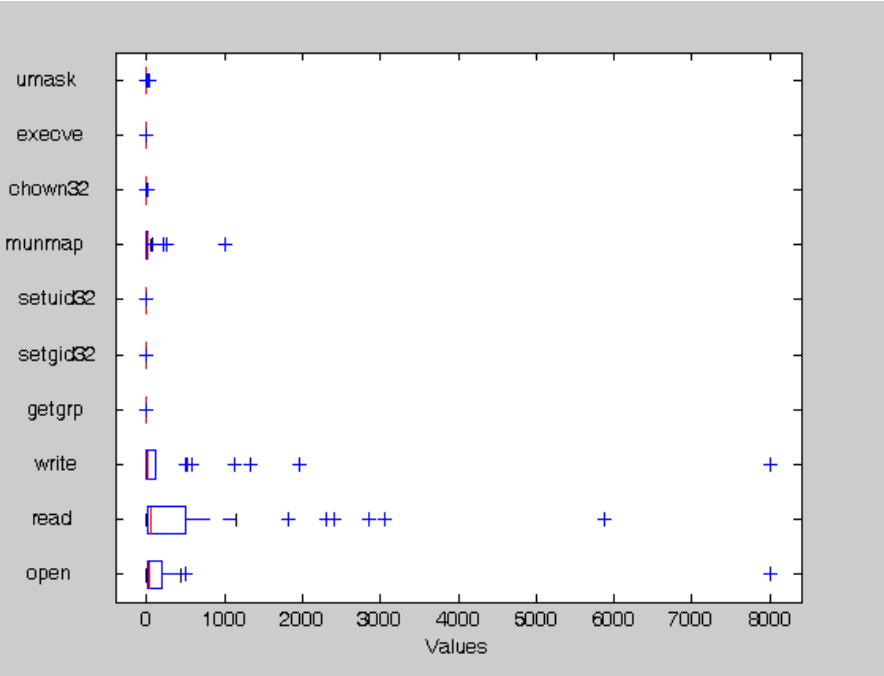
**Jadual 4.3 Jumlah *System Call* di Rekod dengan Perintah Strace**

| *Aktiviti* | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | X9 | X10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **Ftp** | 28 | 71 | 21 | 0 | 0 | 0 | 14 | 0 | 0 | 0 |
| **Mount** | 23 | 8 | 1 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| **Passwd** | 50 | 181 | 13 | 0 | 0 | 0 | 30 | 2 | 0 | 4 |
| **Rlogin** | 14 | 32 | 19 | 0 | 0 | 1 | 9 | 0 | 0 | 0 |
| **Ssh** | 42 | 1153 | 61 | 0 | 1 | 1 | 17 | 0 | 0 | 1 |
| **Telnet** | 19 | 16 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 |
| **Su** | 48 | 62 | 0 | 0 | 0 | 0 | 18 | 0 | 0 | 4 |
| **Login** | 23 | 7 | 1 | 0 | 0 | 0 | 2 | 1 | 0 | 0 |
| **Reboot** | 20 | 11 | 2 | 0 | 0 | 1 | 8 | 0 | 2 | 2 |
| **Finger** | 21 | 20 | 11 | 0 | 0 | 0 | 13 | 0 | 0 | 0 |
| **Netstat** | 21 | 7 | 77 | 0 | 0 | 0 | 5 | 0 | 0 | 0 |
| **Nmap** | 26 | 46 | 24 | 0 | 0 | 0 | 16 | 0 | 0 | 32 |
| **Ps** | 502 | 501 | 95 | 0 | 0 | 0 | 19 | 0 | 0 | 0 |
| **Mozilla** | 386 | 3050 | 497 | 1 | 0 | 0 | 47 | 0 | 1 | 0 |
| **Kde** | 438 | 5874 | 1967 | 0 | 0 | 0 | 260 | 16 | 0 | 0 |
| **Pine** | 77 | 129 | 108 | 0 | 0 | 0 | 22 | 0 | 0 | 0 |
| **Df** | 21 | 6 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **W** | 206 | 235 | 4 | 0 | 0 | 0 | 13 | 0 | 0 | 0 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Wget** | 32 | 27 | 21 | 0 | 0 | 0 | 13 | 0 | 0 | 0 |
| **Who** | 20 | 19 | 2 | 0 | 0 | 0 | 3 | 0 | 0 | 1 |
| **Ping** | 4 | 2 | 12 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| **Du** | 88 | 3 | 591 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| **Telinit** | 4 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| **Uname** | 17 | 3 | 1 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| **Pico** | 12 | 82 | 90 | 0 | 0 | 0 | 4 | 0 | 0 | 0 |
| **Emacs** | 215 | 1825 | 1128 | 0 | 0 | 0 | 74 | 0 | 0 | 0 |
| ***Mozilla*** | 388 | 2316 | 522 | 1 | 0 | 0 | 46 | 0 | 2 | 0 |
| ***Kde*** | 398 | 2862 | 1324 | 0 | 0 | 0 | 225 | 7 | 1 | 16 |
| ***Pico*** | 14 | 59 | 71 | 0 | 0 | 0 | 4 | 0 | 1 | 0 |
| ***Emacs*** | 222 | 2401 | 1577 | 0 | 0 | 0 | 74 | 0 | 1 | 0 |

Dari berbagai-bagai aktiviti pada jadual di atas, boleh dapat dijelaskan bahawa; aktiviti yang ditulis miring adalah aktiviti pencerobohan yang dilakukan dalam bentuk buffer over flow terhadap program aplikasi yang terdapat pada Linux, dan aktivitinya adalah merupakan aktiviti normal pada berbagai program aplikasi.

Dan sebagai langkah awal akan lebih bijaksana jika kita membuat analisis kepada data yang kita gunakan dalam percubaan. Misalnya kita menguji pembahagian ruangan dengan menggunakan fungsi boxplots (Rajah 4.1) dan ringkasan perkiraan pembolehubah pada jumlah pengamatan, nilai purata, piawai penyimpangan, nilai maksimum dan nilai maksimum sytem call yang telah dihasilkan (Jadual 4.4)




**Rajah 4.1 Pembahagian Ruang Pada *System Call* Menggunakan Boxplot**

Untuk melakukan perkiraan, kita menggunakan rumusan statistik untuk menentukan nilai purata seperti;

$$\text{Nilai purata } (X_k) = \frac{\sum_{j=1}^{N} X_{j,k}}{N}$$

Di mana Xi (i =1,2,…10) adalah komponen untuk pembuleh ubah yang ke-i, dan N adalah jumlah pengamatan. Sedangkan untuk menentukan piawai penyimpangan, kita menggunakan rumus:

$$\text{Piawai penyimpangan } (Sd_k) = \sqrt{\frac{(X_{jk} - X_k)^2}{N - 1}}$$

**Jadual 4.4 Ringkasan Perkiraan**

| Variable | N | Purata | Piawai penyimpangan | Minimum | Maksimum |
|---|---|---|---|---|---|
| **Open** | 30 | 371.90 | 1.4488 | 4 | 8000 |
| **Read** | 30 | 700.30 | 1.3577 | 1 | 5874 |
| **Write** | 30 | 488.87 | 1.4930 | 0 | 8000 |
| **Getgrp** | 30 | 0.07 | 0.0003 | 0 | 1 |
| **Setgid32** | 30 | 0.03 | 0.0002 | 0 | 1 |
| **Setuid32** | 30 | 0.01 | 0.0003 | 0 | 1 |
| **Munmap** | 30 | 62.73 | 0.1870 | 0 | 1000 |
| **Chown32** | 30 | 0.87 | 0.0031 | 0 | 16 |
| **Execve** | 30 | 0.27 | 0.0006 | 0 | 2 |
| **Umask** | 30 | 2.00 | 0.0064 | 0 | 32 |

Setelah membuat analisis komponen utama, kita telah memilih 3 komponen utama bagi eigenvalue yang mana memiliki markah lebih besar dari 1 dan sumbangan nisbah pertambahan tidak melebihi 80% (Jadual 4.5). Formula untuk menghitung eigenvalue adalah:

$$\left| \lambda - \lambda I \right| = 0$$

di mana $\lambda$ adalah eigenvalue dan I adalah matriks identiti. Dan untuk mendapatkan nilai daripada eigen vector, terlebih dahulu kita menentukan nilai matriks covariance dengan formula:

$$\Sigma = \frac{\sum(X_p - \mu_p)(X_1 - \mu_1)}{N - 1} \frac{\sum(X_p - \mu_p)(X_2 - \mu_2)}{N - 1}$$

**Jadual 4.5 Eigenvalue dan Sumbangan Nisbah**

| Nombor Komp. | Eigenvalue | Sumbangan | Pertambahan |
|---|---|---|---|
| 1 | 6.21 | 62.13 | 62.13 |
| 2 | 2.12 | 21.23 | 83.86 |
| 3 | 1.05 | 10.51 | 93.87 |

Setelah kita mendapatkan nilai-nilai untuk eigenvalue di atas, selanjutnya kita akan menentukan nilai-nilai eigenvector yang dimiliki setiap pengubah suai yang diramalkan mempunyai kaitan yang erat dengan pencerobohan (Jadual 4.6).

**Jadual 4.6 Komponen Utama dan Eigenvector**

| Variable | Component1 | Component2 | Component3 |
|---|---|---|---|
| **Open** | 0.3791 | 0.0286 | 0.0259 |
| **Read** | 0.3396 | -0.2945 | 0.1796 |
| **Write** | 0.3777 | -0.0808 | 0.0151 |
| **Getgrp** | -0.0122 | -0.0595 | 0.9639 |
| **Setgid32** | 0.3926 | 0.0816 | -0.0382 |
| **Setuid32** | 0.3926 | 0.0820 | -0.0385 |
| **Munmap** | -0.0075 | -0.6694 | -0.0125 |
| **Chown32** | 0.3735 | 0.0583 | -0.0487 |
| **Execve** | 0.3876 | 0.0852 | -0.0.93 |
| **Umask** | 0.0078 | -0.6559 | -0.1751 |

# BAB V

# PROTOKOL PENGURUSAN KEKUNCI KESELAMATAN INTERNET BAGI ALGORITMA PENYULITAN AES DALAM IPSEC

## 5.1 Pengenalan

Internet telah mengubah kebanyakan cara manusia berkomunikasi, berniaga, berhibur, belajar dan sebagainya. Jumlah penggunanya yang senantiasa meningkat menjadikan Internet sebagai satu medium perantaraan yang amat berpotensi untuk mempromosikan perniagaan, penjualan produk dan perkhidmatan, pertukaran surat-surat elektronik, dokumen dan maklumat penting serta penyebaran pelbagai jenis maklumat serta informasi.

Tetapi, sejajar dengan perkembangan teknologi terkini, Internet terdedah kepada pelbagai jenis ancaman penceroboh. Antaranya ialah penafian perkhidmatan, penipuan alamat IP, pengesanan dan kecurian paket data dan juga modifikasi pada data yang dihantar. Maka, dari permasalahan-permasalahan ini, komuniti Internet seluruh dunia

telah mengemukakan beberapa alternatif untuk menyelesaikan masalah ini. Antaranya adalah penggunaan IPSec.

IPsec adalah satu protocol keselamatan IP yang telah dibina oleh sekumpulan penyelidik dari IETF (Internet Engineering Task Force). IETF adalah satu organisasi yang bertanggungjawab menyelaraskan pelbagai aktiviti dalam Internet. IPSec menggunakan pelbagai algoritma kriptografi untuk menghasilkan proses penyulitan dan pengesahan yang berkesan. Ia berfungsi untuk menjaga ketelusan dan kerahsiaan dalam proses penghantaran data. Binaan IPSec direke secara khusus bagi menepati struktur binaan IPV4 dan IPV6. Ini akan memudahkan lagi proses aplikasi IPSec untuk kegunaan masa sekarang dan juga pada masa akan datang.

IPsec mempunyai beberapa kelebihan yang menjadikannya sesuai untuk diadaptasikan ke dalam pelbagai sistem. IPSec sangat fleksibel di mana ia membenarkan penggunaan pelbagai jenis algoritma kriptografi. Ia juga membenarkan pengubahan dilakukan pada polisi dan mekanisma yang digunakan. Ini adalah bertujuan untuk menyesuaikan keadaan persekitaran semasa dengan sistem yang dibina.

Ini bermakna, walaupun IPSec telah menetapkan algoritma-algoritma piawai yang mesti digunakan, namun senibinanya yang fleksibel membolehkan algoritma lain diimplemenkan di dalam IPSec. Oleh itu, menjadi tanggungjawab pereka bagi rekabentuk sistem yang menggunakan algoritma baru, menakrifkan beberapa rangka kerja, mekanisma dan polisi yang mesti dipatuhi, tetapi ia mesti dibuat berdasarkan ketetapan yang telah dibuat oleh IETF.

## 5.2 Rekabentuk & Metodologi

Secara umumnya, metodologi projek ini akan berdasarkan *Formal Discription Technique* (FDT). Penggunaan metodologi yang berasaskan FDT ini akan mewujudkan struktur yang sempurna untuk penghasilan protokol komunikasi dalam OSI (Richard,1984). FDT menyediakan panduan yang lengkap bagi merekabentuk protokol dengan menggunakan pendekatan yang sistematik, analitik dan algoritmatik (Richard,1984).

Antara fasa-fasa yang wujud di dalam FDT adalah:

- Pembahagian perkhidmatan dan elemen protokol
- Penghasilan spesifikasi perkhidmatan
- Penghasilan elemen-elemen protokol berdasarkan spesifikasi perkhidmatan yang dikehendaki
- Penyaringan spesifikasi protokol untuk menguji hubungan elemen protocol dengan spesifikasi yang ditentukan
- Penentuan struktur data

Berikut adalah metodologi bagi projek ini dengan fasa-fasa perlaksanaannya adalah berdasarkan FDT.

(1) Spesifikasi Sistem

- Fasa definasi masalah

- Fasa keperluan sistem dan pengguna
- Fasa mendefinisikan keperluan protokol

(2) Fasa Rekabentuk Protokol

(3) Fasa Penyaringan

- Analisa model protokol

(4) Fasa Simulasi

- Integrasi sistem
- Pengujian sistem
- Penegesahan model

(5) Fasa Implementasi

## 5.3 Rekabentuk Protokol

### 5.3.1 Permodelan Masalah

Bagi fasa ini, masalah telah dimodelkan bagi menyesuaikannya dengan kajian yang akan dilakukan. Secara ringkasnya, kajian ini adalah untuk menghasilkan Protokol Pengurusan Kekunci Set Peraturan Keselamatan (*Internet Security Association Key*

*Management Protocol* - ISAKMP) bagi membolehkan proses penyulitan menggunakan algoritma AES digunakan dalam persekitaran IPSec.

Protokol ini berfungsi sebagai penakrif peraturan dan format paket untuk menghasilkan, membincangkan, mengubahsuai dan menghapuskan Set Peraturan (*Security Association* - SA) dalam IPSec. Melalui kajian ini, didapati, beberapa transformasi pada protokol ini perlu dilakukan bagi membolehkan AES berfungsi dengan sempurna, antaranya adalah dengan melakukan penakrifan DOI (Domain of Interpretation) untuk kegunaan AES, penentuan nilai Identiti Transformasi bagi AES dalam paket muatan Transform dan penghasilan format paket SA untuk AES.

