TUGAS AKHIR

STUDI ANALISIS PENGGUNAAN GNU PRIVACY GUARD (GPG) SEBAGAI ENKRIPSI KEAMANAN EMAIL BERBASIS WINDOWS

Oleh :

Nama : Deffri Riyadi

NIM : 2005-81-143

PROGRAM STUDI S-1 TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS INDONUSA ESA UNGGUL

JAKARTA

2010

STUDI ANALISIS PENGGUNAAN GNU PRIVACY GUARD (GPG) SEBAGAI ENKRIPSI KEAMANAN EMAIL BERBASIS WINDOWS

Tugas Akhir

Diajukan untuk memenuhi persyaratan Gelar Sarjana Komputer
pada Jurusan Teknik Informatika, Fakultas Ilmu Komputer
Universitas INDONUSA Esa Unggul

Oleh :

Nama : Deffri Riyadi

NIM : 2005-81-143

PROGRAM STUDI S-1 TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS INDONUSA ESA UNGGUL

JAKARTA

2010

PENGESAHAN TUGAS AKHIR

Nama	: Deffri Riyadi
NIM	: 2005-81-143
Jurusan	: Teknik Informatika
Fakultas	: Ilmu Komputer
Judul	: Studi Analisis Penggunaan Gnu Privacy Guard (GPG) Sebagai Enkripsi Keamanan Email Berbasis Windows

Tugas Akhir ini telah disetujui dan diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer, Program Studi Strata-1 Fakultas Ilmu Komputer.

Jakarta, Februari 2010

(Zulkarnaen Sanany, M.Sc)	(Ir. I. Joko Dewanto, MM)
Pembimbing Materi	Pembimbing Tulisan

Mengetahui,

(Ir. I. Joko Dewanto, MM)	(Ir. Munawar, MMSI, M.Com)
Kajur Teknik Informatika	Dekan Fakultas Ilmu Komputer

LEMBAR PENGESAHAN PENGUJI SIDANG

NAMA	:	DEFFRI RIYADI
NIM	:	2005-81-143
JURUSAN	:	TEKNIK INFORMATIKA
FAKULTAS	:	ILMU KOMPUTER
JUDUL	:	Studi Analisis Penggunaan GNU Privacy Guard (GPG) Sebagai Enkripsi Keamanan Email Berbasis Windows

Tugas Akhir ini telah disetujui sebagai syarat untuk memperoleh gelar Sarjana Komputer, Program Studi Strata 1 Ilmu Komputer Jurusan Teknik Informatika Universitas Indonusa Esa Unggul.

Jakarta, Maret 2010

Disetujui oleh,

Penguji I	:	Ir. I. Joko Dewanto, MM	:	_____
Penguji II	:	Riya Widayanti, S.Kom, MMSI	:	_____
Penguji III	:	Fransiskus Adikara, S.Kom, MMSI	:	_____

Mengetahui,

Ir.Joko Dewanto,MM

Koordinator Tugas Akhir

UNIVERSITAS INDONUSA ESA UNGGUL

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI TEKNIK INFORMATIKA

PERNYATAAN

Seluruh isi/materi tugas akhir ini menjadi tanggung jawab penulis sepenuhnya.

Jakarta, Maret 2010

Penulis

(Deffri Riyadi)

ABSTRAK

GNU Privacy Guard (GnuPG atau GPG) merupakan suatu perangkat lunak open source yang mengimplementasikan standar OpenPGP secara lengkap sebagaimana yang didefinisikan pada RFC4880. Dengan menggunakan GnuPG, pengguna bisa mengenkripsi dan menandatangani data atau pesan dalam komunikasinya. GnuPG menggunakan kombinasi kriptografi kunci-simetrik konvensional dan kriptografi kunci-publik. GnuPG mengenkripsi email atau data menggunakan kunci public dan untuk mendekripsinya menggunakan kunci pribadi. Layanan keamanan GnuPG meliputi kerahasiaan, manajemen kunci, otentifikasi, dan tanda tangan digital.

GnuPG ini sudah banyak digunakan di berbagai sistem operasi, terutama yang open source seperti Linux, FreeBSD, dan segala varian keduanya, ada beberapa aspek keamanan GnuPG yang patut dipertimbangkan. Salah satu yang perlu diperhatikan adalah keamanan algoritma kriptografi yang digunakan di dalamnya.

Meskipun GnuPG kebanyakan digunakan di lingkungan sistem operasi yang open souce seperti Linux dan FreeBSD, kode program GnuPG juga bisa digunakan di lingkungan Windows. GPG adalah suatu metode enkripsi informasi yang bersifat rahasia sehingga jangan sampai diketahui oleh orang lain yang tidak berhak. Informasi ini biasa berupa E-mail yang sifatnya rahasia, nomor kode kartu kredit, atau pengiriman dokumen rahasia perusahaan melalui Internet. GPG menggunakan metode kriptografi yang disebut "*public key encryption*": yaitu suatu metode kriptografi yang sangat *sophisticated*.

Kata Kunci : Email, Kriptografi, GNU Privacy Guard, Enkripsi , Dekripsi, Kunci Publik.

KATA PENGANTAR

Puji dan syukur saya ucapkan ke hadirat Allah SWT yang telah memeberikan berkah, rahmat dan karunianya sehingga saya dapat menyelesaikan Tugas Akhir ini yang berjudul “Studi Analisis Penggunaan Gnu Privacy Guard (GPG) Sebagai Enkripsi Keamanan Email Berbasis Windows.

Tujuan utama dari penyusunan penulisan Tugas Akhir ini adalah untuk melengkapi persyaratan akademik bagi mahasiswa Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Indonusa Esa Unggul untuk menyelesaikan program pendidikan Sarjana Strata 1 (S-1).

Saya ucapkan terima kasih kepada pihak-pihak yang telah membantu dalam proses penyusunan tugas akhir ini baik moril maupun materil sehingga saya dapat menyelesaikan penyusunan Tugas Akhir ini.

Pada kesempatan ini saya mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Kedua orang tuaku, yang telah memberikan kasih sayang, perhatian, cinta dukungan dan doa, yang membuat saya tetap semangat dan berusaha dalam menyelesaikan Tugas Akhir ini.
2. Bapak Zulkarnaen Sanany, MSc, selaku pembimbing materi yang telah memeberikan kontribusi besar terhadap materi-materi yang berhubungan dengan Tugas akhir ini.
3. Bapak Ir. I. Joko Dewanto, MM selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Indonusa Esa Unggul serta selaku pembimbing tulisan yang telah membantu dalam penulisan penyusunan Tugas akhir ini.
4. Bapak Ir. Munawar MMSI, M.Com selaku Dekan Fakultas Ilmu Komputer Universitas Indonusa Esa Unggul.

5. Seluruh Staff dan Dosen Fakultas Ilmu Komputer yang telah membimbing dan memberikan Ilmu selama perkuliahan.
6. Rekan-rekan di Fakultas Ilmu Komputer, khususnya rekan-rekan Teknik Informatika angkatan 2005.
7. Sahabat-sahabat terbaikku, Yolanda, Prisiella, Eva, Maharani, Anissah dan Ary Setyawati.
8. Para penulis buku, artikel dan blog yang telah membagi ilmunya sebagai bahan referensi untuk tugas akhir ini.
9. Semua pihak yang turut membantu dalam menyelesaikan tugas akhir ini.

Dengan segala keterbatasan dalam penyusunan Tugas Akhir ini, saya mengharapkan partisipasi semua pihak untuk dapat memberikan masukan guna menyempurnakan Tugas akhir ini. Semoga Tugas akhir ini bermanfaat untuk para pembaca.

Jakarta, Maret 2010

Deffri Riyadi

DAFTAR ISI

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi telekomunikasi dan penyimpanan data dengan menggunakan komputer memungkinkan pengiriman data jarak jauh yang relative cepat dan murah. Di lain pihak pengiriman data jarak jauh melalui gelombang radio ,maupun media lain yang digunakan masyarakat luas (*publik*) sangat memungkinkan pihak lain dapat menyadap dan mengubah data yang dikirimkan. Demikian juga pada sistem jaringan komputer maupun secara luas pada internet dengan jumlah pemakai yang banyak.

Dalam teknologi informasi, telah dan sedang dikembangkan cara-cara untuk menangkal berbagai bentuk serangan semacam itu. Salah satu cara yang ditempuh mengatasi masalah ini ialah dengan menggunakan kriptografi yang menggunakan transformasi data sehingga data yang dihasilkan tidak dapat dimengerti oleh pihak ketiga.

Transformasi ini memberikan solusi pada dua masalah keamanan data, yaitu masalah privasi (*privacy*) dan keautentikan (*authentication*). Privasi mengandung arti bahwa data yang dikirimkan hanya dapat dimengerti informasinya oleh penerima yang sah. Sedangkan keautentikan mencegah pihak ketiga untuk mengirimkan data yang salah atau mengubah data yang dikirimkan.

E-mail sudah digunakan orang sejak awal terbentuknya internet pada sekitar tahun 1969 dan merupakan salah satu fasilitas yang ada pada saat itu. Sesuai dengan perkembangan internet, penggunaan email ini juga semakin membesar walaupun pada saat ini persentasinya sudah turun karena adanya sebuah fasilitas baru di internet yang dikenal sebagai Web. Salah satu alasan kenapa email dipakai orang karena memberikan cara yang mudah dan cepat dalam mengirimkan sebuah informasi. Selain itu dengan

email dapat juga informasi yang ukurannya kecil sampai ke file yang ukurannya besar.

Namun sifat e-mail yang memanfaatkan penghantar elektronik tak sepenuhnya dimaksudkan sebagai medium pribadi karena menyimpan potensi bahaya penyalahgunaan yang bukan saja menjengkelkan tetapi juga dapat bersifat fatal.

Ketika kita mengirimkan suatu e-mail, maka e-mail tersebut disampaikan ke suatu sistem komputer yang mungkin kita tidak mengetahui administratornya. Dari komputer tersebut disampaikan ke sistem komputer lain, dan yang lainnya, dan lainnya, sampai kepada penerima yang dituju. Pada beberapa link di rantai ini, e-mail kita dapat dibaca oleh siapa saja yang diinginkan *system administrator*, atau oleh suatu biro penyelidikan yang sedang mencurigai suatu aktivitas kejahatan, atau berbagai kemungkinan lainnya. Tetapi secara ringkasnya adalah ketika kita mengirimkan suatu e-mail, kita tidak mengetahui siapa yang membaca pesan itu, penerima yang diharapkan ataupun barangkali orang lain.

Kerahasiaan email terancam bukan oleh para hacker, melainkan para *system administrator* sendiri. Para *system administrator* terkadang bosan tidak tahu apa yang harus dikerjakan selain membaca-baca email orang. Mereka dapat melakukannya tanpa sedikit pun meninggalkan jejak.

Dalam hal ini penulis mencoba menganalisa penggunaan GnuPG atau GPG untuk keamanan email pada sistem operasi windows.

1.2 Tujuan dan Manfaat

Tujuan dari penulisan Tugas Akhir ini adalah :

1. Sebagai prasyarat kelulusan studi S1 Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Indonusa Esa Unggul.

2. Memberikan informasi kepada para pembaca mengenai teknologi enkripsi E-mail dalam usaha untuk menghindari pembacaan email oleh yang tidak berhak.
3. Memberikan wacana tentang sekuriti e-mail dengan GNU Privacy Guard (GnuPG, atau GPG), khususnya untuk pengguna sistem operasi windows.

Manfaat dari penulisan Tugas Akhir ini adalah :

1. Memaksimalkan penggunaan enkripsi untuk keamanan pengiriman email.
2. Memenuhi kebutuhan akan keamanan informasi dan data yang semakin tinggi.

1.3 Perumusan Masalah

Keamanan email terancam bukan hanya dari gangguan para hacker, melainkan juga para *system administrator*-nya sendiri yang iseng dengan membaca email orang lain maupun pihak ketiga yang tidak berhak. Mereka dapat melakukannya tanpa sedikit pun meninggalkan jejak. Cara mengatasi hal ini maka untuk mendapatkan jaminan kerahasiaan di dalam pengiriman email maka perlu menggunakan program aplikasi yang menjanjikan keamanan yang lebih baik. Salah satunya menggunakan Gnu Privacy Guard.

1.4 Pembatasan Masalah

Untuk memberikan penekanan khusus sesuai dengan judul tugas ini maka dilakukan pembatasan pada penulisan Tugas Akhir ini:

1. Membahas proses email pada internet, masalah-masalah yang timbul pada email beserta sistem keamanan yang dipakai pada email.

2. Sistem sekuriti email dengan menggunakan Gnu Privacy Guard, cara instalasi, dan penggunaan GPG pada sistem operasi windows.
3. Membahas metode algoritma yang dipakai pada GNU Privacy Guard berbasis Windows.
4. Membuat pasangan kunci. Kunci private dan kunci public. Melakukan pertukaran kunci public (ekspor dan impor).
5. Tidak membahas algoritma tanda tangan digital.
6. Membahas kelebihan-kelebihan dan kelemahan-kelemahan GNU Privacy Guard.

1.5 Metode Penelitian

Metode yang digunakan oleh penulis dalan menyusun tugas akhir ini adalah:

1. Metode Literatur

   Penulis melakukan studi literature dalam mencari dan memperoleh data-data dari berbagai buku, artikel, dan website yang berhubungan dengan tugas akhir ini.
2. Metode Analisis

   Penulis melakukan analisa tentang GNU Privacy Guard pada sistem operasi berbasis windows.

1.6 Sistematika Penulisan

Penulisan Tugas Akhir dengan judul "**Studi Analisis Penggunaan Gnu Privacy Guard (GPG) Sebagai Enkripsi Keamanan Email Berbasis Windows"** terbagi dalam 5 bab yang tersusun sebagai berikut:

BAB I PENDAHULUAN

Bab ini menguraikan tentang latar belakang masalah, tujuan dan manfaat, perumusan masalah, pembatasan masalah, metode penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menguraikan tentang teori-teori yang menjadi acuan dalam penyusunan tugas akhir ini.

BAB III GNU PRIVACY GUARD PADA EMAIL BERBASIS WINDOWS

Bab ini menguraikan tentang metode yang digunakan oleh penulis untuk menyusun tugas akhir ini. Serta menjelaskan seputar pemakaian Gnu Privacy Guard serta cara kerja dari Gnu Privacy Guard berbasis Windows.

BAB IV PENGUJIAN GNU PRIVACY GUARD

Bab ini menjelaskan pengujian GPG berbasis Windows yang telah dilakukan oleh penulis.

BAB V PENUTUP

Bab ini mengemukakan kesimpulan dari tugas akhir dan saran dari penulis untuk pengembangan lebih lanjut.

BAB II
LANDASAN TEORI

2.1 Kriptografi

Kriptografi berasal dari bahasa Yunani, *crypto* dan *graphia.* Crypto berarti *secret* (rahasia) dan graphia berarti *writing* (tulisan). Menurut terminologinya, kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain.

2.1.1 Komponen Kriptografi

Pada dasarnya komponen kriptografi terdiri dari beberapa komponen, seperti :

a. Enkripsi.

Merupakan hal yang sangat penting dalam kriptografi, merupakan cara pengamanan data yang dikirimkan sehingga terjaga kerahasiaannya. Pesan asli disebut *plaintext* (teks-biasa), yang diubah menjadi kode-kode yang tidak dimengerti. Enkripsi bisa diartikan dengan *cipher* atau kode. Sama halnya dengan tidak dimengerti sebuah kata maka kita akan melihatnya di dalam kamus atau daftar istilah. Beda halnya dengan enkripsi, untuk mengubah teks-biasa ke bentuk teks-kode kita gunakan algoritma yang dapat mengkodekan data yang kita inginkan.

b. Dekripsi.

Merupakan kebalikan dari enkripsi. Pesan yang telah dienkripsi dikembalikan ke bentuk asalnya. Algoritma yang digunakan untuk dekripsi tentu berbeda dengan yang digunakan untuk enkripsi.

c. Kunci.

Adalah kunci yang dipakai untuk melakukan enkripsi dan dekripsi. Kunci terbagi menjadi dua bagian, yaitu kunci rahasia (*private key*) dan kunci umum (*public key*)

d. Ciphertext.