### 5.3.2 Fasa Rekabentuk

- Rekabentuk IPSec

Protokol Keselamatan Internet (IPSec) adalah antara protokol keselamatan yang paling banyak digunakan kerana ia menyediakan pelbagai pilihan ciri keselamatan berbanding protokol lain. IPSec menggunakan beberapa teknologi bagi mewujudkan satu persekitaran rangkaian komunikasi yang selamat. Antara teknologi yang wujud di dalam IPSec adalah:

- Kekunci penukaran Diffie-Hellman
- Tandatangan digital

- Proses penyulitan
- Algoritma kekunci cincangan
- Gabungan Keselamatan *(Security Assosiation*- SA)
- Bidang Penafsiran (*Domain of Interpretation*- DOI)

Masalah yang sering dihadapi dalam menghasilkan IPSec adalah penghasilan DOI. DOI adalah satu kumpulan protokol yang digunakan untuk menghasilkan SA (*Security Association*). SA pula adalah satu entiti yang amat penting dalam penghasilan ISAKMP. Oleh kerana nilai SA ditentukan pengguna, maka ia sukar untuk ditetapkan. Penentuan SA adalah langkah pertama dalam pembinaan IPSec. Sebelum sebarang penghantaran data berlaku antara pihak yang berkomunikasi, kedua-duanya mestilah berbincang untuk mendapatkan nilai SA. Untuk menghasilkan SA, Protokol Pengurusan Kekunci Set Peraturan Keselamatan (ISAKMP) akan digunakan. SA adalah satu set peraturan di mana di dalamnya terdapat pelbagai parameter yang akan digunakan dalam proses komunikasi (Brown, 1999). Semasa proses penentuan SA, parameter-parameter berikut akan ditentukan dengan menggunakan ISAKMP:

- Mode dan kekunci yang digunakan dalam Pangkal Pengesahan (*Authentication Header*- AH)
- Mode dan kekunci yang digunakan dalam Muatan Keselamatan Tersampul (*Encapsulating Security Muatan*- ESP)
- Bilangan pertukaran kekunci
- Penentuan penggunaan Mod Pengangkutan (*Transport Mode*) atau Mod Terowong (Tunnel Mode) untuk proses penghantaran data

- Protokol penyulitan yang akan digunakan
- Penentuan SA
- Jangkamasa komunikasi

Oleh kerana AES adalah algoritma penyulitan baru yang akan digunakan, beberapa parameter dalam SA akan berubah. Ini seterusnya akan mengubah struktur DOI dan ISAKMP. Oleh kerana rekabentuk IPSec adalah fleksibel, perubahan adalah dibenarkan agar perkhidmatan perlindungan keselamatan yang ditawarkan adalah selari dengan kehendak semasa (Maughan *et,al*, 1998).

Namun begitu, perubahan yang dilakukan mestilah mengikut piawai yang telah ditetapkan oleh IETF. Dalam topik seterusnya, akan dibincangkan kajian-kajian yang dijalankan untuk menentukan parameter, mekanisma dan entiti yang akan digunakan oleh AES dalam ISAKMP bagi membolehkannya berfungsi dengan sempurna dalam persekitaran IPSec.

- **Rekabentuk Protokol Pengurusan Kekunci Set Peraturan Keselamatan (*Internet Security Association Key Management Protocol*- ISAKMP)**

ISAKMP menakrifkan rangka kerja bagi pengurusan set peraturan keselamatan dan penghasilan kekunci kriptografi dalam persekitaran Internet. Rangka kerja ini mengandungi takrifan bagi pertukaran data, muatan data dan panduan perlaksanaan yang akan terjadi dengan penggunaan nilai-nilai di dalam DOI (*Domain of Interpretation*).

Di dalam ISAKMP, DOI digunakan untuk mengumpulkan protokol-protokol tertentu untuk mendapatkan nilai SA (*Security Associataion*). Protokol keselamatan yang berkongsi DOI ini kemudiannya akan memilih protokol-protokol keselamatan dan nilai-nilai transform yang akan digunakan dalam proses komunikasi.

SA mesti menyokong pelbagai jenis algoritma penyulitan, mekanisma pengesahan dan algoritma penghasilan kekunci untuk pelbagai jenis protokol keselamatan. Protokol-protokol keselamatan seperti AH dan ESP akan menentukan atribut SA.Antara atribut yang akan ditentukan oleh SA adalah mekanisma pengesahan, algoritma kriptografi, mod algoritma, panjang kekunci dan vektor penentuan (*Initialization Vector*- IV).

Jika pihak yang berkomunikasi ingin menggunakan protokol keselamatan selain dari AH dan ESP, ia mesti menakrifkan atribut SA bagi kegunaannya. Apabila atribut bagi SA telah dicapai, pihak yang berkomunikasi akan mengahsilkan satu nilai penunding yang dinamakan *Security Parameter Index* (SPI).

Penghasilan ISAKMP melibatkan 2 fasa perbincangan yang utama. Pada fasa pertama, server-server ISAKMP akan berbincang dan menyetujui cara untuk melindungi komunikasi antara keduanya. Seterusnya SA akan terhasil.SA ini kemudiannya akan digunakan untuk melindungi perbincangan bagi protokol-protokol yang digunakan.

Pada fasa kedua pula, SA untuk protokol keselamatan lain pula akan dihasilkan. Fasa kedua ini akan digunakan untuk melindungi pelbagai jenis SA. SA yang dihasilkan

oleh ISAKMP pada fasa ini selalunya adalah berdasarkan SA yang dihasilkan pada fasa pertama. SA yang terhasil ini seterusnya akan digunakan oleh protokol keselamatan (contohnya AH dan ESP) untuk melindungi pelbagai jenis mesej dan data.

Terdapat beberapa kelebihan dengan penggunaan dua fasa bagi menghasilkan SA. Antaranya adalah, oleh kerana server-server ISAKMP menggunakan asas yang dipersetujui bersama semasa fasa pertama perbincangan untuk menghasilkan SA fasa kedua, pelbagai SA boleh dihasilkan pada fasa kedua tanpa perlu memulakan kembali proses komunikasi. Sebagai contoh, jika SA yang dihasilkan pada fasa pertama tidak memenuhi kriteria yang ditetapkan, ia akan disempurnakan pada fasa kedua. Ini akan mengurangkan kos operasi dan menjimatkan masa perbincangan.

## 5.4 Analisa Kajian

Dalam bahagian ini, ia menerangkan hasil kajian dan analisi yang telah dibuat terhadap satu prototaip yang dibina untuk membuktikan kejayaan perlaksanaan penggunaan AES dalam persekitaran IPSec menerusi proses penyulitan pada paket data menggunakan protokol pengurusan kekunci set keselamatan Internet yang dicadangkan. Penggunaan protokol ini diyakini akan menyelesaikan masalah-masalah keselamatan dalam rangkaian komputer yang wujud sebelum ini.

Perbincangan terhadap setiap analisa yang telah dibuat dapat dilihat untuk menentukan prestasi penggunaan algortima AES bagi proses penyulitan dalam IPSec. Untuk menghasilkan analisa ini, beberapa ujikaji yang melibatkan algoritma-algoritma terdahulu yang digunakan di dalam IPSec akan dijalankan. Algoritma-algoritma tersebut adalah 3DES, DES, Blowfish dan Cast128. Keempat-empat algoritma ini diimplemenkan menggunakan protokol pengurusan yang telah ditetapkan oleh NIST bagi algoritma masing-masing.

Hasil analisa perbandingan ini akan memberikan gambaran yang jelas mengenai prestasi sebenar algoritma-algoritma tersebut dalam persekitran IPSec. Diharapkan analisa ini dapat dijadikan asas untuk kajian-kajian seterusnya.

### 5.4.1 Pengujian Dan Pengesahan Model Prototaip

Analisis ini dilakukan pada sistem yang dibina. Terlebih dahulu, pengujian dan pengesahan sistem akan dilakukan. Pengujian ini menggunakan perisian Ethereal. Model sistem ini dibina pada dua hos komputer berplatform FreeBSD. Setelah model prototaip berjaya dibangunkan, pengujian akan dilakukan untuk mengesahkan bahawa model tersebut benar-benar berfungsi mengikut spesifikasi yang telah ditetapkan.

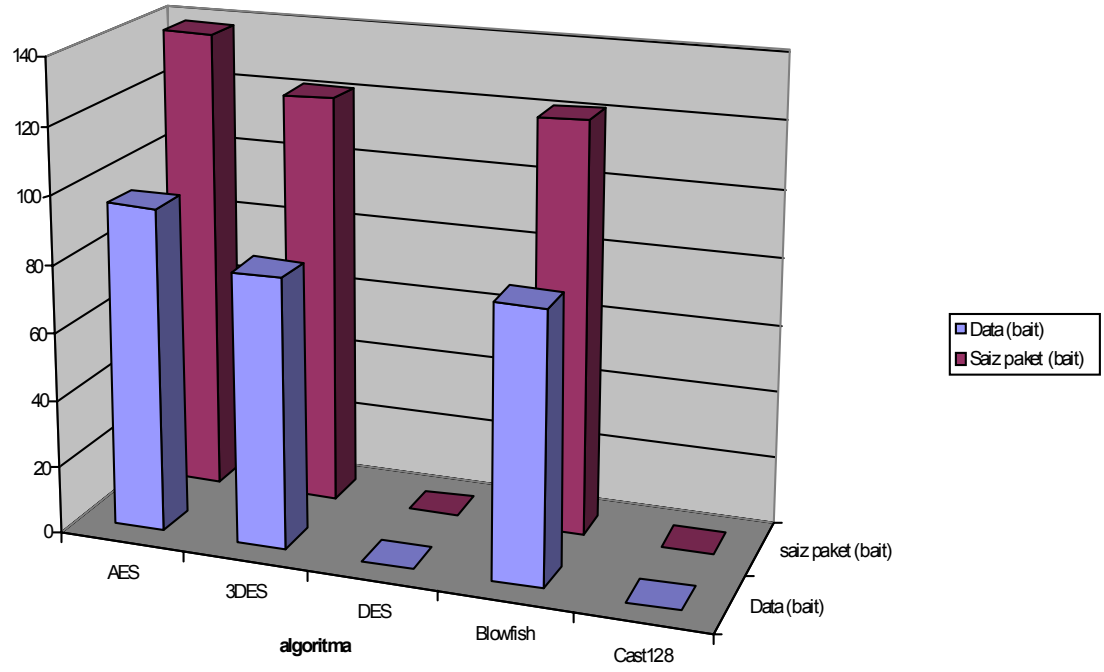
Hos A dan hos B terlebih dahulu akan dikonfigur untuk menghasilkan satu set polisi yang akan digunapakai oleh kedua-dua pihak berdasarkan protokol yang ditentukan. Kemudian kedua-dua hos ini akan ditentukan jenis algoritma penyulitan,

nilai SPI (Security Parameter Index), mod IPSec serta kekunci rahsia. Kedua-dua hos ini mesti mengemukakan satu nilai yang boleh diterima oleh kedua-dua pihak. Jika tidak, kedua-dua hos ini tidak dapat berkomunikasi sepertimana yang dikehendaki dan seterusnya proses interaksi tidak dapat dijalankan..

Hos penghantar kemudiannya akan menghantar data kepada hos penerima. Satu lagi komputer berplatform Windows 98 yang dilengkapi dengan perisian Ethereal akan berfungsi sebagai pemantau dan ia akan melihat pergerakan paket-paket data yang bergerak dalam rangkaian LAN. Pergerakan data-data dari hos A ke hos B dapat dipastikan dengan melihat alamat penghantar dan penerima data. Dari situ, maklumat-maklumat yang berkaitan dengan paket data yang dihantar dapat dilihat dan ini dapat mengesahkan data yang dihantar benar-benar sampai ke destinasi yang dituju dengan membawa maklumat yang berkenaan.

### 5.4.2 Analisis Kefleksibilitian Penggunaan Algoritma AES Dalam Protokol ESP

#### 5.4.2.1 Perbezaan bagi saiz paket dan data yang dihantar menggunakan kekunci 192



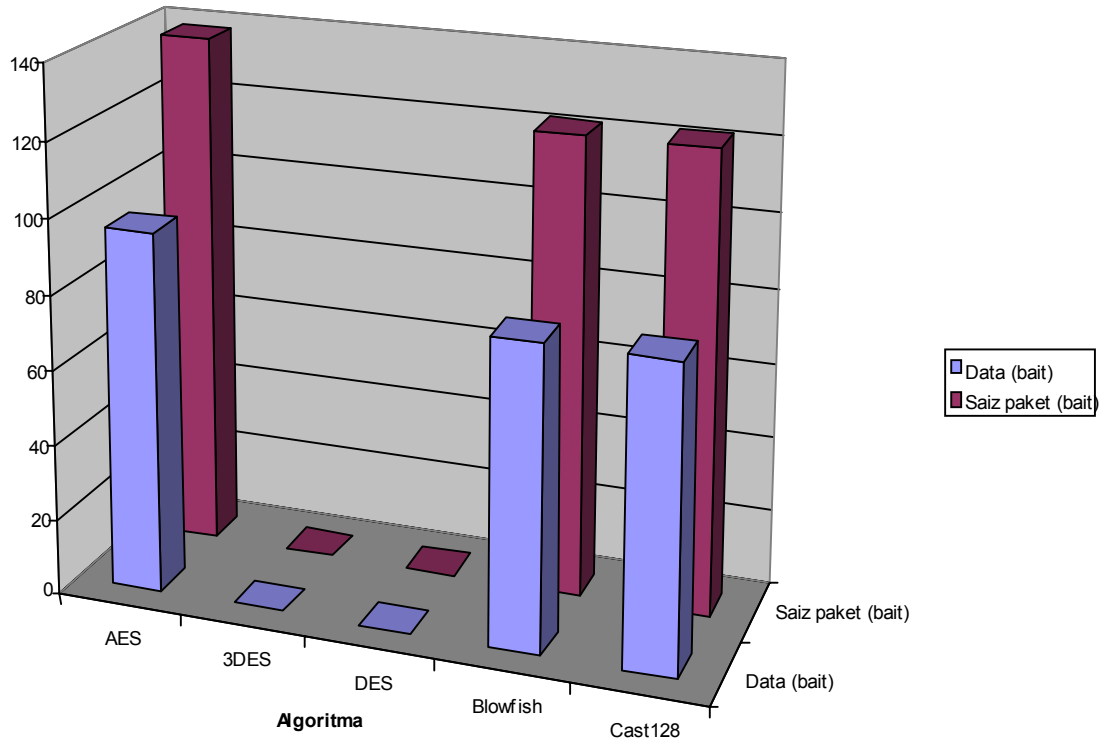

**Rajah 5.1: Graf perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 192**

**Jadual 5.1: Jadual perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 192**

| Algoritma | AES | 3DES | DES | Blowfish | Cast128 |
|---|---|---|---|---|---|
| Data (bait) | 96 | 80 | 0 | 80 | 0 |
| Saiz paket (bait) | 138 | 122 | 0 | 122 | 0 |

Dari rajah dan jadual di atas, dapat dilihat hanya tiga algoritma yang boleh melaksanakan proses penyulitan menggunakan kekunci 192 iaitu AES, 3DES dan Blowfish di mana algoritma AES menunjukkan prestasi terbaik. AES dapat mengenkrip data sebesar 96 bait di mana lebar paket adalah sebesar 138. Manakala kedua-dua algoritma 3DES dan Blowfish dapat mengenkrip data sebesar 80 bait dan lebar paket sebesar 122 sahaja. Perbezaan bagi saiz paket dan data yang dihantar menggunakan kekunci 128. Ini adalah disebabkan hanya algoritma AES, 3DES dan Blowfish sahaja yang dapat menampung kekunci sebesar 192.