Merupakan suatu pesan yang telah melalui proses enkripsi. Pesan yang ada pada teks-kode ini tidak bisa dibaca karena berupa karakter-karakter yang tidak mempunya makna (arti).

e. Plaintext.

Sering disebut dengan *cleartext*. Teks-asli atau teks-biasa ini merupakan pesan yang ditulis atau diketik yang memiliki makna. Teks-asli inilah yang diproses menggunakan algoritma kriptografi untuk menjadi *ciphertext* (teks-biasa).

f. Pesan.

Dapat berupa data atau informasi yang dikirim (melalui kurir, saluran komunikasi data, dsb) atau yang disimpan di dalam media perekaman (kertas, storage, dsb).

g. Cryptanalysist.

Kriptanalisis bisa diartikan sebagai analisis kode atau suatu ilmu untuk mendapatkan teks-asli tanpa harus mengetahui kunci yang sah secara wajar. Jika suatu teks-kode berhasil diubah menjadi teks-asli tanpa menggunakan kunci yang sah, proses tersebut dinamakan *breaking code.* Hal ini dilakukan oleh kriptanalis. Analisis kode juga dapat menemukan kelemahan dari suatu algoritma kriptografi dan akhirnya dapat menemukan kunci atau teks-asli dari teks-kode yang dienkripsi dengan algoritma tertentu.

2.1.2 Fungsi dan Tujuan Kriptografi

Berikut merupakan beberapa fungsi dari metode kriptografi :

- Kriptografi dapat digunakan untuk memeberikan jaminan keamanan pesan, data, atau informasi serta menjaga kerahasiaannya.
- Kriptografi dapat digunakan untuk mengenkripsi file-file yang bersifat sensitive dan rahasia.
- Kriptografi dapat digunakan untuk meningkatkan keamanan komunikasi pada lintas jaringan meskipun komunikasi trsebut dalam jaringan eksternal, yaitu internet.
- Kriptografi juga dapat mengembalikan file-file yang telah dienkripsi menjadi file-file yang sebenarnya dengan menggunakan digital signature. [*Wiharsono Kurniawan,2007*].

Adapun tujuan kriptografi adalah sebagai berikut :

- Untuk menjaga kerahasiaan suatu pesan, data, atau informasi dengan cara mengenkripsi.
- Untuk menjaga keutuhann pesan, data, informasi agar pada saat dibuat, dikirim, dan dibuka kembali tidak mengalami perubahan sedikitpun.
- Dapat digunakan untuk mengidentifikasi keaslian pesan, data, atau informasi.
- Dapat dipakai untuk membuktikannpemilik pesan, data, atau informasi. [*Wiharsono Kurniawan,2007*].

2.1.3 Algoritma Kriptografi

Definisi terminology algoritma adalah urutan langkah-langkah logis untuk menyelesaikan masalah yang disusun secara sistematis.

Algoritma kriptografi merupakan langkah-langkah logis bagaimana menyembunyikan pesan dari orang-orang yang tidak berhak atas pesan tersebut.

Keamanan dari algoritma modern didapat dengan merahasiakan kunci yang dimiliki dari orang lain, tanpa harus merahasikan algoritma itu sendiri. Kunci memilki fungsi yang sama dengan password. Jika keseluruhan dari keamanan algoritma tergantung pada kunci yang dipakai maka algoritma ini bisa dipublikasikan dan dianalisis oleh orang lain. Jika algoritma yang telah dipublikasikan bisa dipecahkan dalam waktu singkat oleh orang lain maka berarti algoritma tersebut tidaklah aman untuk digunakan.

2.1.4 Algoritma Kriptografi Klasik

Kriptografi klasik merupakan suatu algoritma yang menggunakan satu kunci untuk mengamankan data. Dua teknik dasar yang biasa digunakan pada algoritma jenis ini adalah sebagai berikut :

1. Teknik Substitusi : Penggantian setiap karakter teks-asli dengan karakter lain.
2. Teknik Transposisi (Permutasi) : Teknik ini menggunakan permutasi karakter, yang mana dengan menggunakan teknik ini pesan yang asli tidak dapat dibaca kecuali oleh orang yang memiliki kunci untuk mengembalikan pesan tersebut ke bentuk semula.

Kriptografi klasik memiliki ciri :

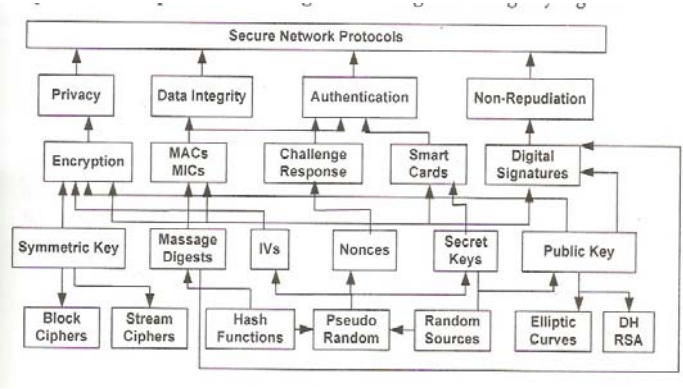
1. Berbasis karakter.
2. Menggunakan pena dan kertas saja, belum ada komputer.
3. Termasuk ke dalam kriptografi kunci simetri

Tiga alasan memepelajari algoritma klasik :

1. Memahami konsep dasar kriptografi.
2. Dasar algoritma modern.
3. Memahami kelemahan sistem kode.

2.1.5 Algoritma kriptografi Modern

Enkripsi modern berbeda dengan enkripsi konvensional. Enkripsi modern sudah menggunakan komputer untuk pengoperasiannya, berfungsi untuk mengamankan data baik yang ditransfer melalui jaringan komputer maupun bukan. Hal ini sangat berguna untuk melindungi *privacy, data integrity, authentication* dan *non-repudiation.*



Gambar 2.1 Skema Kriptografi Modern
(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :107)

Pada gambar, antara faktor yang satu dengan yang lain saling berhubungan untuk mendapatkan keamanan yang dikehendaki, seperti *Privacy* disukung oleh *Encryption, Data Integrity* didukung

oleh pemberian *MAC, Authentication* didukung oleh *MAC, Challenge Response,* dan *Digital Signature. Encryption* terdiri dari *Symetric Key, Public Key, dan IVs,* sedangkan *Symetric Key* terdiri dari blok Cipher dan Stream Ciphers, sementara *Public Key* terdiri dari ECC dan RSA. MAC dibentuk dari Hash Functon yang akan menghasilkan Message Digests.

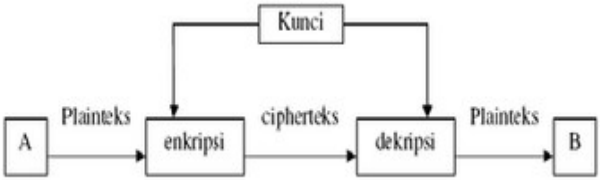
2.1.5.1 Macam-Macam Algoritma Kriptografi Modern

Pada kriptografi modern terdapat berbagai macam algoritma yang dimaksudkan untuk mengamankan informasi yang dikirim melalui jaringan komputer. Algoritma kriptografi modern terdiri dari tiga bagian :

1. Algoritma Simetris.

   Algoritma Simetris adalah algoritma yang menggunakan kunci yang sama untuk enkripsi dandekripsinya. Keamanan dari pesan yang menggunakan algoritma ini tergantung pada kunci. Jika kunci tersebut diketahui oleh orang lain maka orang tersebut akan dapat melakukan enkripsi dan dekripsi terhadap pesan. Algoritma yang memakai kunci simetri di antaranya adalah :

   1. Data Encryption Standard (DES).
   2. RC2, RC4, RC5, RC6.
   3. International Data Encryption Algorithm (IDEA).
   4. Advanced Encryption Standard (AES).
   5. One Time Pad (OTP).
   6. A5.



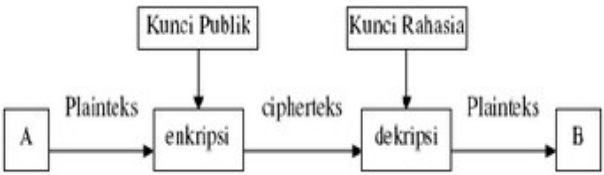
Gambar 2.2 Algoritma Simetris

(sumber : http://wahid.web.ugm.ac.id/sandi/simetris.jpg)

2. Algoritma Asimetri.

Algoritma asimetris adalah pasangan kunci kriptografi yang salah satunya digunakan untuk proses enkripsi dan yang satu lagi untuk dekripsi. Sering juga disebut dengan kunci publik, dengan arti kata kunci yang digunakan untuk melakukan enkripsi dan dekripsi berbeda. Pada algoritma asimetris kunci terbagi menjadi dua bagian, yaitu :

1. Kunci Umum (*Public Key*).
   Kunci yang boleh semua orang tahu (dipublikasikan).
2. Kunci Rahasia (*Private Key*).
   Kunci yang dirahasiakan (hanya boleh diketahui oleh satu orang).



Gambar 2.3 Algoritma Asimetris

(sumber : http://wahid.web.ugm.ac.id/sandi/asimetris.jpg)

Kunci-kunci tersebut berhubungan satu sama lain. Dengan kunci publik orang dapat mengenkripsi pesan tetapi tidak bisa mendekripsinya. Hanya orang yang memiliki kunci rahasia yang dapat mendekripsi pesan tersebut. Algoritma asimetris bisa mengirimkan pesan lebih aman daripada algoritma simetris.

Kriptografi kunci asimetri memiliki beberapa keuntungan yaitu :

- Hanya kunci rahasia yang dijaga kerahasiaannya.
- Tidak ada kebutuhan dalam mengirim kunci privat tidak seperti pada sistem kunci simetri.
- Pasangan kunci publik / privat tidak perlu diubah dalam periode waktu yang lama.
- Dapat digunakan dalam pengiriman kunci simetri.
- Beberapa algoritma kunci publik dapat digunakan dalam pemberian tanda tangan digital.

Adapun kelemahan yang dimiliki kriptografi kunci asimetri adalah :

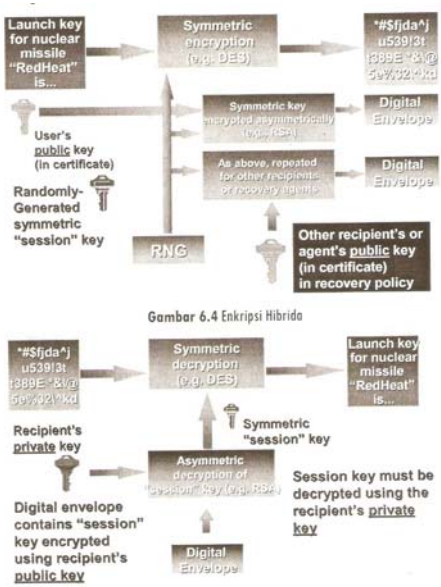
- Enkripsi dan dekripsi memakai waktu yang lama karena menggunakan bilangan yang sangat besar.
- Ukuran *ciphertext* lebih besar daripada *plaintext.*
- Ukuran kunci biasanya lebih besar daripada ukuran kunci simetri.
- *Ciphertext* tidak memberikan informasi mengenai otentikasi Pengirim.

Contoh algoritma yang memakai kunci publik di antaranya adalah :

1. Digital Signature Algorithm (DSA)
2. RSA.
3. Diffie-Hellman (DH).
4. Elliptic Curve Cryptography (ECC).
5. Kriptografi Kuantum

3. Algoritma Hibrida

Algoritma Hibrida adalah algoritma yang memanfaatkan dua tingkatan kunci, yaitu kunci rahasia (simetri) – yang disebut juga session key (kunci sesi) – untuk enkripsi data dan pasangan kunci rahasia-kunci publik untuk pemberian tanda tangan digital serta melindungi kunci simetri.



Gambar 6.4 Enkripsi Hibrida

Gambar 2.4 Algoritma Hibrida

(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :110)

2.1.6 RSA

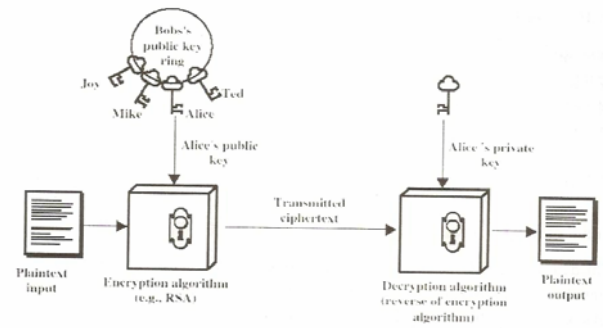
Algoritma ini melakukan pemfaktoran bilangan yang sangat besar. Oleh karena alasan tersebut RSA dianggap aman. Untuk membangkitkan dua kunci, dipilih dua bilangan prima acak yang besar.

Algoritma RSA dibuat oleh 3 orang peneliti dari MIT (*Massachussets Institute of Technology*) pada tahun 1976, yaitu : Ron (R)ivest, Adi (S)hamir, dan Leonard (A)dleman. RSA mengekspresikan teks-asli yang dienkripsi menjadi blok-blok yang mana setiap blok memiliki nilai bilangan biner yang diberi symbol

"n", blok teks-asli "M" dan blok teks-kode "C". Untuk melakukan enkripsi pesan "M", pesan dibagi ke dalam blok-blok numerik yang lebih kecil daripada "n" (data biner dengan pangkat terbesar). Jika bilangan prima yang panjangnya 200 digit, dapat ditambah beberapa bit 0 di kiri bilangan untuk menjaga agar pesan tetap kurang dari nilai "n".

Besaran yang digunakan pada algoritma RSA :

1. p dan q bilangan prima (rahasia)
2. $r = p \cdot q$ (tidak rahasia)
3. $\Phi(r) = (p-1)(q-1)$ (rahasia)
4. PK (kunci enkripsi) (tidak rahasia)
5. SK (kunci dekripsi) (rahasia)
6. X (teks asli) (rahasia)
7. Y (teks kode) (tidak rahasia)



Gambar 2.5 Proses Enkripsi dan Dekripsi RSA
(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :151)

Untuk melakukan enkripsi RSA, teks-asli disusun menjadi blok *x1*, *x2*, …, sedemikian sehingga setiap blok merepresentasikan

nilai di dalam rentang 0 sampai r-1. Setiap blok *xi* dienkripsi menjadi blok *yi* dengan rumus :

$C = M^e \bmod n$

Untuk melakukan dekripsi terhadap teks-kode yang menggunakan algoritma RSA, setiap blok teks-kode *yi* didekripsi kembali menjadi blok *xi* dengan rumus :

$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$

Letak keamanan pada RSA :

Keamanan algoritma RSA terletak pada tingkat kesulitan dalam memfaktorkan bilangan non-prima menjadi faktor primanya, yang dalam hal ini r = p x q. sekali r berhasil difaktorkan menjadi p dan q maka $\Phi(r) = (p-1)\ (q-1)$ dapat dihitung. Selanjutnya, karena kunci enkripsi *PK* diumumkan (tidak rahasia) maka kunci dekripsi *SK* dapat dihitung dari persamaan $(PK.SK) \equiv 1 \pmod{\Phi(r)}$.

Penemu algoritma RSA menyarankan agar panjang nilai p dan q lebih dari 100 digit. Dengan demikian hasil kali $r = p \times q$ akan lebih dari 200 digit. Menurut mereka, usaha untuk mencari faktor bilangan 200 digit membutuhkan waktu komputasi selama 4 milyar tahun (dengan asumsi bahwa algoritma pemfaktoran yang digunakan adalah algoritma yang tercepat saat ini dan komputer yang dipakai mempunyai kecepatan 1 milidetik). Algoritma RSA yang memfaktorkan bilangan yang besar belum ditemukan. Inilah yang membuat algoritma RSA tetap dipakai. Selagi belum ditemukan algoritma yang mampu untuk memfaktorkan bilangan bulat menjadi faktor primanya maka algoritma RSA tetap direkomendasikan untuk digunakan dalam penyandian pesan.

2.1.7 DSA (Digital Signature Algorithm)

Pada bulan Agustus tahiun 1991, National Institute of Standards and Technology (NIST), lembaga Standard dan Teknologi Nasional Amerika, mengusulkan DSA untuk menjadi standar tanda tangan digital, Digital Signature Standard (DSS).