### 5.4.2.2 Perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 128



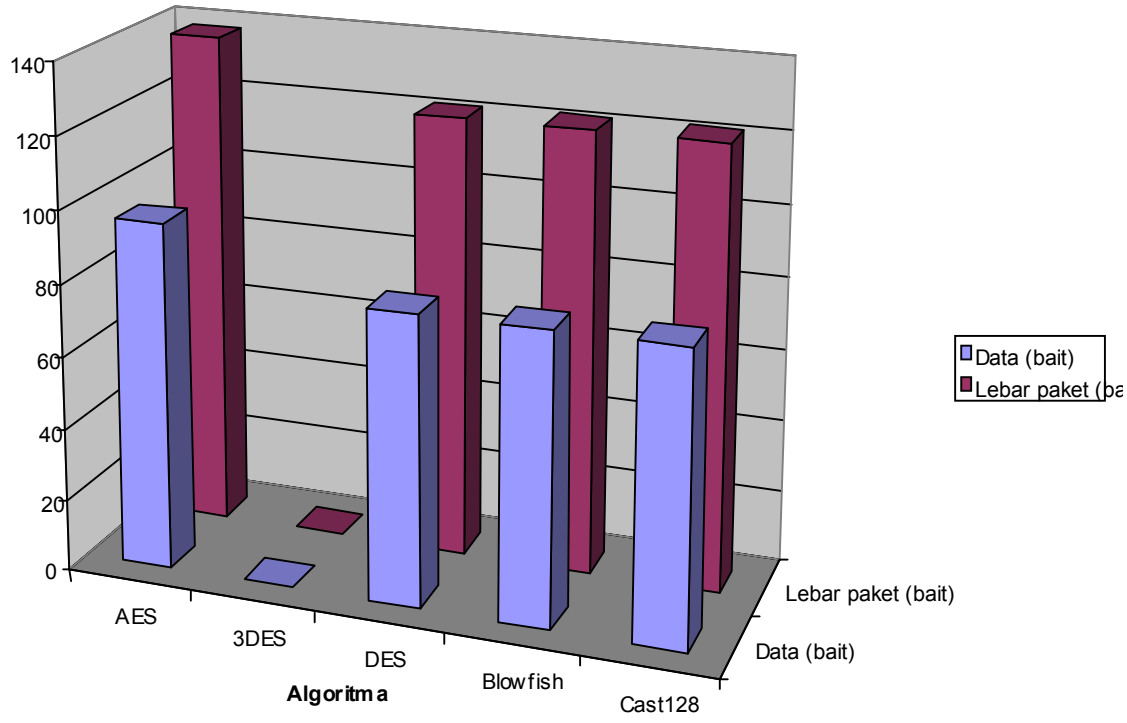

**Rajah 5.2 : Graf perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 128**

**Jadual 5.2 : Jadual perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 128**

| Algoritma | AES | 3DES | DES | Blowfish | Cast128 |
|---|---|---|---|---|---|
| Data (bait) | 96 | 0 | 0 | 80 | 80 |
| Saiz paket (bait) | 138 | 0 | 0 | 122 | 122 |

Bagi penggunaan kekunci bersaiz 128, hanya algoritma AES, Blowfish dan Cast128 yang dapat diimplemenkan di mana AES menunjukkan prestasi terbaik. AES dapat menyulitkan data sebesar 96 bait dengan paket bersaiz 138. Manakala Blowfish dan Cast128 hanya dapat menyulitkan data sebesar 80 bait dengan paket bersaiz 122 saja. Ini bertepatan dengan teori di mana AES, Blowfish dan CAST128 dapat menggunakan kekunci bersaiz 128.

### 5.4.2.3 Perbezaan bagi saiz paket dan data yang dihantar menggunakan kekunci 64




**Rajah 5.3: Graf perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 64**

**Jadual 5.3: Jadual Perbezaan saiz paket dan data yang dihantar menggunakan kekunci bersaiz 64**

| Algoritma | AES | 3DES | DES | Blowfish | Cast128 |
|---|---|---|---|---|---|
| Data (bait) | 96 | 0 | 80 | 80 | 80 |
| Saiz paket (bait) | 138 | 0 | 122 | 122 | 122 |

Dari rajah di atas didapati empat dari lima algoritma yang diuji berjaya menyulitkan data yang dihantar. Algoritma-algoritma tersebut adalah AES, DES, Blowfish dan Cast128. Namun AES masih mengekalkan prestasi terbaik dengan keupayaan mengenkripkan data sebesar 96 dan paket data bersaiz 138, manakala algoritma DES, Blowfish dan CAST128 kesemuanya berupaya menghantar data sebesar 80 bait dengan paket bersaiz 122.

#### 5.4.2.4 Perbincangan

Dari ujikaji ini, dapat dibuktikan bahawa penggunaan AES dalam persekitaran IPSec yang dibina berfungsi dengan sempurna di mana ia memenuhi teori penggunaan saiz kekunci oleh AES. Di samping itu, AES juga menunjukkan prestasi terbaik di mana ia dapat mengenkripkan data pada saiz yang terbesar berbanding algoritma-algoritma yang terlibat dalam kajian.

Di samping itu, AES juga telah menonjolkan prestasi yang terbaik di mana ia berjaya menghasilkan data terenkrip pada saiz yang lebih besar berbanding saiz data terenkrip yang terhasil dari proses penyulitan oleh algoritma-algoritma lain.

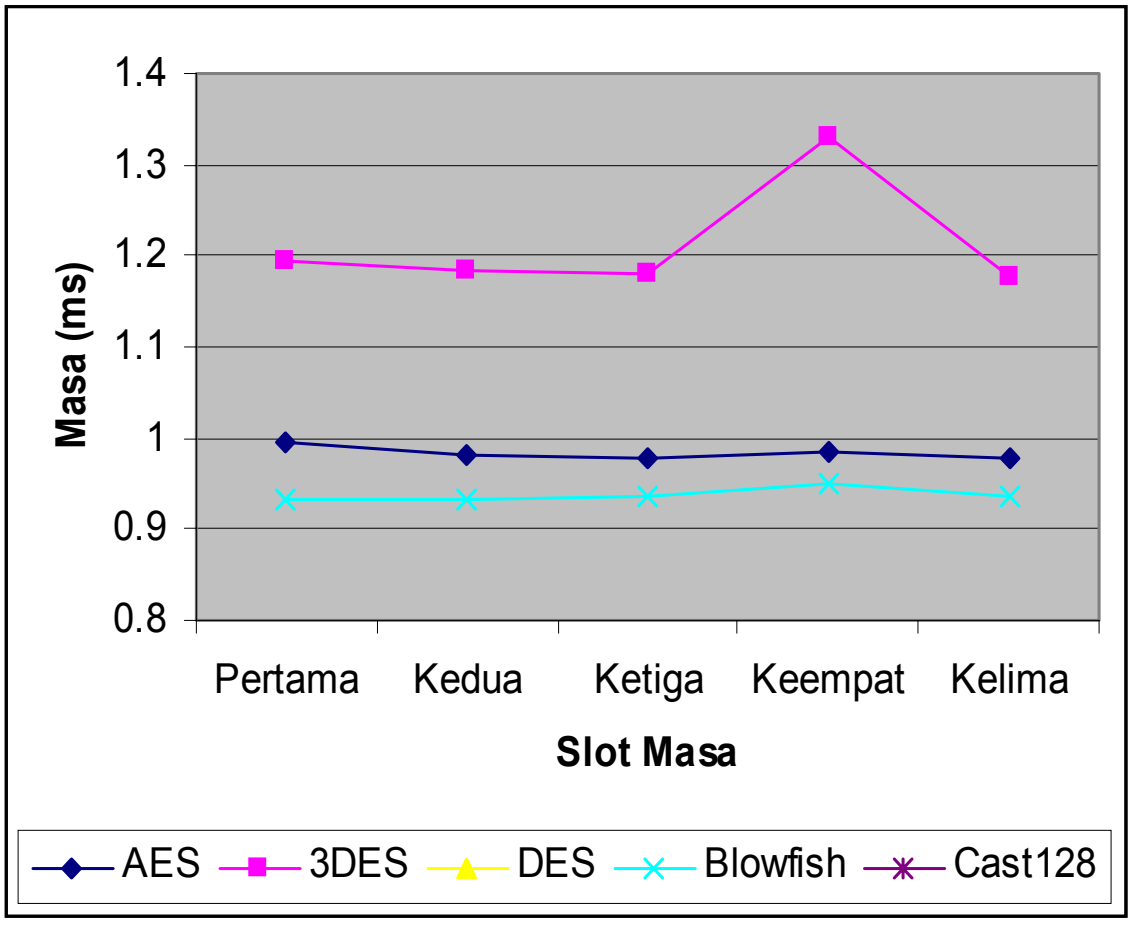
Secara umumnya, ujikaji ini berjalan dengan jayanya di mana kesemua algoritma berfungsi berdasarkan keupayaannya mengikut teori yang ditetapkan. Secara keseluruhan hanya algoritma AES dan Blowfish sahaja yang dapat berfungsi menyulitkan data-data menggunakan kesemua saiz kekunci yang ditetapkan. 3DES

hanya dapat berfungsi dengan penggunaan kekunci bersaiz 192 manakala DES pula hanya berfungsi pada kekunci bersaiz 64. Manakala Cast128 pula dapat mengenkripkan data dengan 2 saiz kekunci iaitu 128 dan 64. Ini menunjukkan AES adalah paling fleksibel di mana ia dapat digunakan pada kekunci pelbagai saiz.

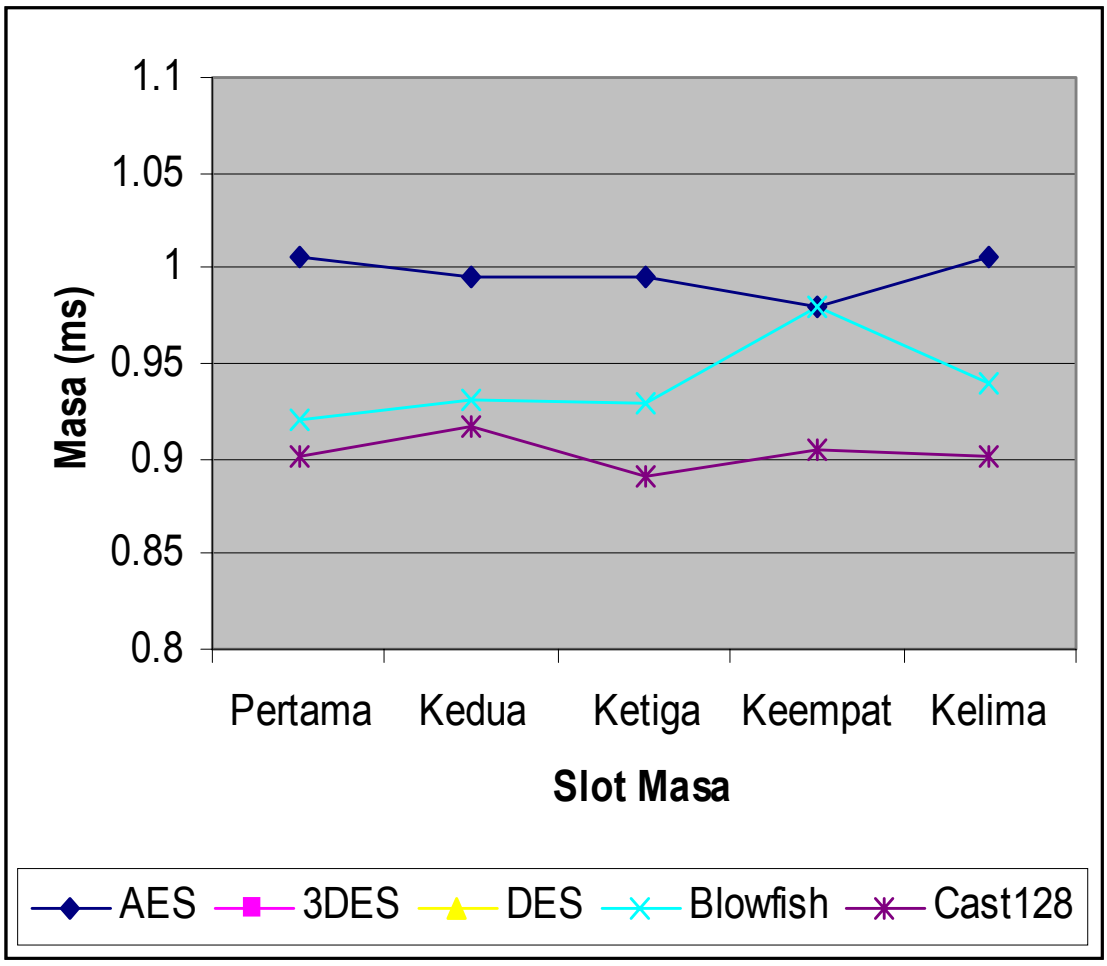
Penggunaan saiz kekunci yang besar amat penting dalam usaha untuk meningkatkan keselamatan bagi data yang dihantar semasa proses komunikasi. Menurut S. Frankel, penggunaan kekunci yang kurang daripada 128 bit patut dihentikan kerana ia sudah tidak mampu menghadapi ancaman keselamatan yang kini semakin hebat. Beliau turut mencadangkan penggunaan kekunci bersaiz 128 bit atau lebih kerana ia lebih sesuai dan efisyen bagi kegunaan semasa (Frankel, 2001).

### 5..4.3 Analisis Kelajuan Penghantaran Data

#### 5.4.3.1 Perbezaan Purata Masa (Ms) Penghantaran Data Bagi Saiz Kekunci Yang Berbeza.




**Rajah 5.4 : Perbezaan purata masa (ms) penghantaran data ICMP bagi kekunci 192**




**Rajah 5.5 : Perbezaan purata masa (ms) penghantaran data ICMP bagi kekunci 128**

**Jadual 5.4: Perbezaan purata masa (ms) penghantaran data bagi saiz kekunci yang berbeza**

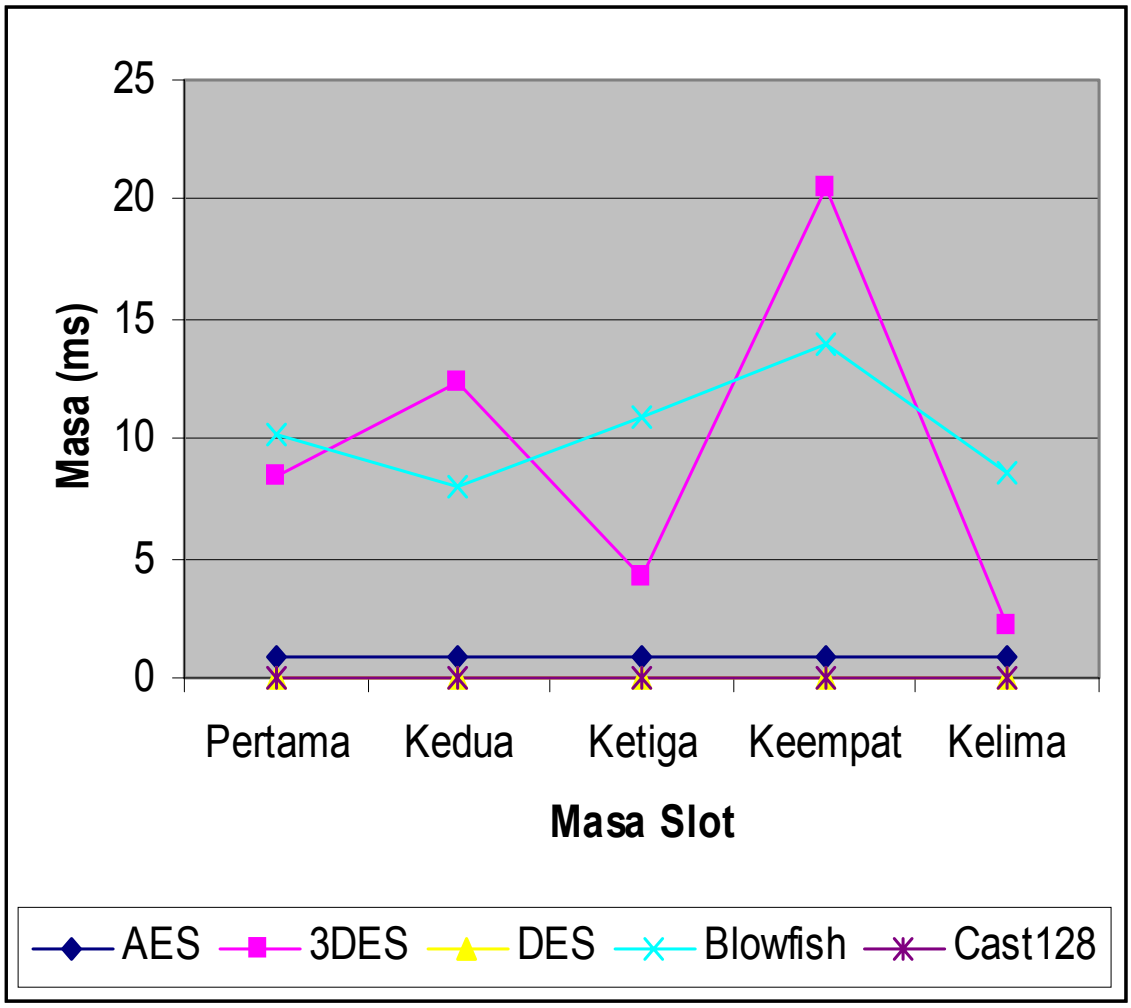
| | | Algoritma | | | | |
|---|---|---|---|---|---|---|
| Kekunci | Slot Masa | AES | 3DES | DES | Blowfish | Cast128 |
| 192 | Pertama | 0.995 | 1.195 | 0 | 0.932 | 0 |
| | Kedua | 0.981 | 1.185 | 0 | 0.934 | 0 |
| | Ketiga | 0.977 | 1.18 | 0 | 0.936 | 0 |
| | Keempat | 0.984 | 1.331 | 0 | 0.949 | 0 |
| | Kelima | 0.978 | 1.176 | 0 | 0.935 | 0 |
| | | | | | | |
| 128 | Pertama | 1.006 | 0 | 0 | 0.92 | 0.902 |
| | Kedua | 0.996 | 0 | 0 | 0.931 | 0.916 |
| | Ketiga | 0.995 | 0 | 0 | 0.929 | 0.891 |
| | Keempat | 0.98 | 0 | 0 | 0.98 | 0.904 |
| | Kelima | 1.005 | 0 | 0 | 0.94 | 0.901 |
| | | | | | | |
| 64 | Pertama | 0.95 | 0 | 1.067 | 0.931 | 0.888 |
| | Kedua | 0.968 | 0 | 1.056 | 0.931 | 0.908 |
| | Ketiga | 1.001 | 0 | 1.025 | 0.932 | 0.888 |
| | Keempat | 1.02 | 0 | 0.885 | 0.937 | 0.922 |
| | Kelima | 0.944 | 0 | 0.824 | 0.937 | 0.888 |

Rajah-rajah di atas menunjukkan perbezaan purata masa dalam milisaat bagi penghantaran data bagi kekunci 192, 128 dan juga 64. Bagi kekunci 192, dapat diperhatikan purata masa penghantaran bagi algoritma-algoritma terlibat adalah sekata

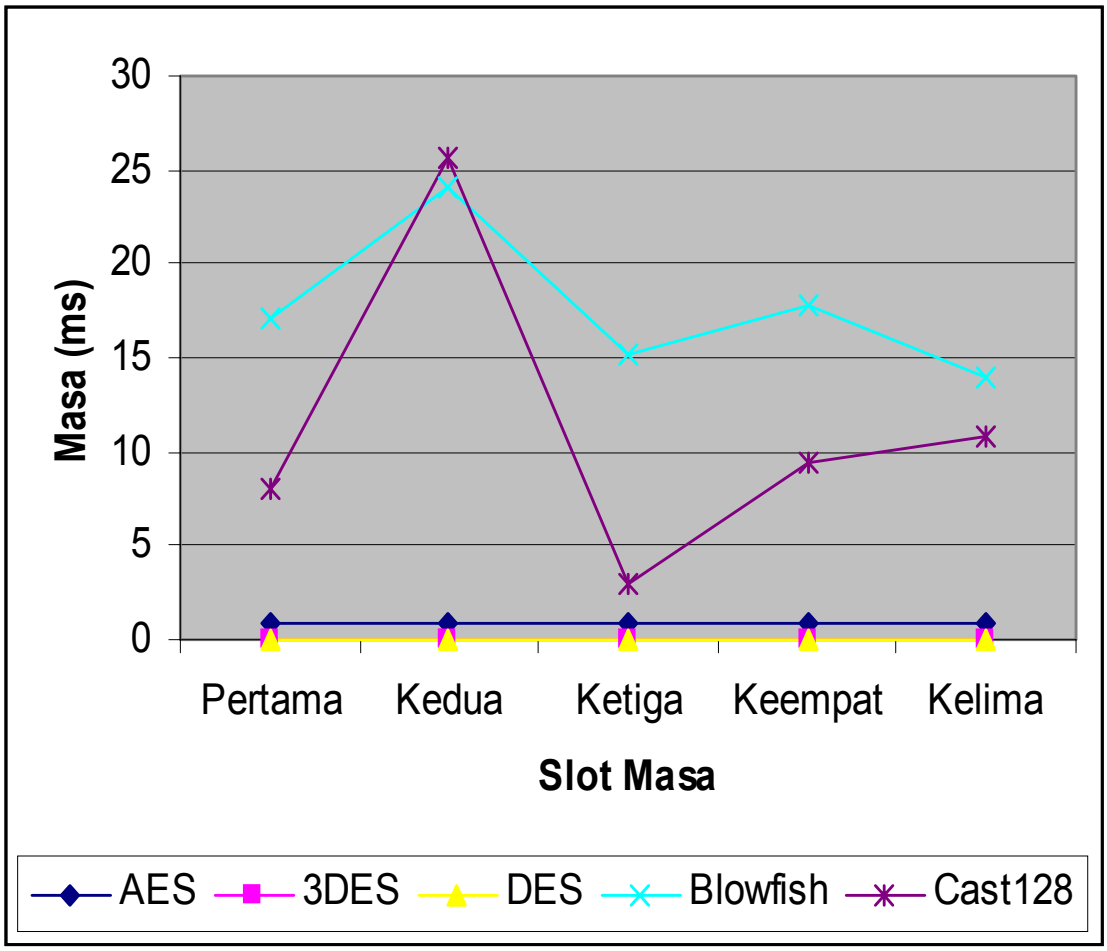
bagi kelima-lima sela masa ujikaji di mana Blowfish menunjukkan prestasi terbaik dengan jumlah purata masa penghantaran sebanyak 0.9372 ms, diikuti oleh AES (0.983 ms) dan 3DES (1.2134 ms). DES dan Cast128 tidak dapat berfungsi dengan penggunaan kekunci bersaiz 192. Begitu juga bagi kekunci 128, purata masa adalah sekata di mana ia berada dalam julat antara 0.9 ms sehingga 1.0 ms. Cast128 adalah terbaik dengan penggunaan masa hanya 0.9028 ms , diikuti oleh Blowfish (0.904 ms) dan AES (0.9964 ms). Algoritma 3DES dan DES tidak dapat berfungsi menggunakan kekunci 128.

Manakala bagi kekunci 64 pula, hanya algoritma 3DES sahaja yang tidak dapat berfungsi. Cast128, Blowfish dan AES menunjukkan purata masa yang sekata, namun sebaliknya bagi DES. Data-data yang dicatatkan oleh DES menunjukkan purata masa yang tidak stabil. Cast128 menunjukkan prestasi terbaik (0.8988 ms) diikuti oleh Blowfish (0.9336 ms), DES (0.9714 ms) dan AES (1.0000 ms).Dengan ini, dapat disimpulkan bahawa, purata masa bagi semua algoritma kajian iaitu AES, 3DES, DES, Blowfish dan Cast128 adalah persis dan berada dalam julat yang lebih kurang sama.

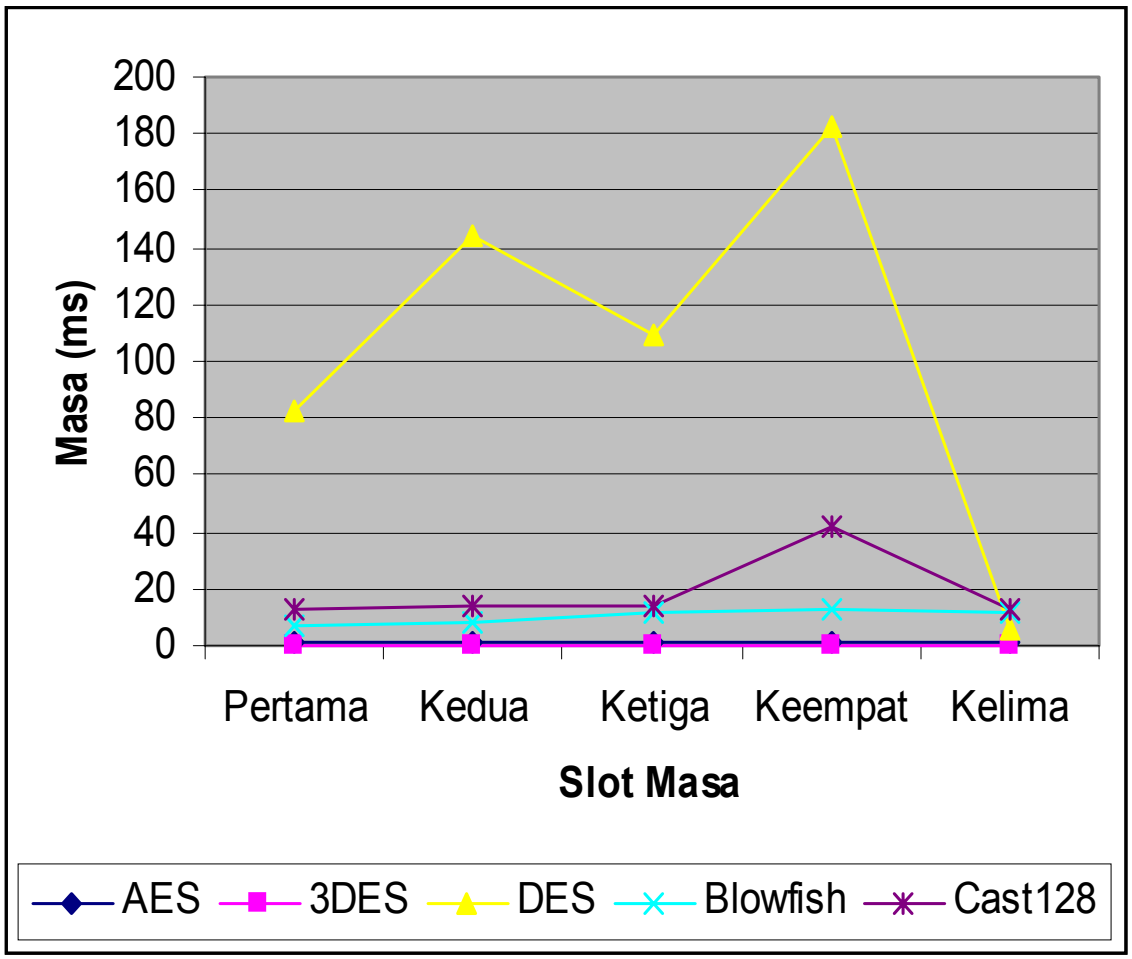
### 5.4.3.2 Perbezaan Masa Maksimum (Ms) Penghantaran Data Bagi Saiz Kekunci Yang Berbeza.