Kritik terhadap DSA :

- DSA tidak dapat digunakan untuk enkripsi atau distribusi kunci.
  Benar. Namun ini adalah standar tanda tangan digital, dan bukan standar enkripsi. Bila ada trapdoor, maka yang dimaksud adalah kemungkinan seseorang memalsukan tanda tangan DSA sehingga membahayakan sistem. Sebenarnya yang mungkin adalah trapdoor dalam implementasinya. Namun ini dapat berlaku bagi algoritma apa saja. Misalnya dalam pembangkitan bilangan prima dibuat supaya mudah ditemukan nilainya.
- DSA dibuat oleh NSA yang dicurigai telah menanamkan pintu belakang terhadap algoritma.
- DSA lebih lambat daripada RSA. Laju pembangkitan tanda tangan sama, namun pemeriksaan tanda tangan dapat lebih lama 10 hingga 40 kali disbanding denagan RSA.
- RSA merupakan standar defakto. Banyak perusahaan telah membelanjakan jutaan dolar untuk mendukung RSA. Bila dipaksa berpindah ke DSA, tentu akan lebih banyakl agi dana yang dikeluarkan.
- Proses pemilihan DSA tidak transparan, tidak diberikan cukup waktu untuk menganalisis DSA.
- Ukuran kunci terlalu kecil, 512 bit. Untuk menjawab kritikan ini, NIST membuat panjang kunci variable dari 512 bit hingga 1024 bit.

2.1.8 Algoritma ElGamal

ElGamal adalah suatu *public key cryptosystem* yang dibuat pada tahun 1985. Algoritma ElGamal digunakan untuk melakukan enkripsi dan tanda tangan digital. Keamanan dari algoritma ElGamal terletak pada susahnya perhitungan logaritma yang terpisah pada GF (p) ketika p merupakan bilangan prima yang besar. Faktorisasi utama dari logaritma yang terpisah dianjurkan untuk diimplementasikan pada RSA dan ElGamal cryptosystem.

Teknik perhitungan dasar untuk enkripsi dan tanda tangan digital menggunakan algoritma ElGamal dengan dua kunci cryptosystem. Algoritma enkripsi ElGamal seperti berikut ini :

Kunci umum :

p (bilangan prima)

g, x < p (dua bilangan acak)

$y \equiv gx \pmod{p}$

y, g dan p : kunci umum

Kunci rahasia :

x < p

Enkripsi :

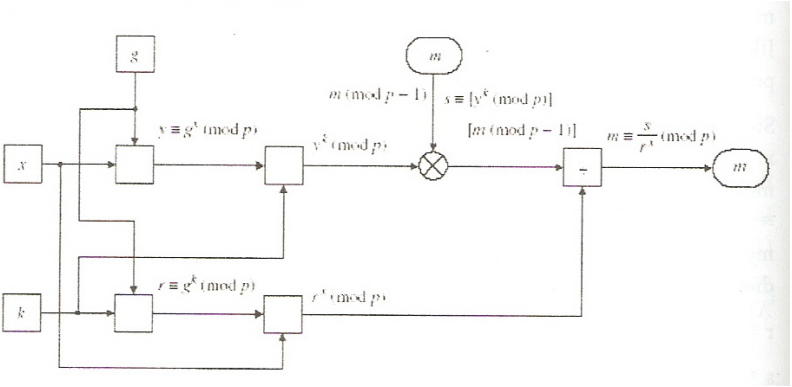
k : bilangan acak gcd (k, p-1)=1

$r \equiv gk \pmod{p}$

$s \equiv (yk \pmod{p})\ (m \pmod{p-1})$

Dekripsi

$m \equiv s/rx \pmod{p}$, 0_m_p-1



Gambar 2.6 Proses Enkripsi ElGamal

(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :164)

2.1.9 AES (Advanced Encryption Standard)

AES dipublikasikan oleh NIST (*National Institute of Standard and Technology*) pada tahun 2001. AES merupakan blok kode simetris untuk menggantikan DES (*Data Encryption Standard*).

Persyaratan AES adalah :

- Algoritma harus dipublikasikan secara luas untuk diperiksa keamanannya.
- Algoritma haruslah merupakan blok cipher.
- Algoritma harus dapat diimplementasikan dengan cepat dalam software dan hardware.
- Algoritma memiliki masukan blok data 128 bit.
- Algoritma harus memiliki kunci yang fleksibel : 128, 192, dan 256 bit.
- AES harus dapat digunakan sebagai fungsi hash.

- Dapat mengenkrip data masukan 32 bit atau 64 bit, supaya dapat mengenkrip aliran video dan audio secara real time.
- Dapat digunakan dalam smart card dengan CPU 8 bit.

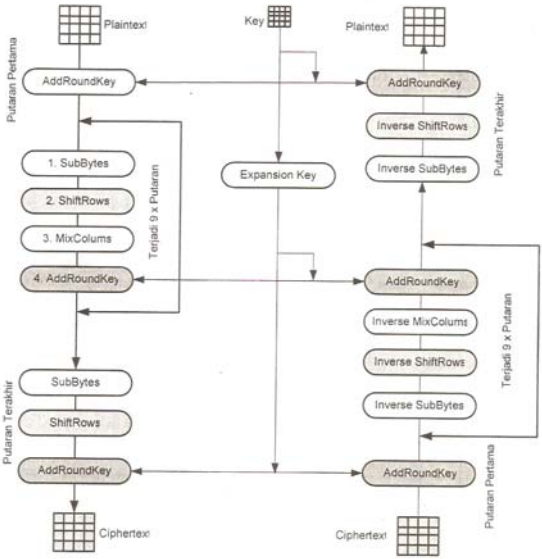
AES mempunyai kunci 128, 192 dan 256 bit sehingga berbeda dengan panjang dari putaran Rijndael.

	AES 128	AES 192	AES 256
Key Size	4 word (16 byte)	6 word (24 byte)	8 word (32 byte)
Pliantext Block Size	4 word (16 byte)	4 word (16 byte)	4 word (16 byte)
Number of Roud	10	12	14
Round Key Size	4 word (16 byte)	4 word (16 byte)	4 word (16 byte)
Expanded Key Size	44 word (176 byte)	52 word (208 byte)	60 word (240 byte)

Tabel 2.1 Parameter AES

(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :169)

Dari tabel tersebut AES-128 bit menggunakan panjang kunci Nk = 4 word (kata) yang setiap katanya terdiri dari 32 bit sehingga total kunci 128 bit, ukuran blok teks-asli 128 bit dan memiliki 10 putaran. Sedangkan putaran untuk kunci terdiri dari Ki = 4 kata dan total putaran kunci 128 bit dan kunci yang diperluas mempunyai ukuran 44 kata dan 176 byte.



Gambar 2.7 Proses Umum Enkripsi dan Dekripsi AES
(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :171)

2.1.10 Standar Enkripsi Data

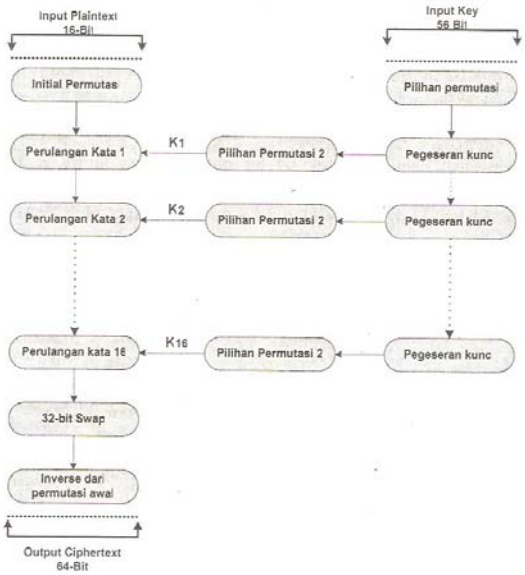
Standar enkripsi data (*Data Encryption Standard* – DES) merupakan algoritma enkripsi yang diadopsi oleh NIST (National Institute of Standard and Technology) sebagai standar pengolahan informasi Federal AS. Secara umum standar enkripsi data terbagi menjadi tiga kelompok, yaitu pemrosesan kunci, enkripsi data 64 bit dan dekripsi data 64 bit yang mana satu kelompok saling berinteraksi satu sama lain.

DES termasuk sistem kriptografi simetri dan tergolong jenis blok kode. DES beroperasi pada ukuran blok 64 bit. DES mengenkripsikan 64 bit teks-asli menjadi 64 bit teks-kode dengan

menggunakan 56 bit kunci internal (internal key) atau upa-kunci (subkey). Kunci internal dibangkitkan dari kunci eksternal (external key) yang panjangnya 64 bit.

Secara umum skema Data Encryption Standard (DES) mempunyai dua fungsi masukan, yaitu :

1. Teks-asli untuk dienkripsi dengan panjang 64 bit.
2. Kunci dengan panjang 56 bit.



Gambar 2.8 Gambaran Umum Algoritma DES

(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :116)

Proses dari permutasi inisial (IP) teks-asli ada tiga :

1. Teks-asli 64-bit di permutasi inisial (IP) dan menyusun kembali bit untuk menghasilkan permutasi masukan.
2. Lagkah untuk melakukan perulangan kata dari teks-asli sebanyak 16 dengan melakukan fungsi yang sama, yang menghasilkan fungsi permutasi subtitusi, yang mana keluaran akhir dari hal tersebut berisi 64-bit (fungsi dari teks-asli dan kunci), masuk ke swap dan menghasilkan pre-output.
3. Pre-output diproses dan permutasi diinversi dari permutasi inisial yang akan menghasilkan teks-kode 64-bit.

Proses dari kunci 56-bit :

1. Kunci melewati fungsi dari permutasi.
2. Penggeseran kunci, yang mana akan dipilih perulangan-perulangan permutasi kunci sebanyak 16 kali yang menghasilkan upa-kunci (Ki) yang diproses dengan kombinasi permutasi.
3. Perbedaan dari upa-kunci (Ki) akan dilakukan penggeseran kunci yang menghasilkan kombinasi teks-asli 64-bit dengan kunci 56-bit.

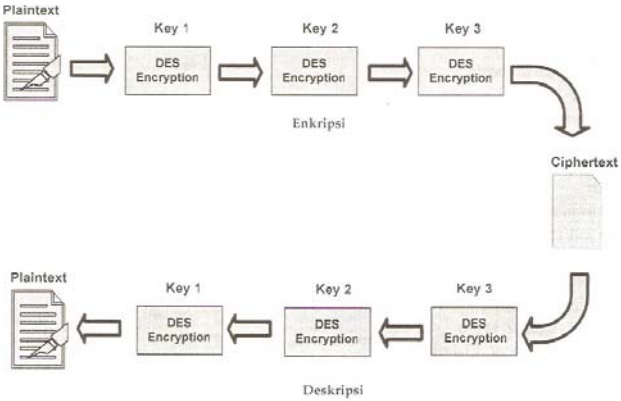
2.1.11 Triple Data Encryption Standard (3DES)

3DES mempunyai perbedaan kecil dengan DES. 3DES lebih aman disbanding DES merupakan pengembangan dari algoritma DES. 3DES mempunyai kunci yang lebih panjang (3 x dari DES).

3DES (*Triple Data Encryption Standard*) merupakan suatu algoritma pengembangan dari algoritma DES (*Data Encryption Standard*). Pada dasarnya algoritma yang digunakan sama, hanya pada 3DES dikembangkan dengan melakukan enkripsi dengan implementasi algoritma DES sebanyak tiga kali. 3DES memiliki tiga

buah kunci yang berukuran 168-bit (tiga kali kunci 56-bit dari DES). Pada algoritma 3DES dibagi menjadi tiga tahap, setiap tahapnya merupakan implementasi dari algoritma DES.

Tahap pertama, plainteks yang diinputkan dioperasikan dengan kunci eksternal pertama (K1) dan melakukan proses enkripsi dengan menggunakan algoritma DES. Sehingga menghasilkan pra-cipherteks pertama. Tahap kedua, pra-cipherteks pertama yang dihasilkan pada tahap pertama, kemudian dioperasikan dengan kunci eksternal kedua (K2) dan melakukan proses enkripsi atau proses dekripsi (tergantung cara pengenkripsian yang digunakan) dengan menggunakan algoritma DES. Sehingga menghasilkan pra-cipherteks kedua. Tahap terakhir, pra-cipherteks kedua yang dihasilkan pada tahap kedua, dioperasikan dengan kunci eksternal ketiga (K3) dan melakukan proses enkripsi dengan menggunakan algoritma DES, sehingga menghasilkan cipherteks (C).



Gambar 2.9 Algoritma 3DES

(sumber : Pengantar Ilmu Kriptografi : Teori, Analisis, dan Implementasi, Halaman :144)

Pada gambar diatas terdapat 3 kunci, yaitu K1,K2,K3. Proses kerja dari 3DES, K1 berfungsi untuk enkripsi, K2 untuk dekripsi, dan K3 untuk enkripsi, atau juga dikenal dengan mode *Encrypt Descrypt Encrypt* (EDE).

2.1.12 Fungsi Hash Satu Arah

Fungsi Hash adalah fungsi yang menerima masukan string yang panjangnya sembarang dan mengonversinya menjadi string keluaran yang panjangnya tetap (fixed), yang umumnya berukuran jauh lebih kecil daripada ukuran semula.

Fungsi Hash Satu Arah (One-Way Hash Function) berfungsi sebagai :

1) Sidik Jari (Fingerprint) : membuat sidik jari dari suatu dokumen atau pesan M yang mana sidik jari merupakan suatu identitas dari si pengirim pesan.
2) Fungsi Kompresi : fungsi kompresi, dokumen D (yang besarnya dapat bervariasi) yang akan di-hash disebut pre-image, sedangkan keluarannya yang memiliki ukuran tetap dalam bentuk aslinya dan pada dasarnya masukan lebih besar dari pada keluaran, seolah-0lah mengalami kompresi, namun hasil dari kompresi tidak bisa dikembalikan ke bentuk awalnya, yang oleh karenanya dinamakan satu arah.
3) Message Digest : dianggap intisari dari suatu dokumen, padahal tidak demikian karena intisari dokumen merupakan ringkasan dokumen yang dapat dipahami maknanya. Message Digest tidak demikian, karena dengan sidik jari orang lain tidak mengerti asli dari dokumen tersebut.

Contoh algoritma fungsi Hash satu arah adalah MD5 dan SHA :

1) Algoritma MD5

Merupakan fungsi Hash yang sering digunakan untuk mengamankan suatu jaringan komputer dan internet yang sengaja dirancang dengan tujuan sebagai berikut :

- Keamanan : hal ini tidak bisa dielakkan karena tidak satupun sistem algoritma yang tidak bisa dipecahkan. Serangan yang sering digunakan untuk menjebol algoritma Hash adalah dengan menggunakan serangan brute force.
- Kecepatan : software yang digunakan mempunyai kecepatan yang tinggi karena didsarkan sekumpulan manipulasi operan 32 bit.
- Simple : tanpa menggunakan struktur data yang kompleks.

2) SHA (Secure Hash Algorithm)

NIST bersama NSA mendesain *Secure Hash Algorithm* (SHA) untuk digunakan sebagai komponen *Digital Signature Standard* (DSS). Standar Hash adalah *Secure Hash Standard* (SHS) dengan SHA sebagai algoritma yang digunakan.

Standar menetapkan SHA yang diperlukan untuk menjamin keamanan *Digital Signature Algorithm* (DSA). Ketika pesan dengan sembarang panjang $< 2^{64}$ bit dimasukkan, SHA menghasilkan 160 bit keluaran yang disebut sebagai *Message Digest* (MD). MD ini kemudian dimasukkan ke dalam DSA, yang menghitung tanda tangan digital untuk pesan tersebut. Penandatanganan MD seringkali meningkatkan efisiensi proses, karena MD biasanya jauh lebih kecil disbanding pesan aslinya.

MD pesan yang sama seharusnya dapat diperoleh oleh pemeriksa tanda tangan ketika menerima pesan dari pengirim dengan cara memasukkan pesan tersebut ke fungsi hash SHA. SHA dikatakan aman karena didesain supaya secara matematis tidak dimungkinkan untuk mendapatkan pesan aslinya bila diberikan hashnya atau tidak mungkin mendapatkan dua pesan yang berbeda yang menghasilkan MD yang sama. SHA dibuat berdasarkan rancangan yang serupa dengan MD4 yang dibuat oleh Profesor Ronal L. Rivest dari MIT. SHAmenghasilkan keluaran sidik jari 160 bit, lebih panjang disbanding MD5.

2.2 Macam-Macam Serangan Cryptanalyst

Terdapat beberapa macam serangan yang mungkin dilakukan oleh pemecah kode (Cryptanalyst), dengan asumsi algoritma enkripsinya telah dikenal luas :

Tabel 2.2 Jenis Serangan Cryptanalyst
(sumber : Kriftografi : Keamanan Internet dan Jaringan Komunikasi, Halaman : 12)

Jenis Serangan	Yang Diketahui Cryptanalyst
Ciphertext Only Attack (hanya tahu kode rahasianya)	Algoritma enkripsi. Ciphertext yang akan dibaca.
Known Plaintext (mengetahui plaintext tertentu)	Algoritma enkripsi. Ciphertext yang akan dibaca. Sepasang atau lebih plaintext-ciphertext yang disusun dengan kunci rahasia tertentu.
Chosen Plaintext	Algoritma enkripsi.

(dapat memilih plaintext)	Ciphertext yang akan dibaca. Plaintext yang dipilih Criptanalyst, bersama dengan ciphertext pasangannya yang dibangkitkan dengan kunci rahasia tertentu.
Adaptive Chosen Plaintext Attack	Algoritma Enkripsi. Ciphertext yang akan dibaca. Plaintext dapat dipilih lebih khusus oleh Cryptanalyst.
Chosen Ciphertext (dapat memilih ciphertext tertentu yang diinginkan)	Algoritma enkripsi. Ciphertext yang akan dibaca. Ciphertext yang isi pokoknya diketahui, dipilih oleh cryptanalyst, bersama dengan plaintext (terdekrip) pasangannya yang dibangkitkan dengan kunci tertentu.
Chosen Text	Algoritma enkripsi. Ciphertext yang akan dibaca. Plaintext yang diplih cryptanalyst, bersama dengan ciphertext pasangannya yang dibangkitkan dengan kunci tertentu. Ciphertext yang isi pokoknyabdiketahui, dipilih oleh cryptanalyst, bersama dengan plaintext (terdekrip) pasangannya yang dibangkitkan dengan kunci tertentu.

2.3 Aspek-Aspek Keamanan Komputer

Keamanan komputer meliputi aspek-aspek berikut, antara lain :

a. Authentication.

   Agar penerima informasi dapat memastikan keaslian pesan, bahwa pesan itu dating dari orang yang dimintai informasi. Dengan kata lain, informasi itu benar-benar dating dari orang yang dikehendaki.

b. Integrity.

   Keaslian pesan yang dikirim melalui jaringan dan dapat dipastikan bahwa informasi yang dikirim tidak dimodifikasi oleh yang tidak berhak.

c. Nonrepudiation.

   Merupakan hal yang berhubungan dengan si pengirim. Pengirim tidak dapat mengelak bahwa dialah yang mengirim informasi tersebut

d. Authority.

   Informasi yang berada pada sistem jaringan tidak dapat dimodifikasi oleh pihak yang tidak berhak untuk mengaksesnya.

e. Confidentiality.

   Merupakan usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. Kerahasiaan ini biasanya berhubungan dengan informasi yang diberikan ke pihak lain.

f. Privacy.

   Lebih ke arah data-data yang bersifat pribadi.

g. Availability.

   Aspek availabilitas berhubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.

h. Access Control.

   Aspek ini berhubungan dengan cara pengaturan akses ke informasi. Hal ini biasanya berhubungan dengan masalah otentikasi dan privasi.

kontrol akses seringkali dilakukan dengan menggunakan kombinasi *user id* dan *password* ataupun dengan mekanisme lain.

2.4 Ancaman Kamanan

Ancaman keamanan yang terjadi terhadap informasi adalah :

a. Interruption

Merupakan ancaman terhadap *availability* informasi, data yang ada dalam sistem komputer dirusak atau dihapus sehingga jika data atau informasi tersebut dibutuhkan maka pemiliknya akan mengalami kesulitan untuk mengaksesnya.

b. Interception.

Merupakan ancaman terhadap kerahasiaan (*secrecy*). Informasi disadap sehingga orang yang tidak berhak dapat mengakses komputer di mana informasi tersebut disimpan.

c. Modification.

Merupakan ancaman terhadap integritas. Orang yang tidak berhak berhasil menyadap lalu-lintas informasi yang sedang dikirim dan kemudian mengubahnyasesuai keinginan orang tersebut.

d. Fabrication.

Merupakan ancaman terhadap integritas. Orang yang tidak berhak berhasil meniru atau memalsukan informasi sehingga orang yang menerima informasi tersebut menyangka bahwa informasi tersebut berasal dari orang yang dikehendaki oleh sipenerima informasi.

2.5 Hacker, Cracker dan Script Kiddies

Hacker dan Cracker merupakan dua buah nama yang sering kita dengar dalam dunia maya, underground, dan cybercrime.tetapi kebanyakan orang memandang kedua nama tersebut adalah sama. Padahal secara prinsip, misi mereka berbeda.

a. Hacker

Merupakan seseorang yang ingin memasuki sebuah sistem untuk mencari informasi, data, atau file dengan tujuan mencari kelemahan yang ada pada sistem tersebut. Kemudian dengan kemampuannya, orang tersebut akan mencoba memperbaiki kelemahan perangkat lunak atau sistem pada komputer tersebut dan akan dipublikasikan secara terbuka pada internet agar sistem tersebut menjadi lebih baik dari sebelumnya.

Karakteristik Hacker :

- Menguasai pemrograman tertentu dan seluk beluk jaringan.
- Senang mempelajari hal-hal baru yang berhubungan dengan sistem keamanan.
- Dapat dengan mudah dan cepat mempelajari pemrograman dan biasanya menggunakan sistem operasi tertentu seperti Unix, Linux, dan lain sebagainya.
- Dapat menghargai hasil karya hacking orang lain.
- Mengerti cara mencari kelemahan pada suatu sistem tanpa harus mengganggu kinerja dari sistem tersebut.[*Wiharsono Kurniawan,2007*].

b. Cracker

Tidak jauh dengan Hacker, tetapi Cracker sifatnya cenderung lebih merusak sistem target apabila terdapat kelemahan pada sistem tersebut. Biasanya Cracker dalam memasuki sistem orang lain akan mem-bypass password atau lisensi yang ada pada program komputer.

Hacker memiliki persamaan karakteristik dengan Cracker, yaitu sama-sama menguasai pemrograman dan mengerti jaringan komputer.

Yang membedakannya adalah Cracker melakukan kegiatannya tersebut untuk maksud jahat, memperoleh keuntungan, atau sebab lain karena merasa mendapat tantangan dari cracker yang lainnya. [*Wiharsono Kurniawan,2007*].

c. Script Kiddies

Berbeda dengan Hacker dan Cracker yang melakukan kegiatannya (hacking dan cracking) karena menguasai pemrograman dan seluk beluk jaringan, Script Kiddies merupakan orang-orang yang hanya menggunakan tool-tool yang telah ada untuk melakukan berbagai aktivitas yang berhubungan dengan keamanan jaringan.[*Wiharsono Kurniawan,2007*].

2.6 Bentuk-Bentuk Serangan

Metode-metode yang digunakan para intruder (hacker,craker) untuk mengontrol komputer anda, secara garis besar dapat digolongkan sebagai berikut [*Rahmat Rafiudin, 2002*]:

- Program-Program Trojan Horse
- Program-Program Back Door dan Remote Administration
- Denial of Service (DoS)
- Membuat perantara penyerangan
- Sharing tak terproteksi
- Mobile Code (Java, JavaScript, dan ActiveX)
- Cross-Site Scripting
- Email Spoofing
- Email-Borne Viruses
- Ekstensi-ekstensi file tersembunyi
- Chat Clients

- Packet Sniffing

2.7 Protokol-Protokol Jaringan

Protokol merupakan himpunan aturan-aturan yang memungkinkan komputer satu dapat berhubungan dengan komputer lain. Aturan-aturan ini meliputi tata cara bagaimana agar komputer bisa saling berkomunikasi, biasanya berupa bentuk (model) komunikasi, waktu (saat berkomunikasi), barisan (traffic saat berkomunikasi), pemeriksaan error saat transmisi data, dan lain-lain .

Protokol jaringan adalah berbagai protokol yang terdapat dari lapisan teratas sampai terbawah yang ada dalam sederetan protokol.

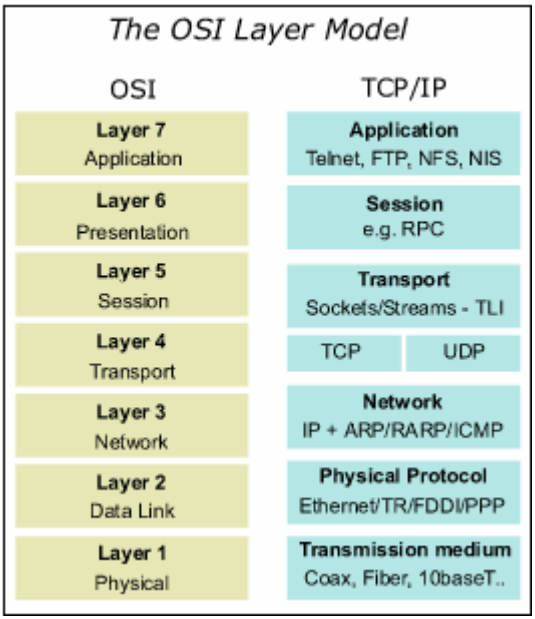
2.8 Elemen Penting Protokol

Elemen-elemen penting dari protokol adalah:

a) **Syntax** mengacu pada strukltur atau format data, yang mana dalam urutan tampilannya memiliki makna sendiri.
b) **Semantic** mengacu pada maksud setiap section bit.
c) **Timing** mengacu pada 2 karakteristik, yakni kapan data harus dikirim dan seberapa cepat data tersebut dikirim.

2.9 Referensi Model OSI

Model ini disebut OSI (*Open System Interconnection*) Reference Model, karena model ini ditujukan untuk pengkoneksian open system, yang dikembangkan oleh *International Organization for Standardization* (ISO) pada tahun 1984. Open system dapat diartikan sebagai suatu sistem yang terbuka untuk berkomunikasi dengan sistem-sistem lainnya.



Gambar 2.10 Model Layer OSI

(sumber : http://zethcorner.files.wordpress.com/2009/01/osi-layer.jpg)

OSI secara konseptual terbagi ke dalam 7 lapisan dimana masing-masing lapisan memiliki fungsi jaringan yang spesifik. Model ini diciptakan berdasarkan sebuah proposal yang dibuat oleh International Standards Organization (ISO) sebagai langkah awal menuju standarisasi protokol internasional yang digunakan pada berbagai layer.

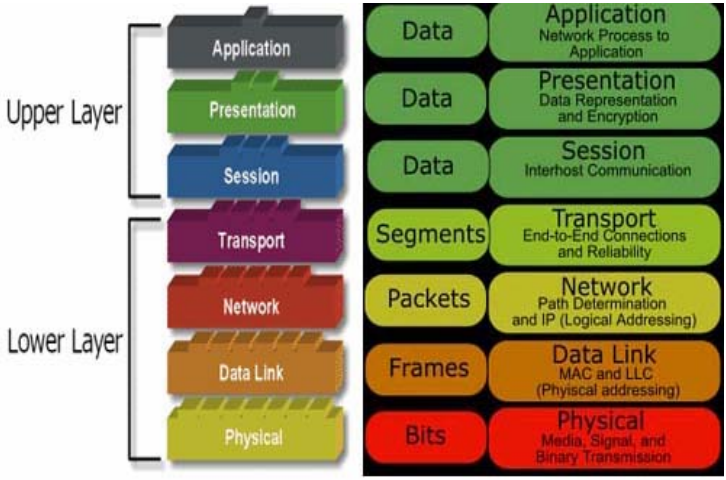
Prinsip-prinsip yang digunakan bagi ketujuh layer tersebut adalah:

1) Sebuah layer haus dibuat bila diperlukan tingkat abstraksi yang berbeda.
2) Setiap layer harus memilki fungsi-fungsi tertentu.

3) Fungsi setiap layer harus dipilih dengan teliti sesuai dengan ketentuan standar protokol internasional.
4) Batas-batas layer diusahakan agar meminimalkan aliran informasi yang melewati interface.
5) Jumlah layer harus cukup banyak sehingga fungsi-fungsi yang berbeda tidak perlu disatukan dalam satu layer di luar keperluannya. Akan tetapi jumlah layer juga harus diusahakan sesedikit mungkin sehingga arsitektur jaringan tidak menjadi sulit dipakai.

2.9.1 Karakteristik Lapisan OSI

Ketujuh lapisan model referensi OSI dapat dibagi kedalam dua kategori, yaitu lapisan atas dan lapisan bawah.



Gambar 2.11 Karakteristik Lapisan OSI

(sumber : http://blog.uad.ac.id/imam_riadi/files/2009/01/osi-layer.jpg)

Lapisan atas dari model OSI berurusan dengan persoalan aplikasi dan pada umumnya diimplementasikan hanya pada software.

Lapisan tertinggi (lapisan aplikasi) adalah lapisan penutup sebelum ke pengguna (user).

Lapisan bawah dari model OSI mengendalikan persoalan transport data. Lapisan fisk dan lapisan data link diimplementasikan ke dalam hardware dan software. Lapisan-lapisan bawah yang lain pada umumnya hanya diimplementasikan dalam software. Lapisan terbawah, yaitu lapisan fisik, adalah lapisan penutup bagi media jaringan fisik (misalnya jaringan kabel), dan sebagai penanggung jawab bagi penempatan informasi pada media jaringan.

2.9.2 Lapisan-Lapisan Model OSI

2.9.2.1 Physical Layer

Berfungsi untuk mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti halnya Ethernet atau Token Ring), topologi jaringan dan pengabelan. Selain itu, level ini juga mendefinisikan bagaimana *Network Interface Card* (NIC) dapat berinteraksi dengan media kabel atau radio.

2.9.2.2 Data Link Layer

Befungsi untuk menentukan bagaimana bit-bit data dikelompokkan menjadi format yang disebut sebagai ***frame***. Selain itu, pada level ini terjadi koreksi kesalahan, *flow control*, pengalamatan perangkat keras (seperti halnya Media Access Control Address (MAC Address)), dan menetukan bagaimana perangkat-perangkat jaringan seperti *hub*, *bridge*, *repeater*, dan *switch layer 2* beroperasi. Spesifikasi IEEE 802, membagi *level* ini menjadi dua level anak, yaitu lapisan *Logical Link Control* (LLC) dan lapisan *Media Access Control* (MAC).

2.9.2.3 Network Layer

Berfungsi untuk mendefinisikan alamat-alamat IP, membuat *header* untuk paket-paket, dan kemudian melakukan routing melalui *internetworking* dengan menggunakan *router* dan *switch layer-3*.

2.9.2.4 Transport Layer

Berfungsi untuk memecah data ke dalam paket-paket data serta memberikan nomor urut ke paket-paket tersebut sehingga dapat disusun kembali pada sisi tujuan setelah diterima. Selain itu, pada level ini juga membuat sebuah tanda bahwa paket diterima dengan sukses (acknowledgement), dan mentransmisikan ulang terhadp paket-paket yang hilang di tengah jalan.

2.9.2.5 Session Layer

Berfungsi untuk mendefinisikan bagaimana koneksi dapat dibuat, dipelihara, atau dihancurkan. Selain itu, di level ini juga dilakukan resolusi nama.

2.9.2.6 Presentation Layer

Berfungsi untuk mentranslasikan data yang hendak ditransmisikan oleh aplikasi ke dalam format yang dapat ditransmisikan melalui jaringan. Protokol yang berada dalam level ini adalah perangkat lunak redirektor (*redirector software*), seperti layanan *Workstation* (dalam Windows NT) dan juga Network shell (semacam *Virtual Network Computing* (VNC) atau *Remote Desktop Protocol* (RDP)).