**Rajah 5.6: Perbezaan masa maks(ms) penghantaran data ICMP bagi kekunci 192**




**Rajah 5.7: Perbezaan masa maks (ms) penghantaran data ICMP bagi kekunci 128**




**Rajah 5.8: Perbezaan masa maksimum (ms) penghantaran data ICMP bagi kekunci 64**

| | | Algoritma | | | | |
|---|---|---|---|---|---|---|
| Kekunci | Slot Masa | AES | 3DES | DES | Blowfish | Cast128 |
| 192 | Pertama | 0.925 | 8.492 | 0 | 10.156 | 0 |
| | Kedua | 0.927 | 12.347 | 0 | 8.048 | 0 |
| | Ketiga | 0.93 | 4.258 | 0 | 10.884 | 0 |
| | Keempat | 0.928 | 20.518 | 0 | 14.002 | 0 |
| | Kelima | 0.928 | 2.109 | 0 | 8.547 | 0 |
| | | | | | | |
| 128 | Pertama | 0.922 | 0 | 0 | 17.065 | 8.039 |
| | Kedua | 0.922 | 0 | 0 | 24.012 | 25.608 |
| | Ketiga | 0.92 | 0 | 0 | 15.218 | 3.018 |
| | Keempat | 0.92 | 0 | 0 | 17.773 | 9.363 |
| | Kelima | 0.92 | 0 | 0 | 13.889 | 10.806 |
| | | | | | | |
| 64 | Pertama | 0.911 | 0 | 82.055 | 6.762 | 13.367 |
| | Kedua | 0.915 | 0 | 144.537 | 8.363 | 14.118 |
| | Ketiga | 0.91 | 0 | 109.631 | 12.029 | 13.425 |
| | Keempat | 0.912 | 0 | 182.285 | 12.32 | 42.365 |
| | Kelima | 0.918 | 0 | 5.753 | 11.96 | 12.261 |

**Jadual 5.5: Perbezaan purata masa maksimum (ms) penghantaran data bagi saiz kekunci yang berbeza**

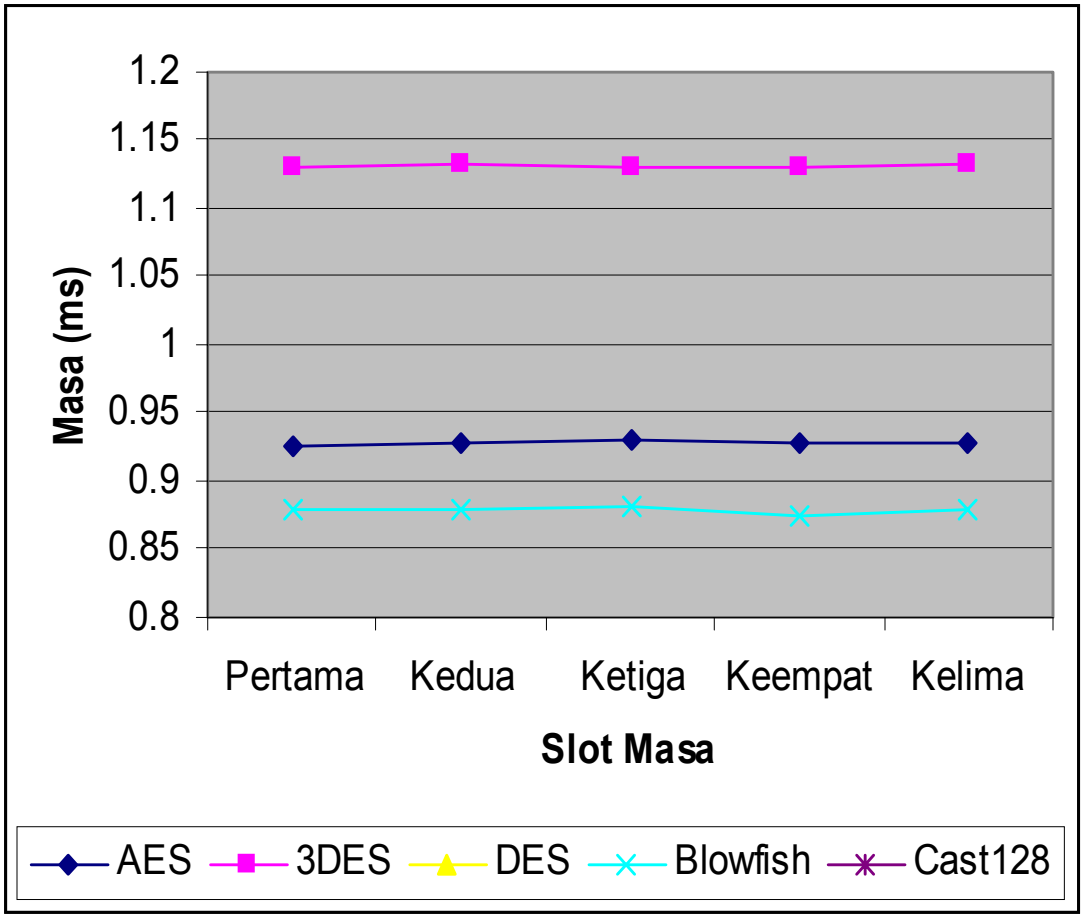
Rajah-rajah di atas menunjukkan masa maksimum yang diambil bagi algoritma-algoritma kajian untuk menghantar paket-paket data dari sumber ke destinasi. Dapat diperhatikan, bagi kekunci 192, 3DES mencatatkan nilai maksimum tertinggi iaitu

20.518 ms iaitu pada slot masa ketiga. Nilai-nilai maksimum masa penghantaran data bagi 3DES juga tidak sekata. Begitu juga bagi Blowfish di mana julat nilai maksimum masa penghantaran datanya adalah antara 8.048 ms sehingga 14.002ms. Berbeza dengan algoritma AES. Ia menunjukkan prestasi yang sangat baik di mana nilai maksimum masa penghantaran datanya adalah terendah antara algoritma-algoritma lain dalam kajian. Nilai tertinggi masa maksima AES adalah 0.93 ms dan julat maksimanya hanya antara 0.925 ms sehingga 0.93 ms sahaja. Ini adalah bacaan yang stabil dan menunjukkan AES dapat berfungsi dalam keadaan yang baik dan pantas dengan penggunaan kekunci bersaiz 198.

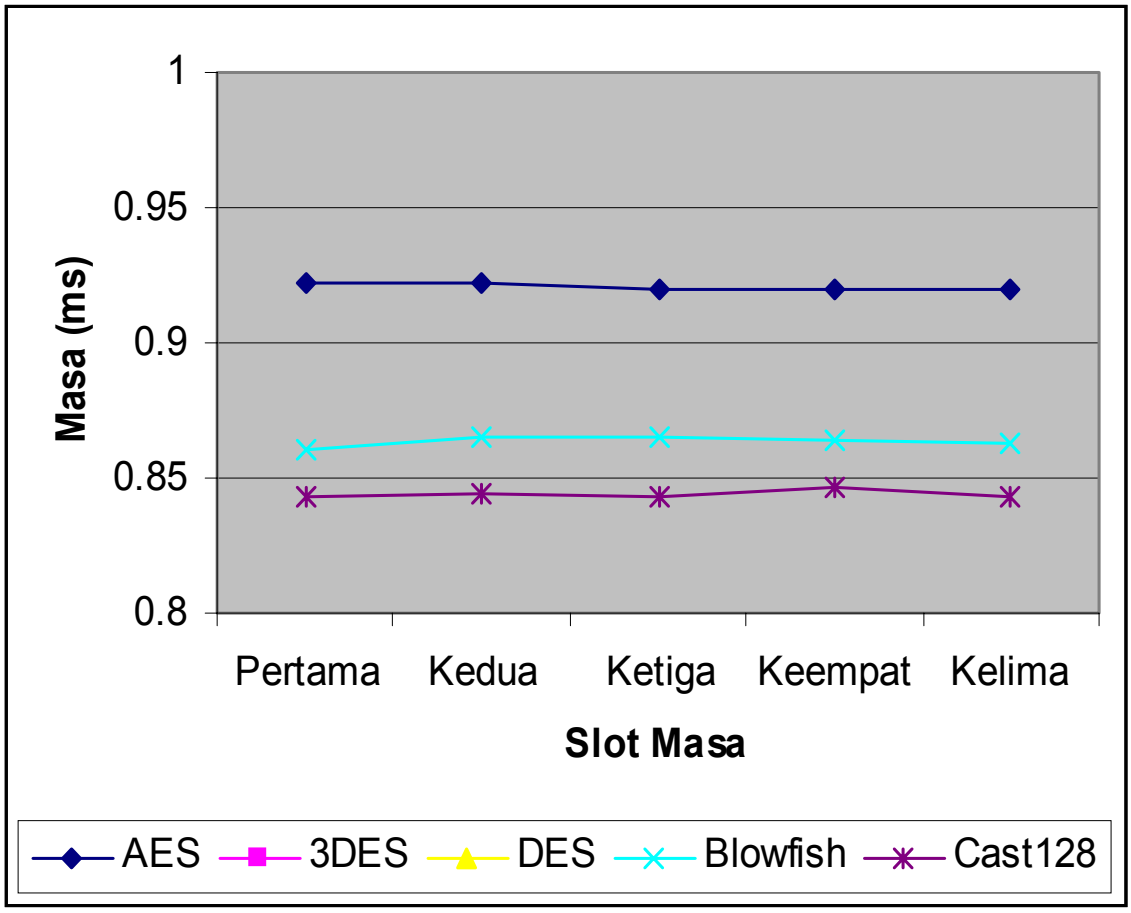
Begitu juga dengan ujikaji bagi penggunaan kekunci saiz 128. Cast128 dan Blowfish menunjukkan nilai maksimum masa penghantaran yang sangat tinggi dengan julat masa yang sangat besar, di mana masa maksimum tertinggi yang dicatatkan adalah oleh Cast128 (25.608 ms). Julat masa bagi Blowfish adalah antara 13.889 ms hingga 24.012 ms manakala bagi Cast128 pula adalah antara 3.018 ms sehingga 25.608 ms. AES masih lagi menunjukkan prestasi terbaik dengan nilai masa maksimum penghantaran data terendah iaitu dalam julat antara 0.92 ms hingga 0.922 ms sahaja. Bagi kekunci 64 pula, DES mencatatkan nilai maksimum tertinggi iaitu 182.285 ms dengan julat masa yang sangat besar iaitu sebesar 177 ms. Nilai kedua tertinggi di catatkan oleh Cast128 iaitu 42.365 ms. Blowfish dan AES menunjukkan julat masa maksimum yang kecil namun AES masih mengatasi Blowfish dengan masa maksimumnya yang sangat rendah iaitu ia berada di dalam julat antara 0.91 hingga 0.918.

Secara keseluruhannya AES menunjukkan prestasi yang sangat baik di mana catatan nilai datanya adalah sangat konsisten dan stabil di samping menunjukkan nilai masa terendah bagi setiap ujikaji. Ini menunjukkan penggunaan AES dalam menghasilkan data tersulit dalam persekitaran Rangkaian Peribadi Maya dapat dilakukan dengan pantas dan cepat berbanding algoritma-algoritma lain.

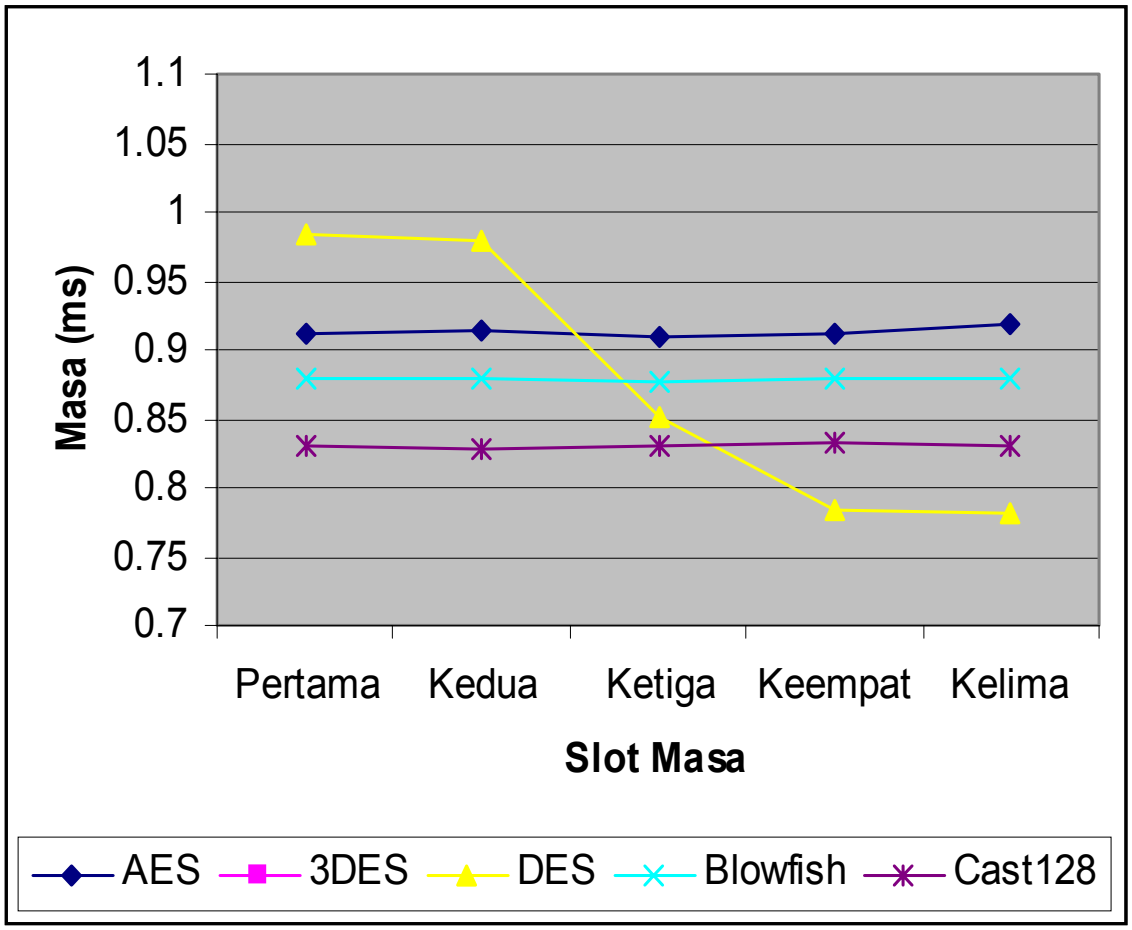
### 5.4.3.3 Perbezaan Purata Masa Minimum (Ms) Penghantaran Data Bagi Saiz Kekunci Yang Berbeza.




**Rajah 5.9: Perbezaan masa min (ms) penghantaran data ICMP bagi kekunci 192**




**Rajah 5.10: Perbezaan masa min (ms) penghantaran data ICMP bagi kekunci 128**




**Rajah 5.11: Perbezaan masa min (ms) penghantaran data ICMP bagi kekunci 164**

| | | Algoritma | | | | |
|---|---|---|---|---|---|---|
| Kekunci | Slot Masa | AES | 3DES | DES | Blowfish | Cast128 |
| 192 | Pertama | 0.925 | 1.131 | 0 | 0.88 | 0 |
| | Kedua | 0.927 | 1.132 | 0 | 0.88 | 0 |
| | Ketiga | 0.93 | 1.131 | 0 | 0.881 | 0 |
| | Keempat | 0.928 | 1.131 | 0 | 0.875 | 0 |
| | Kelima | 0.928 | 1.132 | 0 | 0.88 | 0 |
| | | | | | | |
| 128 | Pertama | 0.922 | 0 | 0 | 0.861 | 0.843 |
| | Kedua | 0.922 | 0 | 0 | 0.865 | 0.844 |
| | Ketiga | 0.92 | 0 | 0 | 0.865 | 0.843 |
| | Keempat | 0.92 | 0 | 0 | 0.864 | 0.847 |
| | Kelima | 0.92 | 0 | 0 | 0.863 | 0.843 |
| | | | | | | |
| 64 | Pertama | 0.911 | 0 | 0.984 | 0.879 | 0.83 |
| | Kedua | 0.915 | 0 | 0.98 | 0.878 | 0.829 |
| | Ketiga | 0.91 | 0 | 0.851 | 0.876 | 0.831 |
| | Keempat | 0.912 | 0 | 0.783 | 0.879 | 0.833 |
| | Kelima | 0.918 | 0 | 0.782 | 0.88 | 0.831 |

**Jadual 5.6: Perbezaan purata masa minimum (ms) penghantaran data bagi saiz kekunci yang berbeza**

Rajah-rajah di atas menunjukkan nilai minimum bagi masa penghantaran data dari hos A ke hos B dalam prototaip yang dibina. Bagi kekunci 192, nilai terendah masa

minimum penghantaran data adalah dicatatkan oleh Blowfish dengan purata masa minimumnya adalah 0.8792 ms, diikuti oleh AES (0.9276 ms) dan 3DES (1.1314 ms). Ketiga-tiga algoritma ini menunjukkan purata minimum masa penghantaran data yang sekata.