2.9.2.7 Application Layer

Berfungsi sebagai antarmuka dengan aplikasi dengan fungsionalitas jaringan, mengatur bagaimana aplikasi dapat mengakses jaringan, dan kemudian membuat pesan-pesan kesalahan. Protokol yang berada dalam lapisan ini adalah HTTP, FTP, SMTP, dan NFS.

2.10 Konsep dasar TCP/IP

TCP/IP merupakan pengetahuan dasar bagi seorang network administrator. Tanpa mengenal TCP/IP seorang hacker sekalipun kemungkinan tidak dapat melangkah maju di dunia perhackingan. Dengan kata lain, TCP/IP merupakan awal dari segala hal yang berhubungan dengan jaringan komputer saat ini.

2.10.1 Definisi TCP/IP

TCP/IP adalah sekumpulan protokol yang terdapat di dalam jaringan komputer (network) yang digunakan untuk berkomunikasi atau bertukar data antarkomputer. TCP/IP merupakan protokol standar pada jaringan internet yang menghubungkan banyak komputer yang berbeda jenis mesin maupun sistem operasi agar dapat berinteraksi satu sama lain.

Karena TCP/IP merupakan protokol yang telah diterapkan pada hampir semua perangkat keras dan sistem operasi, maka rasanya tidak ada rangkaian protokol lain yang begitu powerfull kemampuannya untuk dapat bekerja pada semua lapisan perangkat keras dan sistem operasi.

2.10.2 Layanan TCP/IP

Berikut ini layanan “tradisional” yang dilakukan TCP/IP:

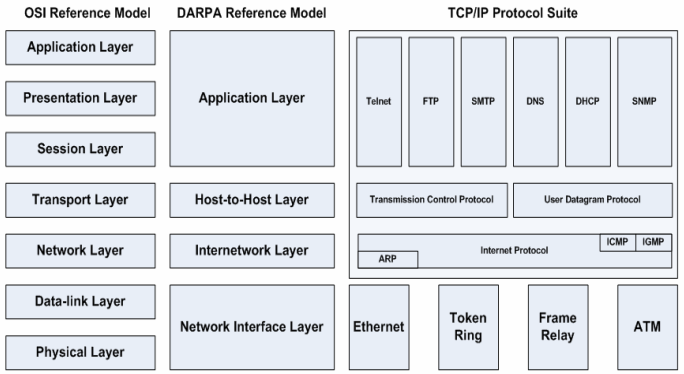
1) Pengiriman File (File Transfer). File Transfer Protokol (FTP) memungkinkan pengguna komputer yang satu untuk dapat mengirim ke ataupun menerima file dari komputer jaringan.

2) Remote Login. Network Terminal Protocol (telnet) memungkinkan pengguna komputer untuk melakukan login ke dalam suatu komputer di dalam suatu jaringan.
3) Computer Mail. Digunakan untuk menerapkan sistem e-mail (elektronik mail).
4) Network File System (NFS). Pelayanan akses file-file jarak jauh yang memungkinkan klien-klien untuk mengakses file-file pada komputer jaringan jaruak jauh walupun file tersebut disimpan secara lokal.
5) Remote Execution. Yang memungkinkan pengguna komputer untuk menjalankan suatu program dari komputer yang berbeda.
6) Name Servers. Nama database alamat yang digunakan pada internet.

2.10.3 Arsitektur TCP/IP

TCP/IP yang merupakan serangkaian protokol, dimana setiap protokol melakukan sebagian atau keseluruhan tugas komunikasi jaringan, tentulah implementasinya tak lepas dari arsitektur jaringan itu sendiri. Arsitektur rangkaian protokol TCP/IP didefinisikan dengan berbagai cara agar fungsi protokol-protokol TCP/IP tersebut dapat saling menyesuaikan.

Protokol TCP/IP itu sendiri merupakan protokol standar yang terdapat pada Referensi Model DoD maupun Referensi Model OSI, berarti hierarki TCP/IP merujuk kepada lapisan 7 lapisan OSI yang setiap lapisannya menyediakan tipe khusus pelayanan jaringan.



Gambar 2.12 Diagram Perbandingan OSI dengan TCP/IP

(sumber : http://indesign14.files.wordpress.com/2008/11/osi-darpa-tcp.png)

2.10.4 Cara kerja TCP/IP

TCP dan IP hanya merupakan protokol yang bekerja pada suatu layer dan menjadi penghubung antara satu komputer dengan komputer lain di dalam network, meskipun kedua komputer tersebut memiliki OS yang berbeda. Untuk memahami lebih jauh, mari kita tinjau proses pengiriman sebuah email.

Dalam pengiriman email ada beberapa prinsip dasar yang harus diperhatikan:

- Pertama, mecakup hal-hal umum seperti siapa yang mengirim email, siapa yan menerima email serta isi dari email tersebut.
- Kedua, bagaimana cara agar email tersebut sampai ke tujuan yang benar.

Dari honsep ini kita mengetahui bahwa pengirim email memerlukan “perantara” yang memungkinkan emailnya sampai ke tujuan. Ini yang menjadi tugas protokol TCP dan IP.

Antara TCP dan IP ada pembagian tugas, yaitu:

- TCP merupakan connection-oriented, yang berarti bahwa kedua komputer yang ikut serta dalam pertukaran data harus melakukan hubungan terlebih dahulu sebelum pertukaran data berlangsung (yang dalam hal ini email). Selain itu TCP juga bertanggungjawab untuk meyakinkan bahwa email tersebut akan sampai ke tujuan, memeriksa kesalahan dan mengirimkan error ke lapisan atas hanya bila TCP tidak berhasil melakukan hubungan (hal inilah yang membuat TCP sukar untuk dikelabuhi). Jika isi email tersebut terlalu besar untuk satu datagram, TCP akan membaginya ke dalam beberapa datagram.
- IP bertanggung jawab setelah hubungan berlagsung. Tugasnya adalah untuk men-rute-kan paket data di dalam network. IP hanya bertugas sebagai kurir dari TCP dan mencari jalur yang terbaik dalam penyampaian datagram. IP "tidak bertanggung jawab" jika data tersebut tidak sampai dengan utuh (hal ini karena IP tidak memiliki informasi mengenai isi data yang dikirimkan), namun IP akan mengirimkan pesan kesalahan (error message) melalui ICMP jika hal ini terjadi dan kemudian kembali ke sumber data.

Karena IP "hanya" mengirimkan data "tanpa" mengetahui urutan data dari mana yang akan disusun berikutnya, maka hal ini menyebabkan IP mudah untuk dimodifikasi di daerah "sumber dan tujuan" datagram. Hal inilah yang menyebabkan adanya paket data yang hilang sebelum sampai ke tujuan.

Datagram dan paket sering dipertukarkan penggunaannya. Secara teknis, datagram merupakan unit dari data yang tercakup dalam protokol. ICMP kependekan dari Internet Control Message Protocol yang bertugas memberikan pesan-pesan kesalahan dan kondisi lain yang memerlukan

perhatian khusus. Pesan/paket ICMP dikirim jika terjadi masalah pada layer IP dan layer di atasnya (TCP dan UDP).

Berikut adalah beberapa pesan potesial yang sering timbul:

a) Destination Unreachable, yang terjadi jika host, jaringan, port, atau protokol tertentu tidak dapat dijangkau.
b) Time Exceded, dimana datagram tidak bisa dikirim karena time to live habis.
c) Parameter Problem, terjadi kesalahan parameter dan letak octet dimana kesahan terdeteksi.
d) Source Quench, yang terjadi karena router/host tujuan membuang datagram karena batasan ruang buffer atau karena datagram tidak dapat diproses.
e) Redirect. Pesan ini member saran kepada host asal datagram mengenai router yang lebih tepat untuk menerima datagram tersebut.
f) Echo Request dan Echo Reply Message. Pesan ini saling mempertukarkan data antara host.

2.10.5 Protokol UDP, TCP, dan IP

1. UDP

User Datagram Protocol (UDP) adalah sebuah protokol yang bekerja pada transport layer, mulai digunakan dan dikembangkan oleh US Department of Defence (DoD) untuk digunakan bersama protokol IP di network layer. Referensi protokol UDP ini terdapat pada RFC 768 yang ditulis oleh John Postel. Protokol UDP memberikan alternative transport untuk proses yang tidak membutuhkan pengiriman yang handal. UDP tidak handal karena tidak menjamin pengiriman data atau perlindungan duplikasi. UDP tidak mengurus masalah penerimaan aliran data dan pembuatan segmen yang sesuai untuk IP.

2. TCP

TCP merupakan protokol yang bertanggung jawab untuk mengirmkan aliran data ke tujuan secara handal, berurutan, dan terdokumentasi secara baik.

3. IP

TCP akan mengirmkan setiap datagram dan meminta IP untuk mengirimkannya ke tujuan (tentu saja dengan cara memberitahukan alamat tujuan pada IP). Inilah tugas IP sebenarnya. IP tidak peduli apa isi dari diagram, atau isi dari TCP header.

Tugas IP sangat sederhana, yaitu hanya mengantarkan datagram tersebut sampai tujuan. Jika IP melewati suatu gateway maka ia kemudian akan menambahkan header miliknya.

2.11 Internet

Interconnected Network atau yang lebih popular dengan sebutan internet adalah sebuah sistem komunikasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia. Setiap komputer dan jaringan terhubung secara langsung maupun tidak langsung ke beberapa jalur utama yang disebut internet backbone dan dibedakan satu dengan yang lain menggunakan unique name yang biasa disebut dengan alamat IP 32 bit.

Secara harafiah, internet (kependekan daripada perkataan ‘internetwork’) adalah rangkaian komputer yang terhubung ke beberapa jaringan lain. Ketika komputer terhubung secara global dengan menggunakan TCP/IP sebagai protocol pertukaran paket data (packet switching communication protocol), maka rangkaian jaringan komputer yang besar ini dapat dinamakan internet. Cara menghubungkan ranglaian komputer dengan kaidah ini dinamakan internetworking.

Internetworking merupakan kumpulan jaringan lokal area, juga metropolitan area yang umumnya terhubung melalui router-router sehingga membentuk jaringan wide area yang begitu besar.

2.12 Email (surat elektronik)

E-mail dalam ilmu komputer adalah singkatan dari electronic mail (surat elektronik), yaitu metode mengirim pesan atau data dari komputer satu ke komputer lainnya melalui jaringan antarkomputer seperti internet atau intranet. Namun sayangnya, kesalahan kita dalam menggunakan email dapat berakibat fatal terhadap kerahasiaan email kita.

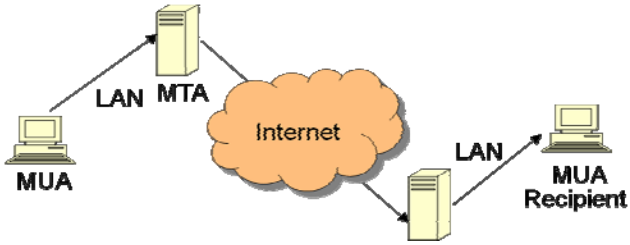
Email (electronic mail) merupakan cara mengirimkan pesan dari pengguna komputer kepada penerima di host tujuan lain. Dengan email ini seseorang dapat berkomunikasi, mengirim pesan, berita atau informasi bahkan dapat berdiskusi. Pesan tersebut terdiri dari :

- Header Lines.
  Memberitahukan komputer bagaimana untuk mengirimkan pesan.
- Message Body.
  Berisi pesan dari mail tersebut dan file attachment.

Header Lines terletak diatas dari pesan surat dan terpisah dari Message Body dengan spasi kosong. Tiap Header Lines dimulai dengan kata kunci yang diakhiri dengan tanda titik dua dan diikuti dengan data.

Email adalah surat elektronik yang penggunaannya sekarang didukung tiga jenis protokol internet, yaitu *Post Office Protocol Versi.3* yang disebut juga sebagai POP3, *Internet Mail Access Protocol* disebut juga IMAP, dan yang ketiga adalah *Simple Mail Transfer Protocol* yang disebut SMTP.

Sistem email terdiri dari dua komponen utama, yaitu *Mail User Agent* (MUA), dan *Mail Tranfer Agent* (MTA).



Gambar 2.13 Proses Bagaimana Email Terkirim

(sumber : Modul Ajar Pens-ITS:Internet Security Halaman : 12)

MUA merupakan komponen yang digunakan oleh pengguna email. Biasanya dia yang disebut program mail. Contoh MUA adalah Eudora, Netscape, Outlook, Pegasus, Thunderbird, pine, mutt, elm, mail, dan masih banyak lainnya lagi. MUA digunakan untuk menuliskan email seperti halnya mesin ketik digunakan untuk menulis surat jaman dahulu.

MTA merupakan program yang sesungguhnya mengantar email. Biasanya dia dikenal dengan istilah mailer. MTA ini biasanya bukan urusan pengguna, akan tetapi merupakan urusan dari administrator. Contoh MTA antara lain postfix, qmail, sendmail, exchange, MDaemon, Mercury, dan seterusnya. MTA merupakan komponen perangkat lunak yang bertujuan untuk menyampaikan pesan keluar serta pesan masuk. MTA berjalan pada protocol komunikasi yang disebut SMTP.

2.12.1 Gambaran Surat Elektronik Internet

Sistem surat elektronik internet menggunakan TCP (Transmission Control Protocol) sebagai protocol transportasinya dan merupakan aplikasi client server. Pada LAN biasa, surat elektronik internet melibatkan tiga langkah terpisah, yaitu :

a. Client megirim pesan dengan menggunakan SMTP.

b. Client menerima pesan dari server menggunakan POP3.
c. Server (atau server ISP) mengirim dan menerima pesan dari server lain menggunakan SMTP.

2.12.2 Protokol SMTP

SMTP (*Simple Mail Transfer Protocol*) merupakan salah satu protokol yang umum digunakan untuk pengiriman surat elektronik di Internet. Protokol ini dipergunakan untuk mengirimkan data dari komputer pengirim surat elektronik ke server surat elektronik penerima. Protokol ini timbul karena desain sistem surat elektronik yang mengharuskan adanya server surat elektronik yang menampung sementara sampai surat elektronik diambil oleh penerima yang berhak.

2.12.3 Protokol POP3

POP3 (*Post Office Protocol version 3*) adalah protokol yang digunakan untuk mengambil surat elektronik (email) dari server email.Protokol ini erat hubungannya dengan protokol SMTP dimana protokol SMTP berguna untuk mengirim surat elektronik dari komputer pengirim ke server. Protokol POP3 dibuat karena desain dari sistem surat elektronik yang mengharuskan adanya server surat elektronik yang menampung surat eletronik untuk sementara sampai surat elektronik tersebut diambil oleh penerima yang berhak. Kehadiran server surat elektronik ini disebabkan kenyataan hanya sebagian kecil dari komputer penerima surat elektronik yang terus-menerus melakukan koneksi ke jaringan internet. Protokol ini dispesifikasikan pada RFC 1939.

2.12.4 Protokol IMAP

IMAP (*Internet Message Access Protocol*) adalah protokol standar untuk mengakses/mengambil e-mail dari server. IMAP memungkinkan pengguna memilih pesan e-mail yang akan ia ambil, membuat folder di server, mencari pesan e-mail tertentu, bahkan menghapus pesan e-mail yang ada. Kemampuan ini jauh lebih baik daripada POP (*Post Office Protocol*) yang hanya memperbolehkan kita mengambil/*download* semua pesan yang ada tanpa kecuali.

2.12.5 Masalah Keamanan Email

Ada beberapa masalah keamanan yang terkait dengan sistem email, yaitu:

- Disadap
- Dipalsukan
- Disusupi virus
- Spamming
- Mailbomb
- Mail Relay

2.12.6 Pendekatan dan Masalah Keamanan Email

Tujuan utama dari keamanan email adalah untuk meyakinkan hal-hal di bawah ini terpenuhi:

- Non repudiasi
- Pesan hanya dibaca oleh penerima yang dituju
- Integritas pesan
- Otentikasi sumber
- Verifikasi pengiriman
- Pelabelan material yang sensitif

- Kontrol akses.

Standar-standar dibawah ini dikembangkan untuk menangani beberapa atau semua masalah diatas :

a. Secure / Multi purpose Internet Mail Extensions (S/MIME).

S/MIME (Secure / Multipurpose Internet Mail Extensions) adalah sebuah protokol yang menambahkan tanda tangan digital dan enkripsi pada pesan-pesan Internet MIME (Multipurpose Internet Mail Extensions) yang didekripsikan pada RFC 1521. MIME adalah format standar resmi yang diajukan untuk Internet electronic mail. Pesan Internet e-mail terdiri dari dua bagian, header dan body. Header membentuk sekumpulan field/pasangan nilai terstruktur yang memberikan informasi penting untuk transmisi pesan. Struktur header dapat ditemukan pada RFC 822. Body pada umumnya adalah tidak terstruktur kecuali e-mail dalam format MIME. MIME menjelaskan bagaimana srtuktur body dari pesan e-mail message. Format MIME mengijinkan e-mail untuk berisi teks, grafik, audio, dan lebih dalam cara standar melalui sistem mail MIME-compliant. MIME sendiri tidak menyediakan layanan keamanan apapun. Tujuan S/MIME adalah untuk menyediakan layanan semacam itu, mengikuti sintaks yang ada pada PKCS #7 untuk tanda tangan digital dan enkripsi.. bagian body dari MIME membawa pesan PKCS #7, dimana ia sendiri adalah hasil dari pemrosesan kriptografik pada bagian body MIME yang lain. Standarisasi S/MIME telah diganti ke IETF, dan sekumpulan dokumen yang menggambarkan S/MIME versi 3 telah dipublish disana.

b. MIME Object Security Services (MOSS)

MOSS menyediakan layanan keamanan email yang lebih fleksibel dengan mendukung model kepercayaan yang berbeda. Diperkenalkan pada tahun 1995, MOSS menyediakan otentikasi, integritas, kerahasiaan, dan non repudiasi untuk email. MOSS menggunakan MD2/MD5, kunci publik RSA, dan DES. MOSS juga mengijinkan identifikasi pengguna diluar standar X.509.

c. Privacy Enhanced Mail (PEM)

PEM adalah standar yang diusulkan oleh IETF untuk menjadi compliant dengan standar kriptografi kunci publik (PKCS), yang dikembangkan oleh konsorsium yang terdiri dari Microsoft, Novell, dan Sun Microsystems. PEM mendukung enkripsi dan otentikasi Internet email. Untuk enkripsi pesan, PEM menggunakan Triple DES-EDE menggunakan sepasang kunci simetris. Algoritma hash, RSA, MD2 atau MD5 digunakan untuk menghasilkan message digest, dan enkripsi kunci publik TSA mengimplementasi tanda tangan digital dan distribusi kunci rahasia. PEM menggunakan sertifikat yang berdasar pada standar X.509 dan dihasilkan oleh CA formal.

d. Pretty Good Privacy

Untuk mempopulerkan keamanan email, Phil Zimmerman mengembangkan software Pretty Good Privacy (PGP). Zimmerman menurunkan nama PGP dari Ralph's Pretty Good Groceries, yang mensponsori acara radio Prairie Home Companion oleh Garrison Keillor. Pada PGP, cipher simetris

IDEA digunakan untuk mengenkrip pesan, dan RSA digunakan untuk pertukaran kunci simetris dan untuk tanda tangan digital. Selain menggunakan CA, PGP menggunakan Web of Trust. Pengguna dapat mensertifikasi satu sama lain dalam mesh model, yang baik diterapkan untuk kelompok yang lebih kecil. PGP berbeda dengan teknik enkripsi konvensional. PGP menggunakan dua kunci melakukan proses enkripsi dan dekripsi. Teknik konvensional menggunakan kunci yang sama utuk melakukan proses enkripsi dan dekripsi.

Cara kerja PGP :

PGP mengkombinasikan fitur-fitur terbaik yang terdapat pada kriptografi konvensional dengan kriptografi kunci umum. PGP merupakan sistem kriptografi hibrida. Enkripsi pada PGP menggunakan kriptografi kunci umum dan juga sistem yang menggabungkan kunci umum tersebut dengan identitas pengguna. Versi pertama dari sistem ini memperkenalkan skema *web of trust* yang berbeda dengan sistem X.509 yang menggunakan pendekatan berdasarkan otoritas sertifikat (*Authority Certificate*). Versi terbaru dari PGP menyediakan kedua alternative tersebut melalui manajemen server secara otomatis.

Enkripsi email pada PGP menggunakan algoritma enkripsi kunci asimetri dengan pasangan kunci-umum-kunci-privat. Pengirim email menggunakan kunci umum penerima untuk melakukan enkripsi kunci rahasia yang digunakan pada algoritma kode simetri. Pada akhirnya kunci akan digunakan untuk melakukan enkripsi teks-asli.

Penerima email yang terenkripsi menggunakan kunci sesi (*session key*) untuk melakukan dekripsi terhadap email tersebut. Kunci sesi ini terdapat pada email yang terenkripsi dan diperoleh dengan mendekripsinya menggunakan kunci rahasia. PGP menyediakan cara untuk mendistribusikan kunci umum dengan menggunakan sertifikat identitas yang dibangkitkan dengan algoritma kriptografi. PGP telah memberikan suatu skema pembuatan sertifikat secara internal yang disebut *Web of Trust.*

Web of Trust adalah suatu model kepercayaan yang komulatif. Sebuah sertifikat dapat dipercaya langsung atau dipercaya memlalui perantaraan sertifikat lain. Ketika pengguna menandatangani kunci lain maka pengguna tersebut akan menjadi pengenal (*introducer*) bagi kunci tersebut. Dengan demikian terbentuklah suatu jaringan kepercayaan yang disebut *Web f Trust.* Pada lingkunag PGP, setiap pengguna dapat berperan sebagai pihak yang berwenang terhadap sertifikat. Setiap pengguna dapat melakukan validasi terhadap sertifikat kunci umum milik pengguna lain. Akan tetapi suatu sertifikat hanya akan dianggap valid jika pengguna tersebut percaya ke pengguna lain yang berperan sebagai pengenal.

Pada spesifikasi OpenPGP, tanda tangan yang terpercaya dapat digunakan untuk mendukung pembuatan sertikat otoritas (*Authority Certificate*). Tanda tangan yang terpercaya ini menandakan bahwa suatu kunci merupakan milik pemiliknya dan pemilik kunci tersebut dipercaya juga untuk menandatangani kunci lain yang memiliki tingkat kepercayaansatu tingkat dibawahnya. Tanda tangan tingkat 0 dapat dibandingkan dengan tanda tangan *Web of Trust* karena hanya validasi kunci yang disertifikasi. Tanda tangan tingkat 1 sama dengan kepercayaan

yang diberikan seseorang pada sertifikat otoritas karena kunci yang dengan tanda tangan tingkat 1 dapat membuka tanda tangan tingkat 0 tanpa adanya batasan jumlah. Tanda tangan tingkat 2 dapat disamakan dengan asumsi kepercayaan pengguna ketika menggunakan sertifikat otoritas pada Internet Explorer dan memungkinkan pemilik kunci untuk membuat sertifikat otoritas kunci lainnya. PGP juga memiliki fitur untuk membatalkan (*revoke*) sertifikat identitas yang sudah tidak valid. Hal ini kurang lebih sama dengan *certificate revocation list* pada skema *Publik Key Infrastruktur.* PGP versi terakhir juga mendukung fitur untuk memeriksa sertifikat yang sudah tidak berlaku lagi.

2.13 Konsep Dasar Gnu Privacy Guard atau GPG

GnuPG menggunakan beberapa konsep termasuk kriptografi cipher simetrik, cipher kunci publik, dan hashing satu arah.

2.13.1 Chiper Simetrik

Sebuah cipher simetris adalah cipher yang menggunakan kunci yang sama untuk kedua enkripsi dan dekripsi. Dua pihak berkomunikasi menggunakan andi simetris harus setuju pada kunci sebelumnya. Begitu mereka setuju, pengirim mengenkripsi pesan menggunakan kunci, mengirimnya ke penerima, dan penerima pesan decrypts menggunakan kunci. Contoh cipher simetris modern termasuk 3DES, Blowfish, dan IDEA. Cipher yang baik meletakkan semua keamanan di kunci dan tidak ada dalam algoritma. Dengan kata lain, seharusnya tidak membantu bagi seorang penyerang jika dia tahu sandi yang sedang digunakan. Hanya jika ia memperoleh pengetahuan akan kunci dari algoritma diperlukan. Karena semua keamanan adalah terletak di kunci, maka penting sehingga sangat

sulit untuk menebak kunci. Dengan kata lain, kemungkinan himpunan kunci, yaitu ruang kunci, perlu besar.

2.13.2 Chiper Kunci Publik

Masalah utama dengan sandi simetris bukan keamanan mereka tetapi dengan pertukaran kunci. Setelah pengirim dan penerima sudah saling bertukar kunci, kunci yang dapat digunakan untuk berkomunikasi dengan aman, tapi apa saluran komunikasi aman digunakan untuk berkomunikasi kunci itu sendiri? Secara khusus, hal itu mungkin akan lebih mudah bagi penyerang untuk bekerja untuk menghalangi kunci daripada mencoba semua kunci di ruang kunci. Sandi kunci publik diciptakan untuk menghindari masalah pertukaran kunci seluruhnya. Sebuah sandi kunci publik menggunakan sepasang kunci untuk mengirim pesan. Dua kunci milik orang yang menerima pesan. Salah satu kuncinya adalah sebuah kunci publik dan dapat diberikan kepada siapa pun. Kunci lain adalah sebuah kunci pribadi dan rahasia disimpan oleh pemiliknya. Seorang pengirim mengenkripsi pesan menggunakan kunci publik dan sekali dienkripsi, hanya kunci pribadi dapat digunakan untuk dekripsi.

Semua yang diperlukan adalah bahwa beberapa waktu sebelum komunikasi rahasia si pengirim mendapatkan salinan kunci publik penerima. Lebih jauh lagi, satu kunci publik dapat digunakan oleh siapa pun yang ingin berkomunikasi dengan penerima.Sandi kunci publik didasarkan pada satu arah fungsi trapdoor. Sandi kunci publik berdasarkan faktorisasi prima, kunci publik berisi nomor komposit terbuat dari dua faktor prima besar, dan algoritma enkripsi yang menggunakan komposit untuk mengenkripsi pesan. Algoritma untuk mendekripsi pesan membutuhkan pengetahuan faktor-faktor utama, sehingga dekripsi adalah mudah jika Anda memiliki kunci pribadi yang berisi salah satu faktor, tetapi sangat sulit jika Anda tidak memilikinya.

Seperti sandi simetris, dengan baik sandi kunci publik semua terletak keamanan pada kunci. Oleh karena itu, ukuran kunci adalah ukuran dari sistem keamanan, tetapi orang tidak dapat membandingkan ukuran kunci sandi yang simetris dan sebuah kunci publik kunci cipher sebagai ukuran relatif keamanan mereka. Dalam brute force serangan terhadap sebuah cipher simetrik dengan ukuran kunci 80 bit, penyerang harus menghitung sampai 2^{80} kunci untuk menemukan kunci yang cocok. Dalam brute force menyerang sebuah sandi kunci publik dengan ukuran kunci 512 bit, penyerang harus faktor nomor komposit dikodekan dalam 512 bit (hingga 155 angka desimal). Beban kerja untuk penyerang pada dasarnya berbeda tergantung pada cipher dia menyerang. Sementara 128-bit yang cukup untuk cipher simetrik, mengingat teknologi saat ini anjak public key with 1024 bit dianjurkan untuk sebagian besar tujuan.

2.13.3 Chiper Hibrida

Sandi Kunci Publik tidaklah selalu ampuh. Banyak sandi simetris yang lebih kuat dari segi keamanan, dan enkripsi kunci publik dan dekripsi lebih mahal daripada operasi yang sesuai dalam sistem simetris. Sandi kunci publik tetap saja alat yang efektif untuk mendistribusikan kunci cipher simetrik, dan itu adalah bagaimana mereka digunakan dalam sistem penyandian hibrida.

Sebuah sandi hibrida menggunakan kedua sandi yaitu sandi simetris dan sandi kunci publik. Ini bekerja dengan menggunakan sandi kunci publik untuk berbagi kunci untuk sandi simetrik. Pesan yang sebenarnya dikirim kemudian dienkripsi menggunakan kunci dan dikirim ke penerima. Karena kunci simetris berbagi adalah aman, kunci simetris yang digunakan adalah berbeda untuk setiap pesan yang dikirim. Oleh karena itu kadang-kadang disebut kunci sesi.

Baik PGP dan GnuPG menggunakan sandi hibrida. Kunci sesi, dienkripsi dengan menggunakan kunci publik sandi, dan pesan sedang dikirim, dienkripsi dengan sandi simetris, secara otomatis digabungkan dalam satu paket. Penerima menggunakan kunci pribadinya untuk mendekripsi session key dan session key ini kemudian digunakan untuk mendekripsi pesan.

2.13.4 Tanda Tangan Digital

Sebuah fungsi hash adalah fungsi many-to-one yang memetakan input ke nilai dalam sebuah himpunan berhingga. Biasanya ini adalah menetapkan berbagai bilangan natural. Fungsi hash sederhana adalah f (x) = 0 untuk semua bilangan bulat x. Sebuah fungsi hash yang lebih menarik adalah f (x) = x mod 37, yang memetakan x ke sisa membagi x dengan 37. Sebuah dokumen tanda tangan digital adalah hasil dari penerapan fungsi hash pada dokumen. Beberapa sandi kunci publik dapat digunakan untuk menandatangani dokumen. Para penandatangan mengenkripsi dokumen dengan kunci pribadi. Siapa saja yang ingin untuk memeriksa tanda tangan dan melihat dokumen hanya menggunakan kunci publik penandatangan untuk mendekripsi dokumen.

Sebuah alternatif lain adalah dengan menggunakan fungsi hash dirancang untuk memenuhi dua sifat penting ini. SHA dan MD5 adalah contoh dari algoritma tersebut. Menggunakan semacam algoritma, sebuah dokumen yang ditandatangani oleh hashing itu, dan nilai hash adalah tanda tangan. Orang lain dapat memeriksa tanda tangan oleh mereka juga hashing salinan dokumen dan membandingkan nilai hash mereka dengan nilai hash dari dokumen asli. Jika mereka cocok, hampir pasti bahwa dokumen identik.

Tentu saja, masalahnya sekarang adalah dengan menggunakan fungsi hash untuk tanda tangan digital tanpa mengizinkan penyerang untuk

mengganggu dengan memeriksa tanda tangan. Bila dokumen dan tanda tangan dikirim tidak terenkripsi, penyerang dapat memodifikasi dokumen dan menghasilkan tanda tangan yang sesuai tanpa sepengetahuan penerima. Kalau saja dokumen dienkripsi, seorang penyerang dapat merusak tanda tangan dan menyebabkan tanda tangan cek untuk gagal. Pilihan ketiga adalah dengan menggunakan hibrida enkripsi kunci publik untuk mengenkripsi baik tanda tangan dan dokumen. Para penandatangan menggunakan kunci pribadinya, dan siapa pun dapat menggunakan kunci publik untuk memeriksa tanda tangan dan dokumen. Jika algoritma ini benar-benar mengamankan dokumen itu juga aman dari gangguan dan tidak akan ada kebutuhan untuk tanda tangan. Masalah yang lebih serius, bagaimanapun, adalah bahwa ini tidak melindungi baik tanda tangan atau dokumen dari gangguan. Dengan algoritma ini, hanya kunci sesi untuk cipher simetrik dienkripsi menggunakan kunci pribadi penanda tangan. Siapa saja dapat menggunakan kunci publik untuk memulihkan kunci sesi. Oleh karena itu, sangat mudah bagi penyerang untuk memulihkan kunci sesi dan menggunakannya untuk mengenkripsi dokumen dan tanda tangan pengganti untuk dikirimkan kepada orang lain dalam nama pengirim.