Begitu juga bagi kekunci 128, nilai-nilai masa minimum yang dicatatkan adalah sekata dengan nilai purata minimum yang paling rendah adalah Cast128 (0.844 ms) diikuti oleh Blowfish (0.8636 ms) dan AES (0.9208 ms). Bagi kekunci 64 pula, AES, Blowfish dan Cast128 menunjukkan nilai-nilai masa minimum yang sekata manakala DES pula menunjukkan nilai-nilai masa minimun yang agak tidak stabil. Masa minimum terendah dicatatkan oleh Cast128 dengan nilai purata masa yang diambil adalah 0.8308 ms, seterusnya DES (0.816 ms), Blowfish (0.8784 ms) dan AES (0.9132 ms).

Daripada graf-graf yang dihasilkan, dapat disimpulkan bahawa masa minimum bagi penghantaran data terenkrip menggunakan algoritma AES, 3DES, DES, Blowfish dan Cast128 adalah berada dalam julat yang lebih kurang sama.

#### 5.4.3.4 Perbincangan

Dari ujikaji yang dijalankan, dapat dipastikan bahawa hanya algoritma AES dan Blowfish sahaja yang dapat berfungsi pada kesemua ketiga-tiga saiz kekunci. Cast128 pula hanya dapat berfungsi pada kekunci bersaiz 128 dan 64. Manakala 3DES hanya

dapat berfungsi pada kekunci 192 sementara DES pula hanya pada kekunci 64. Ini membuktikan teori saiz kekunci yang dibenarkan oleh setiap algoritma.

Bagi AES, ia mampu menampung kekunci bersaiz 128,192 dan 256 bit dan mana-mana kekunci dalam gandaaan 32 contohnya 32, 64, 96, 128 dan seterusnya. Jadi ia dapat berfungsi pada semua saiz kekunci yang digunakan dalam ujikaji iaitu 64, 128 dan 192. Bagi 3DES pula ia hanya boleh menggunakan kekunci bersaiz 192 bit dan DES pula, hanya boleh menggunakan kekunci 64 bit sahaja. Bagi Blowfish, ia membenarkan penggunaan pelbagai saiz kekunci dari 32 bit sehingga 256 bit. Ini bermaksud, ia meliputi kesemua saiz kekunci yang digunakan di dalam ujikaji. Manakala bagi Cast128 pula, saiz kekuncinya boleh dipilih dalam julat 40 bit sehingga 128 bit di mana ia membenarkan penggunaan kekunci pada gandaaan 8. Sebagai contoh 40, 48, 56, 64, …, 112, 120 dan 128 bit. Jadi, dalam ujikaji yang dijalankan hanya proses penyulitan menggunakan kekunci 64 dan 128 bit sahaja yang dapat digunakan.

Secara keseluruhan AES menunjukkan prestasi terbaik di mana ia mencatatkan purata masa penghantaran data yang sekata bagi kesemua saiz kekunci iaitu kekunci 192, 128 dan kekunci 64 di mana julat penghantaran data bagi masa purata, masa maksimum dan masa minimumnya adalah antara 0.9 ms sehingga 1.0 ms sahaja. Jika dibandingkan dengan Blowfish, julat masa yang diambil untuk menghantar data bagi masa purata, maksimum dan minimum adalah lebih besar iaitu antara julat 0.8 ms sehingga 18.0 ms. Begitu juga bagi Cast128.

Bagi 3DES, prestasi penggunaannya pada tahap kekunci bersaiz 192 adalah berada dalam tahap purata (average) dengan algoritma-algoritma lain. Begitu juga dengan DES yang berfungsi hanya pada tahap kekunci bersaiz 64. Prestasinya juga adalah dalam tahap purata.

Ini mengesahkan penggunaan protokol pengurusan kekunci dan set peraturan Internet yang dirangka berfungsi dengan sempurna dan lancar. AES juga telah membuktikan penggunaannya dalam persekitaran IPSec akan menghasilkan satu proses penyulitan data yang dapat dilakukan dengan pantas. Ini dapat dibuktikan dengan memerhatikan nilai-nilai data ujikaji. Didapati masa maksima penghantaran data oleh AES adalah paling minima sekali jika dibandingkan dengan algoritma-algoritma lain dalam kajian. Bagi nilai penghantaran data masa minimum dan purata pula AES berada di tempat kedua selepas Cast128, namun Cast128 hanya dapat berfungsi pada tahap kekunci bersaiz 64 dan 128 sahaja.

### 5.4.4 Analisis Pengukuran Prestasi Truput Dengan Penggunaan Algoritma AES Dan Algoritma Penyulitan Lain Dalam Persekitaran IPSec

Berikut adalah laporan hasil analisa bagi beberapa siri eksperimen yang dilakukan untuk mengukur prestasi TCP dan UDP dengan penggunaan algoritma penyulitan yang digunakan dalam proses penyulitan data dalam model IPSec yang dibina. Dalam analisa ini, satu program pengukuran prestasi yang dinamakan TTCP (Test TCP) digunakan.

TTCP digunakan untuk mengukur truput TCP yang melalui satu laluan IP. Truput adalah kebolehan pemprosesan Unit Pemprosesan Pusat (UPP). Ia merujuk kepada bilangan proses yang telah atau dapat diselesaikan dalam satu unit masa. Truput juga bergantung kepada saiz proses yang terlibat. Jika proses adalah bersaiz besar, maka sedikit proses yang dapat diselesaikan dalam satu unit masa tertentu berbanding jika proses tersebut bersaiz kecil.

Menurut David G. Andersen ( Anderson, D.G., 2001) nilai truput yang diperolehi dari program TTCP boleh juga diklasifikasikan sebagai masa yang diambil untuk proses penyulitan dan penyahsulitan paket data yang dihantar.

Untuk menggunakan program ini, hos penerima dan hos penghantar mestilah berada pada satu laluan IP.Hos penghantar akan menghantar beberapa paket data kepada hos penerima. Diakhir siri ujian, kedua-dua hos akan memaparkan bilangan bait yang dihantar dan masa yang diambil untuk paket itu dihantar dari hos penghantar ke

hos penerima. Ciri ini juga boleh digunakan untuk menguji kelajuan komunikasi antara dua hos yang yang mempunyai sambungan IP antara keduanya.

Proses penghantaran data tersebut akan melalui proses penyulitan data yang terhasil dari penggunaan algoritma AES berdasarkan protokol pengurusan yang dibina pada persekitaran IPSec. Selain daripada AES, algoritma-algoritma lain yang akan digunakan sebagai pembanding adalah DES, 3DES, Blowfish dan Cast128.

Ujikaji akan dilakukan ke atas 2 hos yang berbeza kelajuannya. Hos pertama berkelajuan 300MHz manakala hos kedua berkelajuan 200MHz. Di samping itu, saiz penimbal yang akan digunakan adalah 7000, 8192, 9000 dan 10 000 di mana saiz penimbal yang piawai adalah 8192.

#### 5.4.4.1 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 64 (CPU 300MHz)

**Jadual 5.7: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 64 bagi hos berkelajuan 300MHz**

Kekunci: 64

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | **15.14** | **17.92** | **20.91** | **21.54** |
| 3DES | tiada | tiada | tiada | tiada |
| CAST128 | 18.83 | 22.04 | 23.99 | 26.65 |
| BLOWFISH | 20.49 | 21.87 | 23.99 | 27.22 |
| AES | 18.46 | 21.88 | 24.19 | 26.45 |

Berikut adalah analisis prestasi truput bagi semua saiz penimbal bagi hos penghantar yang berkelajuan 300MHz dengan menggunakan kekunci bersaiz 64. Bagi penimbal bersaiz 7000, nilai truput bagi DES adalah 15.14ms berbanding AES (18.46ms), Cast128 (18.83ms) dan Blowfish (20.49ms). 3DES pula tidak berfungsi gdengan penggunaan kekunci bersaiz 64.

Bagi penimbal bersaiz 8192, nilai truput DES ialah 17.92ms, manakala penggunaan DES dengan penimbal bersaiz 9000 pula adalah sebanyak 20.19ms dan bagi penimbal bersaiz 10000 ia mengambil masa 21.54ms. Manakala Cast128, Blowfish dan

AES pula, untuk penimbal bersaiz 8192, ketiga-tiga algoritma ini mengambil masa antara 21 hingga 22 ms, penimbal bersaiz 9000 mengambil masa antara 23 hingga 24 ms dan bagi penimbal bersaiz 10 000, ia mengambil masa antara 26 hingga 27 ms.

Secara keseluruhannya, bagi kekunci bersaiz 64, DES menunjukkan prestasi terbaik bagi keempat-empat saiz penimbal iaitu 7000, 8192, 9000 dan 10 000.

#### 5.4.4.2 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 128 (CPU 300MHz)

**Jadual 5.8: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 128 bagi hos berkelajuan 300MHz**

kekunci: 128

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | tiada | tiada | tiada | tiada |
| 3DES | tiada | tiada | tiada | tiada |
| CAST128 | 18.68 | 22.51 | 24.5 | 26.98 |
| BLOWFISH | **18.66** | 21.77 | **24.09** | 26.86 |
| AES | 95.41 | **18.67** | 24.35 | **18.24** |

Di dalam analisis ini, didapati DES dan 3DES tidak dapat berfungsi dengan penggunaan kekunci bersaiz 128. Dengan penggunaan penimbal sebesar 7000, algoritma

Blowfish menunjukkan prestasi terbaik di mana purata nilai truputnya adalah 18.66 ms, berbanding dengan Cast128 dan AES yang mana masing-masing purata truputnya adalah 18.68 ms dan 95.41 ms.

Bagi penggunaan penimbal bersaiz 8192 pula AES menunjukkan prestasi terbaik dengan purata nilai truputnya adalah 18.67 ms berbanding Blowfish (21.77 ms) dan Cast128 (22.51 ms). Namun, bagi penimbal bersaiz 9000, kebanyakan algoritma menunjukkan prestasi yang lebih kurang sama, namun Blowfish agak menonjol sedikit dengan purata nilai truput adalah 24.09 ms diikuti oleh AES (24.35 ms) dan Cast128 (24.5 ms). Berbeza pula bagi penggunaan penimbal bersaiz 10 000 di mana AES adalah algoritma terbaik dengan purata nilai truputnya adalah 18.24 ms. Ini diikuti oleh Blowfish (26.86 ms) dan Cast128 (26.98 ms).

Secara keseluruhan, Cast128, Blowfish dan AES menunjukkan prestasi yang setara antara satu sama lain. Namun, bagi penimbal bersaiz 7000 dan 9000, Blowfish lebih menonjol manakala bagi penimbal bersaiz 8192 dan 10 000 pula, AES yang lebih menonjol.

### 5.4.4.3 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 192 (CPU 300MHz)

**Jadual 5.9: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 192 bagi hos berkelajuan 300MHz**

Kekunci: 192

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | tiada | tiada | tiada | tiada |
| 3DES | 33.12 | 39.06 | 42.63 | 47.25 |
| CAST128 | tiada | tiada | tiada | tiada |
| BLOWFISH | 18.73 | 22.08 | 24.62 | 26.84 |
| AES | **16.36** | **19.35** | **21.07** | **23.84** |

Dengan penggunaan kekunci bersaiz 192, algoritma DES dan Cast128 tidak dapat berfungsi. Namun secara keseluruhannya, bagi penggunaan kekunci bersaiz 192, AES menunjukkan prestasi terbaik bagi keempat-empat saiz penimbal yang digunakan. Bagi penimbal bersaiz 7000, AES secara puratanya nilai truputnya adalah 16.36 ms diikuti oleh Blowfish (18.73) ms dan 3DES (33.12 ms).

Bagi penimbal bersaiz 8192 pula, purata masa truput oleh AES adalah 19.35 ms diikuti oleh Blowfish (22.08 ms), dan 3DES (39.06 ms). Sementara itu, bagi penimbal bersaiz 9000, AES purata nilai truputnya adalah 21.07 ms, manakala Blowfish pula

selama 24.62 ms dan 3DES 33.12 ms. Bagi penimbal bersaiz 10 000, nilai truput oleh AES adalah 23.84 ms, diikuti oleh Blowfish (26.84 ms) dan 3DES (47.25 ms).

Secara keseluruhannya, dengan penggunaan kekunci bersaiz 256, prestasi AES adalah yang terbaik diikuti oleh Blowfish seterusnya 3DES. Namun, prestasi 3DES tidak begitu memuaskan di mana nilai truputnya adalah lebih kurang dua kali ganda masa truput yang diambil oleh AES.

#### 5.4.4.4 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 64 (CPU 200MHz)

**Jadual 5.10: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 64 bagi hos berkelajuan 200MHz**

Kekunci: 64

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | **15.64** | **17.91** | **20.89** | **21.53** |
| 3DES | tiada | tiada | tiada | tiada |
| CAST128 | 18.82 | 22.02 | 23.98 | 26.65 |
| BLOWFISH | 16.34 | 21.85 | 23.98 | 27.21 |
| AES | 18.63 | 21.84 | 24.12 | 26.59 |

Dengan penggunaan kekunci bersaiz 64, secara keseluruhannya DES menunjukkan prestasi terbaik bagi semua saiz penimbal. Namum 3DES pula tidak dapat berfungsi dengan penggunaan kekunci bersaiz 64. Penimbal bersaiz 7000 menghasilkan nilai truput paling cepat bagi kesemua algoritma diikuti oleh penimbal bersaiz 8192, 9000 dan 10,000. Ini bermakna masa yang diperlukan untuk melaksanakan proses penghantaran data bagi penggunaan kekunci bersaiz 64 adalah berkadaran langsung dengan saiz penimbal yang digunakan.

Bagi penimbal bersaiz 7000, DES purata nilai truputnya adalah 15.64 ms, diikuti oleh Blowfish (16.34 ms), AES (18.63ms) dan Cast128 (18.82 ms). Untuk penimbal bersaiz 8192, DES mendahului dengan purata masa 17.91 ms diikuti oleh AES (21.84 ms), Blowfish (21.85 ms) dan Cast128 (22.02 ms). Sementara bagi penimbal bersaiz 9000, Des mencatatkan purata masa nilai truput sebanyak 20.89 ms, diikuti oleh Cast128 dan Blowfish (23.98 ms) dan AES (24.12 ms). Bagi penimbal bersaiz 10000 pula, masa yang diambil oleh DES adalah 21.53 ms, AES 26.59 ms, Cast128 26.65 ms dan Blowfish 27.21 ms.