Secara khusus, nilai hash dienkripsi dengan menggunakan kunci pribadi penandatangan, dan siapa pun dapat memeriksa tanda tangan menggunakan kunci publik. Dokumen yang sudah ditandatangani dapat dikirim menggunakan algoritma enkripsi lainnya tidak termasuk jika itu adalah dokumen publik. Bila dokumen adalah memodifikasi signature check akan gagal, namun inilah apa yang seharusnya memeriksa tanda tangan untuk menangkap. The Digital Signature Standard (DSA) adalah algoritma tanda kunci publik yang berfungsi sebagai baru saja dijelaskan. DSA adalah algoritma penandatanganan utama yang digunakan dalam GnuPG.

BAB III
GNU PRIVACY GUARD PADA EMAIL BERBASIS WINDOWS

3.1 Metodologi Penelitian

3.1.1 Tahapan Penelitian

Tahapan penelitian merupakan unsur sistematis di dalam melakukan penelitian. Pada penyusunan tugas akhir ini tahapan analisis yang dilakukan secara umum sebagai berikut :

- Melakukan studi literature.
- Pemilihan platforms atau sistem operasi.
- Pemilihan type GPG.
- Melakukan analisis terhadap GPG.
- Menguji coba Gnu Privacy Guard atau GPG dalam pengiriman email.
- Mengambil kesimpulan dan hasil pengujian.

3.1.2 Metode Penelitian

Metode penelitian yang digunakan dalam rangka mengumpulkan data adalah sebagai berikut :

- Studi Literatur

  Melakukan studi literature untuk mendapatkan data-data yang diperlukan. Data-data tersebut didapatkan melalui buku-buku perpustakaan, artikel, jurnal, paper, makalah, dan blog yang didapat melalui situs-situs di internet.

3.1.3 Tempat dan Waktu Penelitian

Penelitian dilakukan di tempat yang memungkinkan terdapatnya sumber data atau informasi, seperti perpustakaan dan internet. Waktu pelaksanaannya adalah dari bulan September 2009 sampai dengan bulan Februari 2010.

3.1.4 Metode Analisis

Dalam penelitian ini penulis menggunakan metode analisis deskriptif untuk menganalisa data-data yang diperoleh.

3.1.5 Tahapan Pengujian

Tahapan uji coba yang dilakukan dalam penelitian adalah sebagai berikut :

- Pemilihan platforms atau sistem operasi untuk Gnu Privacy Guard atau GPG. Sistem operasi yang digunakan adalah sistem operasi Windows XP SP2. Sistem operasi Windows diplih karena mudah dalam pemakaian (*User Friendly).*
- Pemilihan tipe Gnu Privacy Guard. Penulis memakai *gnupg-w32cli-1.4.10b.* yang merupakan GPG versi Windows berupa perinta baris (*Command Line).*
- Analisis Gnu Privacy Guard. Melakukan analisis terhadap GPG. Peran serta kapabilitas dengan keamanan email.
- Uji coba Gnu Privacy Guard. Melakukan pengujian GPG untuk mendapatkan pembuktian seperti yang diharapkan, yaitu khususnya dalam hal mengenkripsi dan dekripsi pesan dalam hal keamanan pada pengiriman sebuah email.

3.2 Gnu Privacy Guard

(Sapty, F. R., 2005) GNU *Privacy Guard* (GnuPG, atau GPG) merupakan software enkripsi email pengganti PGP yang lengkap dan bebas. GnuPG adalah suatu program yang digunakan untuk mengamankan komunikasi dan penyimpanan data. Program ini dapat menyandikan data serta membuat tanda tangan digital. Karena tidak menggunakan algoritma yang dipatenkan, GnuPG dapat digunakan secara bebas. GnuPG menggunakan kriptografi *Public key* (*public key cryptography*) sehingga para penggunanya dapat saling berkomunikasi secara aman. Dalam sistem *Public key*, setiap pengguna mempunyai sepasang kunci yang terdiri dari *Private key* dan *Public key*. *Private key* dirahasiakan (hanya diketahui oleh pemiliknya), sementara *Public key* dapat diberikan pada siapa saja yang dikehendaki pemilik, sehingga pemilik dapat berkomunikasi dengan pengguna lain yang diberi *Public key* tersebut. GnuPG dibuat oleh tim GnuPG yang terdiri dari Matthew Skala, Michael Roth, Niklas Hernaeus, R Guyomarch and Werner Koch. Gael Queri, Gregory Steuck, Janusz A. Urbanowicz, Marco d'Itri, Thiago Jung Bauermann, Urko Lusa and Walter Koch yang membuat translasi resmi dan Mike Ashley yang mengerjakan *GNU Privacy Handbook.*

GnuPG adalah software enkripsi email pengganti PGP yang lengkap dan bebas. Bebas karena tidak menggunakan algoritma enkripsi yang telah dipatenkan sehingga bisa dipakai oleh siapa saja tanpa batasan. GnuPG memenuhi spesifikasi OpenPGP RFC2440. Beberapa fitur yang ditawarkan GnuPG adalah:

1) Penggantian penuh terhadap pemakaian PGP
2) Tidak menggunakan algoritma yang telah dipatenkan
3) Bebas, berlisensi GNU Public Lisence (GPL)
4) Fungsi yang lebih baik dibandingkan PGP
5) Kompatibel dengan PGP versi 5 dan yang lebih tinggi

6) Mendukung algoritma ElGamal (signature dan enkripsi), DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160, dll.
7) Mudah diimplementasikan jika ada algoritma baru (penggunaan *extension modules*)
8) Banyak bahasa yang sudah mentranslasikan
9) Terintegrasi dengan HKP keyservers.
10) Mempunyai program antarmuka grafis (GUI).

Pembuatan kunci dapat menggunakan algoritma RSA, DSA maupun ElGamal. Kemudian ditentukan panjang kunci yang akan dibuat. Yang harus dipertimbangkan dalam menentukan panjang kunci, seseorang harus memilih antar aspek kerahasiaan dan waktu yang diperhitungkan. Jika kunci semakin panjang resiko untuk meng-*crack* pesan akan menurun. Tetapi dengan kunci yang lebih besar waktu kalkulasinya juga bertambah. Panjang kunci minimal pada GnuPG adalah 768 bit. Untuk DSA 1024 bit adalah ukuran yang standar. Password digunakan agar dapat menggunakan fungsionalitas yang dimiliki kunci rahasia. Setelah semua data dimasukkan sistem mulai menggenerate kunci. Proses ini membutuhkan waktu beberapa lama.

Orang lain dapat menghubungi seseorang secara aman dengan kunci publik yang telah dibuat. Mempublish kunci dapat dilakukan dengan mempublishnya pada *homepage* sendiri (dengan *finger*) melalui server kunci . Saat menerima kunci publik dari orang lain harus dilakukan penambahan data ke basis data kunci.

Sekali sebuah kunci diimpor harus dilakukan validasi. GnuPG menggunakan model kepercayaan yang handal dan fleksibel yang tidak membutuhkan seseorang untuk secara personal memvalidasi setiap kunci yang diimpor. Tetapi beberapa kunci mungkin harus divalidasi secara

personal. Sebuah kunci divalidasi dengan memverifikasi *fingerprint* kunci dan kemudian menandatangani kunci untuk mensertifikasinya sebagai kunci yang valid.

User dapat menarik/mencabut kembali kuncinya untuk beberapa alasan. Misalnya kunci rahasia telah dicuri atau menjadi tersedia untuk orang yang salah. Jika user melupakan passphrasenya atau jika kunci pribadi telah diketahui orang lain atau hilang, sertifikat ini dapat dipublikasikan untuk memberitahukan orang lain bahwa kunci publik tidak dapat digunakan kembali. Kunci publik yang ditarik kembali tetap dapat digunakan untuk memverifikasi tanda tangan yang dibuat di masa lalu, tetapi tidak dapat digunakan lagi untuk mengenkripsi pesan. Juga berdampak pada kemampuan untuk mendekrip pesan yang dikirimkan kepadanya di masa lalu jika user tetap memiliki akses ke kunci privat.

3.3 Cara Kerja Gnu Privacy Guard

Sebagaimana yang telah ditetapkan dalam standar OpenPGP, GnuPG menyediakan layanan integritas pesan dan file data dengan teknologi tanda tangan digital, enkripsi, kompresi, dan konversi Radix-64. GnuPG juga menyediakan layanan manajemen dan sertifikat kunci.

GPG mengkombinasikan fitur-fitur terbaik yang terdapat pada kriptografi konvensional dengan kriptografi kunci umum. GPG merupakan sistem kriptografi hibrida. Enkripsi pada GPG menggunakan kriptografi kunci umum dan juga sistem yang menggabungkan kunci umum tersebut dengan identitas pengguna.

Enkripsi email pada GPG menggunakan algoritma enkripsi kunci asimetri dengan pasangan kunci-umum-kunci-privar. Pengrim email menggunakan kunci umum penerima untuk melakukan enkripsi kunci rahasia yang digunakan pada algoritma kode simetri.

Untuk menjamin kerahasiaan pesan atau file data, GnuPG menggunakan kombinasi kriptografi kunci-kunci simetrik dan kriptografi kunci-publik. Adapun langkah-langkah menjaga kerahasian data pada pengiriman suatu pesan dengan melakukan enkripsi pada GnuPG adalah sebagai berikut:

1. Pengirim membuat pesan.
2. Pengirim membangkitkan sebuah bilangan acak atau memberikan sandi lewat (*passphrase*) sebagai *session key* untuk pesan saat ini.
3. Pengirim mengenkripsi *session key* tersebut dengan kunci publik masing-masing penerima. Hasil enkripsi *session key* ini menjadi awal dari pesan yang dikirim.
4. Pengirim mengenkripsi pesan (yang biasanya sudah dikompresi) yang akan dikirim dengan menggunakan *session key*.
5. Penerima mendekripsi pesan dengan kunci privatnya.
6. Penerima mendekripsi pesan dengan *session key*. Jika pesan yang diterima merupakan hasil kompresan, maka pesan harus didekompresi.

Baik layanan tanda tangan digital maupun kerahasiaan, bisa diaplikasikan pada pesan yang sama. Caranya, tanda tangan digital dibangkitkan dan dibubuhkan pada pesan. Lalu, pesan dan tanda tangan tersebut dienkripsi dengan menggunakan *session key* yang simetrik. Lalu, *session key* dienkripsi menggunakan enkripsi kunci-publik dan ditaruh di awal blok yang terenkripsi.

Sedangkan langkah-langkah otentifikasi yang dilakukan GnuPG melalui tanda tangan digital adalah sebagai berikut:

1. Pengirim membuat pesan.
2. Pengirim membangkitkan kode *hash* dari pesan.
3. Pengirim membangkitkan tanda tangan digital dari kode *hash* pesan dengan menggunakan kunci privat pengirim.

4. Tanda tangan digital tersebut dilekatkan pada pesan.
5. Penerima menerima pesan yang bertanda tangan.
6. Penerima membangkitkan kode *hash* dari pesan yang diterima dan memverifikasinya dengan menggunakan tanda tangan digital pada pesan. Jika verifikasi berhasil, berarti pesan yang diterima tersebut otentik.

3.4 Layanan Gnu Privacy Guard

Tabel 3.1 Layanan Pada Gnu Privacy Guard

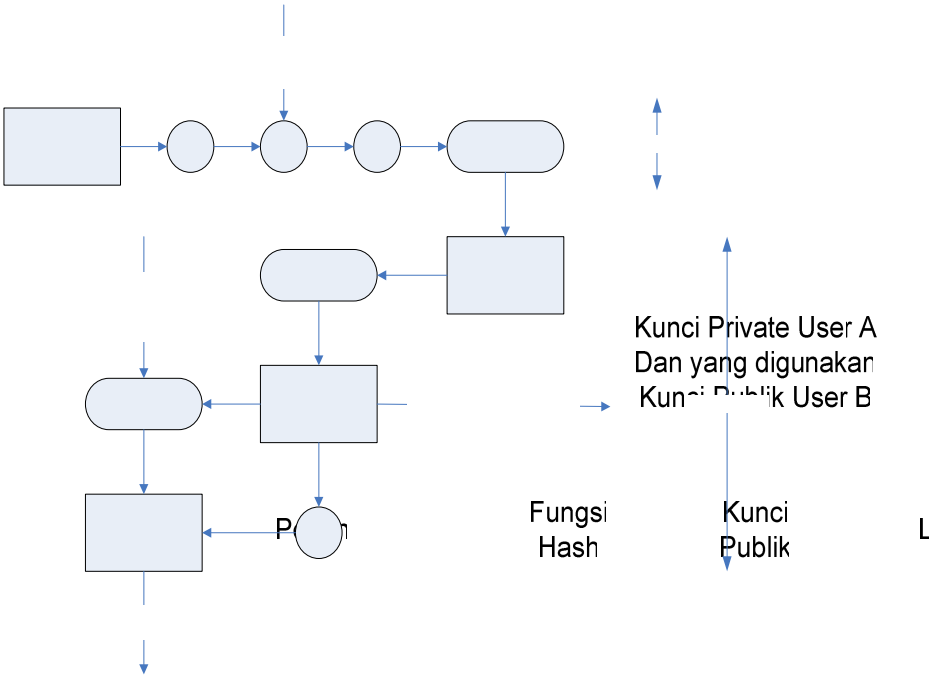
Fungsi	**Algoritma yang Digunakan**	**Deskripsi**
Enkripsi Pesan	3DES, RSA	Pesan yang dienkripsi dengan menggunakan 3DES dengan sesi sekali pakai yang dibangkitkan oleh pengirim. Kunci sesi dienkripsi menggunakan RSA dengan kunci penerima dan digabungkan ke dalam pesan.
Tanda Tangan Digital	DSA, MD5	Kode Hash pesan dibuat menggunakan MD5. Hash ini dienkripsi menggunakan DSA dengan kunci rahasia pengirim dan digabungkan ke dalam pesan.
Kompresi	ZIP	Pesan dikompres, disimpan dan dikirim dengan menggunakan

		file ZIP.
Kompatibilitas email	Konversi Radix(base)64	Untuk mempermudah penggunaannya dalam aplikasi email, pesan yang terenkripsi dapat dikonversi ke dalam string ASCII menggunakan konversi radix(base)64.