### 5.4.4.5 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 128 (CPU 300MHz)

**Jadual 5.11: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 128 bagi hos berkelajuan 300MHz**

kekunci: 128

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | tiada | tiada | tiada | tiada |
| 3DES | tiada | tiada | tiada | tiada |
| CAST128 | 18.66 | 22.5 | 24.48 | 26.97 |
| BLOWFISH | **18.65** | 21.75 | **24.08** | 26.86 |
| AES | 19.07 | **17.65** | 24.34 | **18.24** |

Penggunaan kekunci bersaiz 124 hanya boleh digunakan oleh algoritma Cast128, Blowfish dan AES. Bagi penimbal bersaiz 7000, Blowfish menunjukkan prestasi terbaik di mana purata nilai truputnya adalah 18.65 ms sahaja diikuti oleh Cast128 (18.66 ms) dan AES (19.07ms).

Sementara bagi penimbal bersaiz 8192 pula, AES menunjukkan prestasi yang terbaik dengan catatan purata nilai truput selama 17.65 ms diikuti oleh Blowfish (21.75 ms) dan Cast128 (22.5 ms).Bagi penimbal bersaiz 9000 pula, Blowfish menghasilkan nilai truput paling pantas (24.08 ms), diikuti oleh AES (24.34 ms) dan Cast128 (24.48 ms). Untuk penimbal bersaiz 10 000 pula AES menunjukkan bacaan purata masa yang

sangat pantas berbanding algoritma lain di mana ia hanya mengambil masa selama 18.24 ms berbanding Blowfish (26.86 ms) dan Cast128 (26.97 ms).

Didapati, purata masa yang diambil untuk proses penghantaran data adalah berkadaran langsung dengan saiz penimbal yang digunakan.

#### 5.4.4.6 Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 256 (CPU 200MHz)

**Jadual 5.12: Purata nilai truput bagi kesemua saiz penimbal dengan menggunakan kekunci bersaiz 256 bagi hos berkelajuan 200MHz**

kekunci: 256

| Algoritma\penimbal | 7000 | 8192 | 9000 | 10000 |
|---|---|---|---|---|
| DES | tiada | tiada | tiada | tiada |
| 3DES | 33.1 | 39.05 | 42.61 | 47.22 |
| CAST128 | tiada | tiada | tiada | tiada |
| BLOWFISH | 18.72 | 22.07 | 24.14 | 26.84 |
| AES | **16.35** | **19.34** | **21.06** | **23.84** |

Bagi penggunaan kekunci 256, hanya 3DES, Blowfish dan AES sahaja yang dapat digunakan. Dengan penggunaan penimbal bersaiz 7000, AES mendahului algoritma lain dengan catatan masa nilai truput sebanyak 16.35 ms berbanding Blowfish

(18.72 ms) dan 3DES (33.1 ms). Bagi penimbal bersaiz 8192, AES juga menunjukkan prestasi terbaik (19.34 ms) diikuti oleh Blowfish (22.07 ms) dan 3DES (39.05 ms). Begitu juga bagi penimbal bersaiz 9000, AES menunjukkan catatan masa nilai truput terpantas iaitu 21.06 ms, diikuti seterusnya oleh Blowfish (24.14 ms) dan 3DES (42.61 ms). Bagi penimbal bersaiz 10 000 pula, catatan terbaik adalah oleh AES (23.84 ms), diikuti oleh Blowfish (26.84 ms) dan 3DES (47.22 ms).

#### 5.4.4.7 Perbincangan

Secara keseluruhannya, perbezaan antara purata masa yang diambil untuk proses penghantaran data oleh algoritma-algoritma dalam kajian antara hos yang berkelajuan 300 MHz dengan hos yang berkelajuan 200 MHz adalah sangat sedikit iaitu dalam julat antara 0.00 ms hingga 1.02 ms. Begitu juga dengan paten prestasi algoritma-algoritma penyulitan yang digunakan di dalam kajian. Prestasi yang ditunjukkan oleh algoritma-algoritma pada hos 200 MHz adalah sama dengan prestasi algoritma-algoritma pada hos 300 MHz.

Didapati juga, masa yang diambil untuk menghantar data adalah berkadaran secara langsung dengan saiz penimbal yang digunakan. Daripada ujikaji-ujikaji yang dijalankan, semakin besar saiz penimbal yang digunakan, semakin banyak masa yang diambil untuk menghantar data. Ini bermakna, penggunaan penimbal yang besar akan mengurangkan nilai truput sistem.

Bagi penggunaan kekunci bersaiz 64, algoritma DES menunjukkan prestasi terbaik dengan penggunaan keempat-empat saiz penimbal. Bagi penggunaan kekunci bersaiz 128 pula, algoritma Blowfish menunjukkan prestasi terbaik dengan penggunaan penimbal bersaiz 7000 dan 9000, manakala algoritma AES berprestasi terbaik dengan penggunaan penimbal bersaiz 8192 dan 10 000.

Dari ujikaji ini, didapati bahawa truput sistem bagi penggunaan algoritma AES adalah paling baik dengan penggunaan kekunci bersaiz 192. Nilai truputnya akan berkurangan dengan berkurangnya saiz kekunci. Ini menunjukkan AES berfungsi pada tahap yang optima dengan penggunaan kekunci bersaiz 192.

# BAB VI

# PENUTUP

## 6.1 Perbincangan dan Kesimpulan

Salah satu isu yang amat penting di dalam E-Dagang ialah kebimbangan pengguna terhadap keselamatannya. Ramai pengguna masih ragu-ragu terhadap keselamatan transaksasi melalui talian dimana terdapat kemungkinan maklumat peribadi mereka serta nombor kad kredit akan dipintas oleh penggodam sistem komputer. Namun dengan kemunculan teknologi seperti SET, SSL serta peningkatan firewall, pengguna tidak perlu bimbang lagi tentang keselamatan semasa melakukan pembayaran. Teknologi SET contohnya memerlukan pembeli, penjual dan pihak bank berdaftar dengan badan persijilan tertentu bagi mengesahkan identiti mereka semasa melakukan urusniaga. Pengesahan ini akan dilakukan secara luar talian seperti melalui telefon, faks dan sebagainya.

Di dalam projek ini, kumpulan penyelidikan telah mengupas secara mendalam tiga kategori utama yang melibatkan isu keselamatan di dalam E-Dagang ini. Ianya tertumpu kepada keselamatan pelayan dan laman web, keselamatan data yang sah pada pelanggan mahupun pelayan dan keselamatan penghantaran data pada lapisan rangkaian.

Di dalam penyelidikan mengenai keselamatan pelayan dan laman web, *Web Document Integrity Detector* (WebDID) telah dibangunkan sebagai satu alternatif untuk mengesan pencerobohan ke atas laman web pada pelayan web dan memberi amaran pecerobohan kepada pentadbir pelayan web. Penyelidikan ini telah membuktikan bahawa penggunaan fungsi cincang iaitu RIPEMD-160 dapat mengesan pencerobohan melalui pengubahsuai dengan cara membandingan nilai cincangan yang dihasilkan sebelum dan selepas pencerobohan. RIPEMD-160 dipilih atas sebab keselamatan yang dimiliki berbanding fungsi cincang yang lain. *Cryptix 3.2* adalah perpustakaan keselamatan yang diimport untuk mendapatkan algoritma cincangan ini. Ia mengandungi pelbagai jenis algoritma cincangan dan penyulitan.

Semasa fasa pengujian, didapati sistem mampu mengesan pengubahsuaian laman web dan memberi amaran kepada pentadbir pelayan web sama ada melalui emel, bunyi dan perkhidmatan mesej ringkas (SMS). Untuk membolehkan penghantaran emel, kelas *Javamail 1.2* diperlukan. Manakala, amaran bunyi memerlukan kelas audio di dalam Java iaitu *javax.sound.**. Selain itu, proses pemulihan (*discovery*) laman web dapat dilaksanakan.

Pembangunan prototaip sistem ini telah dilakukan menggunakan bahasa pengaturcaraan Java menerusi Jbuilder 4.0. Memandangkan data yang disimpan tidak besar, maka pangkalan data Microsoft Access 2000 menjadi pilihan. Keselamatan pangkalan data juga diutamakan dengan penggunaan katalaluan. Walaubagaimana pun, katalaluan ini tidak begitu selamat apabila pengondam dapat meneka/memperolehi katalaluan tersebut. Di sini, cirri-ciri keselamatan perlu dipertingkatkan memandangkan pangkalan data ini menyimpan senarai direktori-direktori laman web yang diawasi.

Sistem ini juga mempunyai satu ciri keselamatan yang lain, dimana katalaluan diperlukan sebelum pentadbir laman web boleh membuka dan menggunakan sistem ini. Katalaluan ini selamat kerana ia telah dijana dan disimpan pada pangkalan data dalam bentuk nilai cincang iaitu perenambelasan. Dengan cara ini, pengondam sukar untuk mendapatkan katalaluan sebenar bagi memasuki dan menggunakan sistem. Ini kerana fungsi cincang merupakan satu fungsi satu hala. Selain itu, pengondam juga tidak dapat mengubah sebarang proses di dalam sistem seperti menamatkan proses pengesahan kerana ia dikawal menggunakan katalaluan juga.

Akhir sekali, satu perbandingan antara WebDID dan perisian lain telah dilakukan dan perbezaannya digambarkan di dalam Jadual 6.1. Diharap sistem WebDID yang dibangunkan ini dapat memperlihatkan sejauh mana implementasi algoritma cincang iaitu RIPEMD-160 dalam mengesan pencerobohan ke atas laman web dan menghalang pencerobohan ke atas sistem.

**Jadual 6.1: Perbandingan WebDID dengan perisian-perisian lain**

| | WebDID | Tripwire for Web Pages | WebAgain | WebAlarm |
|---|---|---|---|---|
| Fungsi | Memantau fail web secara masa nyata dan sistem pemulihan. | Hanya memantau fail web apabila pengimbas membuat permintaan & sistem pemulihan. | Memantau fail web secara masa nyata dan sistem pemulihan | Memantau fail web secara masa nyata dan sistem pemulihan. |
| Pengesanan | Mengesan pengubahsuaian kandungan web dengan membuat perbandingan nilai cincang. | Mengesan pengubahsuaian kandungan web dengan membuat perbandingan nilai cincang. | Membuat perbandingan berdasarkan saiz fail dan tanda masa/tarikh. | Mengesan pengubahsuaian dan penghapusan kandungan web dengan membuat perbandingan nilai cincang |
| Amaran dan pemberitahuan | E-mel, bunyi dan mesej ringkas ke telefon bimbit. | Hanya e-mel untuk amaran. | Hanya e-mel untuk pemberitahuan pembaikan. | E-mel, bunyi, mesej ringkas dan panggilan telefon. |
| Sandar dan pemulihan | Pemulihan fail web secara automatik. ith backup of the original files | Automatically recover tampered web content with temporary page. | Republishes the appropriate pages. | Automatically recover tampered web content with backup of the original files |
| Teknologi | Fungsi cincang 160-bit (RIPEMD-160) | Fungsi cincang 128-bit (MD5) | Binari | Fungsi cincang 160-bit (SHA1) |
| Bahasa pengaturcaraan | Java | C | Tidak diketahui. | Tidak diketahui |
| Pankalan data | MS Access 2000 | Tidak diketahui | Tidak diketahui | Tidak diketahui |
| Persekitaran | Pelanggan/Pelayan pada mesin yang sama. | Pelanggan/Pelayan pada mesin yang berlainan. | Pelanggan/Pelayan pada mesin yang sama atau berlainan. | Agent dalam sistem teragih. |

Di dalam penyelidikan mengenai keselamatan data yang sah pada pelanggan mahupun pelayan pula, ia menjurus kepada penyelidikan tentang pengesanan pencerobohan di dalam persekitaran E-Dagang. Hampir semua kaedah pengesanan pencerobohan anomali memerlukan kos yang sangat besar untuk mesin hos, ianya memerlukan suatu kapasiti yang besar untuk merekod semua aktiviti pengguna dan membuat profilnya sebagai ukuran untuk mengenalpasti dan mengesahkan sesuatu pencerobohan (Denning, 1987). Bahagian penyelidikan ini telah berjaya menganalisis *system call* dengan kaedah analisis pembezaan yang merupakan suatu kaedah yang sangat efektif dalam melakukan pengesanan pencerobohan. Sistem yang dibangunkan berupaya mengesan aktiviti pencerobohan tanpa menggunakan semua system call yang sedia ada pada sistem operasi Linux, tetapi ia menggunakan 10 *system call* sahaja untuk mengesan pencerobohan tersebut. Dan penggunaan kaedah dapat mengurangkan kos operasi dan kos pengubahsuaian dan juga kos analisis.

Pada penyelidikan yang terakhir iaitu penyelidikan pada keselamatan penghantaran data pada lapisan rangkaian, ia mengupas secara terperinci tentang kepentingan penggunaan IPSec yang selamat khususnya di dalam persekitaran E-Dagang.. Penggunaan IPSec untuk membendung masalah pencerobohan di dalam E-Dagang dapat dilihat sebagai penyelesaian kepada ancaman-ancaman ini. Namun, isu penggunaan algoritma penyulitan yang lemah dalam IPSec telah menyebabkan sistem keselamatan dalam IPSec menjadi kurang berkesan.

Untuk mengatasi masalah ini, satu algoritma yang mempunyai kekunci yang kukuh perlu digunakan bagi menggantikan penggunaan algoritma yang sedia ada. Dari kajian yang dilakukan, di dapati AES telah memenuhi kriteria ini di samping pelbagai kelebihan lain yang menambahkan lagi keberkesanan penggunaannya.

Untuk membolehkan AES berfungsi dengan sempurna dalam persekitaran IPSec, protokol pengurusan kekunci keselamatan yang sedia ada perlu diubahsuai. Fungsi protokol ini adalah untuk menakrifkan prosedur dan format paket data untuk berunding, menghasilkan, mengubahsuai dan menghapuskan SA (Set Keselamatan) yang lama jika terdapat SA yang baru dibina.

Kajian ini meliputi proses penentuan nilai ID transform bagi AES serta menentukan format paket SA yang baru bagi membolehkan AES berfungsi dalam IPSec. Setelah rangka kerja protokol terbina, ia diimplemenkan dalam persekitaran IPSec untuk menguji perlaksanaannya.

Dari ujian yang dilakukan, protokol yang dibina telah berfungsi dengan sempurna. Kajian ini diteruskan dengan analisis perbandingan penggunaan AES dalam persekitaran IPSec dengan beberapa algoritma penyulitan yang pernah digunakan dalam IPSec iaitu 3DES, DES, Blowfish dan Cast128. Didapati AES adalah paling fleksibel, paling tinggi nilai truputnya dan paling laju dari segi tempoh penghantaran datanya.

Diharapkan agar maklumat-maklumat yang diperolehi hasil dari kajian ini akan menjadi pemangkin kepada kajian-kajian seterusnya dan menjadi sumber rujukan kepada komuniti-komuniti penyelidikan yang lain yang sekarang ini semakin berminat untuk mengkaji potensi-potensi yang wujud dalam IPSec.

# RUJUKAN

Adams, C. (1997) " The CAST-128 Encryption Algorithm." RFC 2144, Internet Engineering Task Force.

Allen, J.,Christie, A., Fithen, W., McHugh, J., Pickel, J. and Stoner, E. (2000). "State of the Practice of Intrusion Detection Technologies." Technical Report. CMU/SEI-99-TR-028.

Andersen, D.G (2001). "Resilient Overlay Networks." Massachusetts Institute of Technology; Tesis Sarjana.

Anderson, T. W. (1994). "An Introduction to Multivariate Statistical Analysis." The second ed., Jhon Wiley & Sons, New York, 1994.

Anup, K.G (1998). "E-Commerce Security Weak Links, Best Defenses.", Canada: Wiley Computer Publishing.

Atkinson, R. (1995a). "IP Authentication Header.", RFC 1826, Internet Engineering Task Force.

Atkinson, R. (1995b). "IP Encapsulating Security Payload.", RFC 1827, Internet Engineering Task Force.

Atkinson, R. (1995c). "Security Architecture for the Internet Protocol.", RFC 1825, Internet Engineering Task Force.

Bellovin, S.M. (1997). "Probable Plaintext Cryptanalysis of the Security Protocols.", Proceeding of the 1997 Symposium on Network and Distributed System Security.

Biryukov, A., Kushilevitz, E. (1998). “Improved Cryptanalysis of RC5.”, Advance in Cryptology – EUROCRYPT’98 Proceedings, Spinger-Verlag, m.s. 85-99.

Blaha, Michael and Premerlani, William (1998). "Object-Oriented Modeling and Design for Database Application." Prentice -Hall, Inc. United State of America.

Blakely, J., Holley, J. and Sooter, M. (2001). “Critical Consideration for LAN-to-LAN Virtual Private Networks.”, Capstone Proceedings.

Blaze, M., Diffie, W., Rivest, R., Schneier, B., Shimomura, T., Thompson, E. and Weiner, M. (1996 ). “Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security.”,

Blaze, M., Ionnidis, J. and Keromytis, A..D. (2001). “ Trust Management and Network Layer Security Protocol.”, NDSS 2001, San Diego.

Bodacion Technologies, (2002), White Paper “Preventing Web Site Hacking”, Available at: www.bodacion.com/Downloads/PreventingWWWHacking.pdf

Braun, T., Gunter, M., Khalil, I. and Liu, L. (2000), “Performance Evaluation for Virtual Private Network”, Institute for Informatics and Angewandte Mathematics (AIM), University Bern.

Brown, S. (1999). “Implementing Virtual Private Networks.”, Mc-Graw-Hill, New York.

Brustoloni, J.C. and Garay, J.A. (1996 ).“ Application-Independent End-to-End Security in Shared-Link Access Networks.”, UNISEX Computing Systems, Vol 9.

Burroes, James H., (1995) "Secure Hash Standard", Federal Information Processing Standards Publication 180-1, National Institute of Standards ang Technology, 17 April. (download daripada http://www.itl.nist.gov/fipsspubs/fip180-1.htm.)

Carver , C.A., Hill, J.M., Surdu, J.R. and Pooch U.W. (2000). "A Methodology for Using Intelligent Agents to Provide Automated Intrusion Response." Proceedings of the 2000 IEEE Workshop on Information Assurance and Security. West Point, USA:IEEE.

Cheung, K.H. and Misic, J. (2002) "On Virtual Private Network Security Design Issues." *Computer Network Journal*. **Vol 38**. Issue 2.

Chung, R., (1984). "A Methodology for Protocol Design and Specification Based on an Extended State Transition Model." *Proceedings of the ACM SIGCOMM Symposium on Communications Architectures and Protocol.* Montréal, Canada.

Crosbie, *et.al* (1996). "IDIOT User Guide". Technical Report CSD-TR-96-050 (COAST TR 96-04), Department of Computer Science Purdue University.

CSI/FBI (2002) Computer Crime and Security Survey. Computer Security Issues and Treads. Vol. Viii, No.1, Spring 2002, pp. 1-24.

Daemen, J. and Rijmen, V. (1999). " AES Proposal: Rinjdael.", National Institute of Standard and Technology, September,.

Davis, C.R. (2001). "IPSec Securing VPNs.", McGraw-Hill, California.

Deering, S. and Hinden, R. (1998). "Internet Protocol, Version 6 (IPV6) Specification." RFC 2460, Internet Engineering Task Force.

Denning, D. E. (1987). "Haystack: An Intrusion-Detection model." IEEE Trans. Software Eng., vol. SE-13 No.2, 1987, pp. 222-232.

Dobbertin, H., Bosselaers, A., Preneel, B., (1996) "RIPEMD-160: A Strengthened Version of RIPEMD". Fast Software Encryption, LNCS 1039, Springer-Verlag, pp. 71-82.

Dobbertin, H., Bosselaers, A., Preneel, B., (1997) "The Cryptographic Hash Function RIPEMD-160". RSA Laboratories CryptoBytes, The technical newsletter of RSA Laboratories, a division of RSA Data Security, Inc., Volume: 3, Number: 2, Autumn, pp. 9-14.

Eberle, H. (1992). "A High-speed DES Implementation for Network Applications.", CRYPTP '92, 12th Annual International Cryptology Conference, Santa Barbara.

Endler, D. (1998). "Intrusion Detection Applying Machine Learning to Solaris Audit Data." Proceedings of the 14th Annual Computer Security Application Conference (ACSAC'98), Phoenix, Arizona.

E-Lock Corporation (2001), WebAlarm. Available at: http://www.elock.com.my/pdetail_webalarm.htm

Ferguson, N., and Schneier, B. (2001). "A Cryptographic Evaluation of IPSec.", Counterpane Internet Security, Inc., Minneapolis, USA.

Ferguson, P. and Huston, G. (1998). "What Is a VPN?- Part 2." , *The Internet Protocol Journal.*

Forrest, S., Hafmeyr, S. A., Somayaji, A. and LongStaff, T. A. (1996). "A Sense of Self for UNIX Process." Proceeding of IEEE.

Frankel, S., Kelly, S. and Glenn, R., (2002). "The AES Cipher Algorithm and Its Use With IPSec.", Internet Draft, IPSec Working Groups, NIST.

Gasparro, D. (1999) "Next-Gen VPNs: The Design Challenge." Data Communications, Vol.28, p.83.

Hadley, T. M. and Knapp, L.J. (2001). "Please Explain VPNs (Virtual Private Networks).", IBM.

Haller, N. and Atkinson, R. (1994). "On Internet Authentication." Internet Engineering Task Force, RFC 1704.

Hare, C., Karanjit Siyan (1996). "Internet Firewall and Network Security." New Readers Publishing, Indianapolis, USA.

Herscovitz, Eli. (1999). "Secure Virtual Private Networks; The Future Trends of Data Communications." RADGUARD Ltd.

Hofmeyr, S. A., Forrest, S. and Somayaji, A. (1998). "Intrusion Detection using Sequences of System Calls." Journal of Computer Security.

Hollander, Y., (2000) "Prevent Web Site Defacement". Internet Security Advisor, pp. 2-4.

Ilgun, K., Richard, A., Kemmerer and Porras, P. A. (1995). "State Transition Analysis: A Rule-Based Intrusion Detection Approach." IEEE Transactions on Software Engineering, 21(3): 181-199.

Jeffrey A.H, Joseph, S.V., and Joey, F.G., (1996) "Modern System Analysis and Design",The Benjamin/Cummings Publishing Company, Inc., United States of America and Canada.

Kawaguchi, M. (1973). “An Introduction to Multivariate Analysis.” Morikita, Tokyo.

Kazumaro Aoki and Lipmaa, H. (2000). “Fast Implementations of AES Candidates (2000).” The Third Advanced Encryption Standard Candidate Conference.

Kent, S. and Atkinson, R. (1998). “Security Architecture for Internet Protocol.” Internet Engineering Task Force, RFC 2401.

Kent, S. (1998). “IP Encapsulating Security Payload (ESP).” Internet Engineering Task Force, RFC 2406.

Kent, S., and Atkinson, R. (1998). "IP Authentication Header." Internet Engineering Task Force, RFC 2402.

Kumar, S. and Spafford E. (1994). “An Application of Pattern Matching in Intrusion Detection.” Technical Report 94-013, Purdue University Department of Computer Science.

Kosir, D., (1998). “Building and Managing Virtual Private Networks.” New York, NY, John Wiley and Sons, p.19

Lockstep System (2001), WebAgain, The Ultimate Web Site Protection Utility Automatically Fixes Hacked Web Sites. Available at: http://www.lockstep.com/products/webagain/wa-product.html

Maughan, D., Schertler, M., Schneider, M. and Turner, J. (1998). “Internet Security Association and Key Management Protocol (ISAKMP).” Internet Engineering Task Force, RFC 2408.

Midori, A., Masahiko, T., Takefumi, O., Shunji, O. and Shigeki, G. (1999). "Local Attack Detection and Intrusion Route Tracing." IEICE Trans. on Commun., Vol.E82-B No.11, pp.1826-1833.

Midori, A., Takefumi,. O., Tadeshi, I., Shunji, O. and Shigeki,G. (2001). "A new Intrusion Detection Method Based on Discrininant Analysis." IEICE Trans. on Info. & Syst., Vol.E84-B No.5, pp.570-577.

Midori, A., Takefumi, O., Tadeshi, I. and Shigeki, G. (2002) "Remote Attack Detection Method in IDA: MLSI-Based Intrusion Detection using Discriminant Analysis", IEEE SAINT2002, Nara, Japan.

Mohd Aizaini Maarof, Mazleena Salleh, Rabiah Ahmad dan Subariah Ibrahim (2000). Nota Kuliah Kriptografi.Universiti Teknologi Malaysia.

Nechvatal, J., Barker, E., Bassham, L., Burr, W., Dworkin, M., Foti, J. and Roback, E. (2000). "Report on the Development of the Advance Encryption Standard.(AES)." Computer Security Division, Information Technology Laboratory, National Institute of Standard and Technology.

NISER (2002) - National ICT Security & Emergency Response Centre, **"**60 Government Web Sites Hacked". Available at: http://www.niser.org.my/news/2001_04_06_02.html

Penny, A.K., (1996), "Introduction to System Analysis and Design; A Structured Approach", Times Mirror Higher Education Group, Inc. Company, United States of America.

Pereira, R. and Adams, R. (1998). "The ESP CBC-Mode Cipher Algorithm." Internet Engineering Task Force, RFC 2451.

Pereira, R. and Adams, R. (1998). “The ESP CBC-mode Cipher Algorithms.” Internet Engineering Task Force, RFC 2452.

Pfleeger, Charles.P. (1997). "Security in Computing." Prentice-Hall International. United State of America.

Postel, J. (1981). “Internet Protocol.” Internet Engineering Task Force, RFC 791.

Rivest, R.L. (1995). “The RC5 Encryption Algorithm.” In B. Preneel, editor, Fast Software Encryption, volume 1008 of Lecture Notes in Computer Science, pages 86--96, Springer Verlag.

Robshaw, M.J.B., (1996) “On Recent Result for MD2, MD4 and MD5. RSA Laboratories Bulletin”, New and advice from RSA Laboratories, Number: 4, November 12.

Rubin, A.D., Geer, D.E., Jr**.,** (1998) “A survey of Web security”**.** Computer, Volume: 31 Issue: 9, Sept. Page(s): 34 –41

Sandra, DD., (1996), “System Analysis and Design and the Transition to Objects”, McGraw-Hill Companies, Inc., United States of America.

Schneier, B. (1994). “Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish).” Fast Software Encryption, Cambridge Security Workshop Proceeding, Spinger-Verlag, m.s..191-204.

Schneir, B., Mudge (1998). “Cryptanalysis of Microsoft’s Point-to-Point Tunneling Protocol (PPTP).” Counterpane Publication.

Scott, C., Wolfe, P., and Erwin, M., (1998) "Virtual Private Networks." O'reilly Association. Suite A Sebstapol CA.

Simpson, W. (1995). "IP in IP Tunneling." Internet Engineering Task Force, RFC 1853.

Smith R.E (1997)."Internet Cryptography." Addison-Wesley.

Smith, R.E. (1997). "Internet Cryptography." Addison Wesley Longman, Inc. Massachusetts.

Soh, B.C., Young, S., (1998) "Network System and World Wide Web security". Computer Communication *20*, pp. 1431-1436

Somayaji, A. and Forrest, S. (2000). "Automated Response using System Call Delays." In Proceeding of the 9th USENIX Security Symposium, Denver.

Sturat, M., Saumil Shah and Sheeraj Shah, (2003), "Web Hacking Attacks and Defence, Pearson Education, Inc., United Ststes of America, Canada.

Tanebaum, A. S. (1987). "Operating System : Design and Implementation." New Jersey: Prentice-Hall Software Series.

Takefumi, O., Tadeshi, I. And Midori, A. (2001). "A Protection Mechanism for an Intrusion Detection System Base on Mandatory Access Control." In the 13th FIRST Conference 2001, Toulouse, France.

Tripwire Inc (2002)., Tripwire for Web Pages, Apache Edition. Available at: http://www.tripwire.com/products/web_pages/

Vaccaro, H. S. and Liepins, G. E. (1989). “Detection of Anomalous Computer Session Activity.” Proceedings of the 1989 IEEE Symposium on Research in Security and Privacy, pp. 280-289.

Whiting, D., Schneir, B. and Bellovin, S. (2000). “ AES Key Agility Issues in High-Speed IPSec Implementations.” Counterpane Internet Security, Inc., Minneapolis, USA.

Wright, M.A (2001). “The Advanced Encryption Standard.” *Journal Network Security*, m.s. 11-13.

# UNIVERSITI TEKNOLOGI MALAYSIA

## BORANG PENGESAHAN
## LAPORAN AKHIR PENYELIDIKAN

TAJUK PROJEK : RANGKA KERJA KESELAMATAN TRANSAKSI BAGI PELANGGAN-PELAYAN BERASASKAN PERDAGANGAN ELEKTRONIK (A SECURE TRANSACTION FRAMEWORK FOR CLIENT-SERVER BASED E-COMMERCE

Saya PROF. DR. ABDUL HANAN ABDULLAH
**(HURUF BESAR)**

Mengaku membenarkan **Laporan Akhir Penyelidikan** ini disimpan di Perpustakaan Universiti Teknologi Malaysia dengan syarat-syarat kegunaan seperti berikut :

1. Laporan Akhir Penyelidikan ini adalah hakmilik Universiti Teknologi Malaysia.
2. Perpustakaan Universiti Teknologi Malaysia dibenarkan membuat salinan untuk tujuan rujukan sahaja.
3. Perpustakaan dibenarkan membuat penjualan salinan Laporan Akhir Penyelidikan ini bagi kategori TIDAK TERHAD.
4. * Sila tandakan ( / )

| | | |
|---|---|---|
| ☐ | SULIT | (Mengandungi maklumat yang berdarjah keselamatan atau Kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972). |
| ☐ | TERHAD | (Mengandungi maklumat TERHAD yang telah ditentukan oleh Organisasi/badan di mana penyelidikan dijalankan). |
| ☑ | TIDAK TERHAD | |

_____
TANDATANGAN KETUA PENYELIDIK

_____
Nama & Cop Ketua Penyelidik

Tarikh : _____

**CATATAN :** ***** *Jika Laporan Akhir Penyelidikan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan ini perlu dikelaskan sebagai SULIT dan TERHAD.*