Layanan keamanan GnuPG secara umum dalam mengirimkan pesan di antaranya adalah :

3.4.1 Otentikasi (Authentication)

Authentication waktu mengirimm pesan adalah hal yang sangat penting dalam hal keamanan data. Meskipun tanda tangan ummumnya ditempelkan ke pesan atau file, tanda tangan terpisah juga dapat digunakan. Tanda tangan yang terpisah itu dapat dikirim secara terpisah dengan pesan. Tanda tangan yang terpisah dapat dieksekusi dan dapat digunakan untuk mengecek apakah program tersebut masih asli atau sudah diserang virus ataupun Trojan horse.tanda tangan yang terpisah juga dapat digunakan untuk menandatangani seluruh dokumen yang terpisah. Yang perlu diingatadalah bahwa pada setiap tanda tangan yang terpisah terdapat tanda tangan lain proses kerja sistem pemeriksaan keabsahan tersebut iilustrasikan pada gambar dibawah ini :



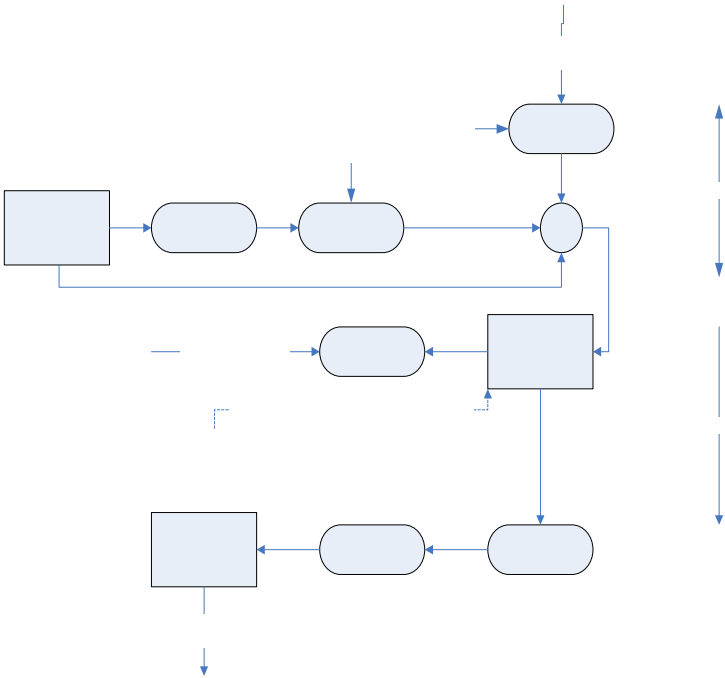
Gambar 3.1 Proses Otentikasi

Contoh otentikasi pesan :

1) Pengirim membuat pesan.
2) Dengan menggunakan kode SHA 160-bit untuk pesan tersebut.
3) Kode-kode Hash dienkripsi dengan menggunakan RSA kunci-rahasia dari pengirim dan pesan di pra-pending.
4) Penerima menggunakan RSA dengan kunci umum pengiim untuk melakukan dekirpsi dan mengekstraksikan kode-kode Hash.
5) Penerima mendapatkan kode Hash baru untuk pesan tersebut dan kemudian membandingkan kode-kode Hash yang didapatkan itu dan bilamana cocok maka berarti pesan tersebut otentik

3.4.2 Perahasiaan (Confidentialty)

Layanan lain yang diberikan GPG adalah Confidentialty yang berfungsi sebagai pengirim pesan dan merahasiakan pesan tersebut. Pada kasus ini digunakan enkripsi simetri algoritma CAST-128 dan juga bisa IDES (3DES). Kunci GPG simetri hanya sekali digunakan. Setiap pesan di enkrip, kunci generative 128 bit untuk pesan tersebut digunakan secara acak untuk melindungi kunci supaya tidak diketahui oleh orang lain. Oleh sebab itu kunci sesi akan dienkripsi menggunakan kunci-umum si penerima. Ilustrasinya bisa dilihat pada gambar dibawah ini :



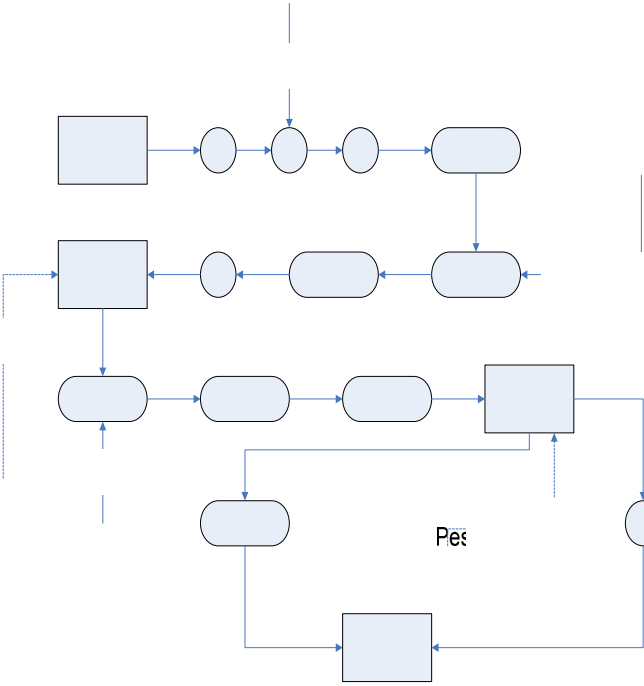
Gambar 3.2 Proses Confidentialty

Contoh pesan rahasia :

1) Pesan generatif pengguna menggunakan sesi kunci random 128 bit untuk pesan.
2) Pesan dienkripsi menggunakan CAST-128 (atau 3DES) dengan sesi kunci.
3) Sesi kunci dienkripsi menggunakan algoritma RSA, kunci umum penerima juga digunakan dan pesan di pra-pending.
4) Penerima menggunakan RSA dengan kunci rahasia untuk melakukan dekripsi dan menemukan kembali sesi kunci.
5) Sesi kunci akan mendekripsikan pesan yang diterima.

3.4.3 Otentikasi dan Perahasiaan

Dengan menggunakan dua fasilitas, yaitu Authentication dan Confidentialty, penerima akan menerima tanda tangan digital dan pesan yang terjamin kerahasiaan dan keabsahannya. Proses tersebut diilustrasikan pada gambar di bawah ini :



Gambar 3.3 Proses Otentikasi dan Perahasiaan

Pesan

Link

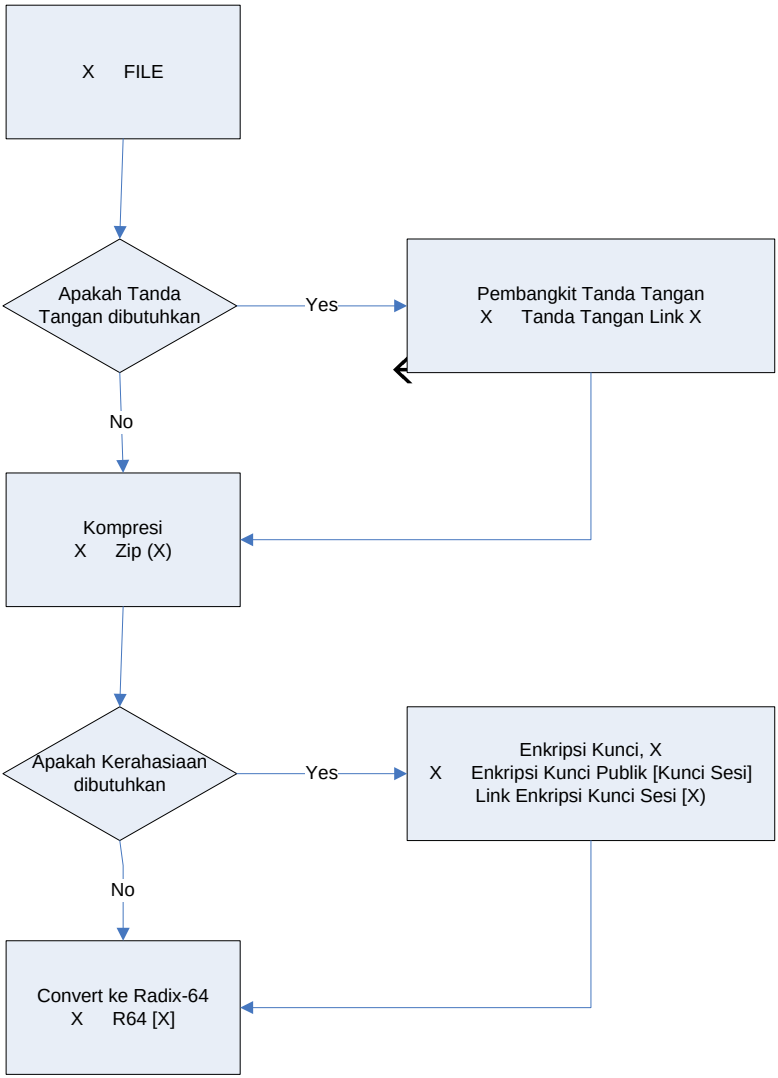
Enkripsi Publik Key
(Kunci Sesi yang digunakan
Kunci Simmetric))

Dekripsi
Kunci
Publik

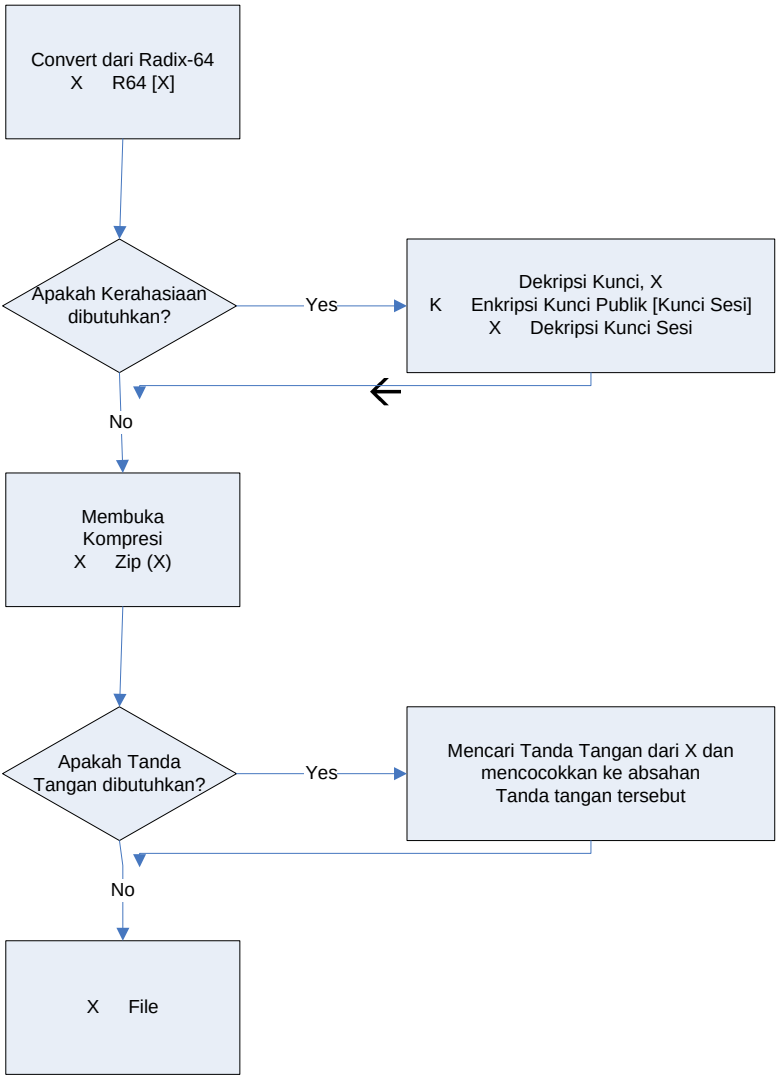
Dekripsi
Simetris

Kunci Private User B
Dan yang digunakar
Kunci Publik User A

Dekripsi
Kunci
Publik



Gambar 3.4 Flow Chart Proses Pengiriman Pesan



Gambar 3.5 Flow Chart Proses Penerimaan Pesan

3.4.4 Kompresi

Setting awal (*default*), GPG sudah melakukan kompresi terhadap pesan yang akan dikirim setelah ditanda-tangani sebelum enkripsi terjadi. Fungsi ini berguna untuk mengurangi ukuran file yang besar yang akan dikirim melalui jarinngan komputer supaya tidak terjadi penumpukan sewaktu pengiriman. Pada dasarnya kompresi algoritma yang akan melalui jaringan akan selalu diinversi jika telah sampai pada tujuan. Tanda tangan digital ditempatkan sebelum terjadi kompresi. Ada dua alasan, yaitu :

1) Tanda tangan dibangkitkan sebelum kompresi generatif.
2) Enkripsi pesan ditentukan setelah kompresi untuk memperkuat keamanan kriptografi karena pesan yang dikompres memeiliki redundansi disbanding teks-asli sehingga akan mempersulit analisis kode oleh para kriptanalisis.

GPG menggunakan teknik konversi pengkodean radix-64. Teknik ini merupakan pemetaan untuk mengubah masukan numerik ke bentuk karakter sebagai keluarannya. Bentuk dari pengkodean relevan dengan sifat-sifat sebagai berikut :

1) Cakupan dari fungsi karakter merupakan sesuatu yang mudah dan dapat dimengerti dan mudah untuk ditempatkan. Bukan hanya penyandian dalam biner yang bisa dilakukan tetapi juga bisa untuk penyandian dalam bentuk apapun tergantung dari sistem yang digunakan.
2) Jumlah karakter yang terdiri dari 65 karakter salah satunya digunakan untuk “pad” dengan $2^6 = 64$ yang tersedia dan setiap karakter dapat ditempatkan dengan masukan 6 bit.
3) Tidak ada control terhadap karakter yang masuk ke dalam tahap pengaturan. Pesan yang dikodekan, yang menggunakan radix-64,

bisa melewati sistem mail-handing yang akan men-scan data untuk mengontrol karakternya.

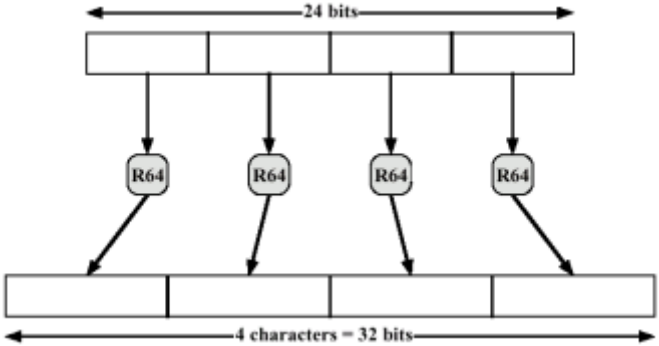
4) Karakter dengan tanda "-" tidak digunakan karena karakter ini penting sekali untuk format RFC 822 dan sebaiknya dihindari.

Tabel 3.2 Pengkodean Radix Base 64

Nilai 16 Bit	Karakter Pengkodean	Nilai 16 Bit	Karakter Pengkodean	Nilai 16 Bit	Karakter Pengkodean
0	A	22	W	44	S
1	B	23	X	45	T
2	C	24	Y	4	U
3	D	25	Z	47	V
4	E	26	A	48	W
5	F	27	B	49	X
6	G	28	C	50	Y
7	H	29	D	51	Z
8	I	30	E	52	0
9	J	31	F	53	1
10	K	32	G	54	2
11	L	33	H	55	3
12	M	34	I	56	4
13	N	35	J	57	5
14	O	36	K	58	6
15	P	37	L	59	7
16	Q	38	M	60	8
17	R	39	N	61	9
18	S	40	O	62	+

19	T	41	P	63	/
20	U	42	Q	(pad)	=
21	V	43	R		

Pada tabel tersebut, pemetaan masukan 6 bit yang mempunyai nilai karakter dan terdapat karakter alfanumerik, “+”, “/” dan”=”. Karakter ini digunakan sebagai tambahan karakter.



Gambar 3.6 Pengkodean Data Binari ke Format Radix-64

Pada gambar di atas proses masukan yang terdiri dari blok-blok 3 oktet atau sama dengan 24 bit, 6 bit dari setiap blok akan dipetakan ke bentuk karakter dan setiap karakter terdiri dari 8 bit. Pada gambar di atas, masukan terdiri dari 24 bit dan keluaran menjadi 32 bit.

Contoh :

Proses encoding dari grup masukan 8 bit ke keluaran string karakter radix-64 aphabet :

1. Input Text : 0x 15 d0 2f 9e b7 4c

8 bit octet : 00010101 11010000 00101111 10011110 10110111 01001100

6 bit indeks : 000101 011101 000000 101111 100111 101011 011101 001100

Desimal : 5 29 0 47 39 43 29 12

Karakter keluaran (radix-64 encoding) : F d A v n r d M

ASCII format (0x : 46 64 41 76 6e 72 64 4d

Binari : 01000110 01100100 01000001 01110110 01101110 01110010 01100100 01001101

2. Input text : 0x 15 d0 2f 9e 43 29 12

8 bit octet : 00010110 11010000 00101111 10011110 10110111

6 bit indeks : 000101 011101 000000 10111 100111 101011 011100

Ditambah dengan 00 (=)

Desimal : 5 29 0 47 39 43 28

Karakter keluaran : F d A v n r c =

3. Masukkan raw text : 0x 15 d0 2f 9e

8 bit octet : 00010101 11010000 00101111 10011110

6 bit indeks : 000101 011101 000000 101111 100111 100000

Ditambah dengan 0000 (==)

Decimal : 5 29 0 47 39 32

Karakter keluaran : F d A v n g = =

3.4.5 Kompatibilitas Email

Untuk memepermudah penggunaan aplikasi email, pesan yang terenkripsi dapat dikonversi ke dalam string ASCII menggunakan konversi radix (base) 64. Pada GPG, blok-blok yang sudah dikonversi dienkripsi. Jika hanya tanda tangan digital yang dikirim maka intisari pesan dienkripsi menggunakan kunci rahasia, dan jika layanan kompatibilitas digunakan tanda tangan dan pesan akan dienkripsi dengan kunci simetri dan setiap blok berisi 8 bit.

Bagaimanapun sistem email elektronik hanya menggunakan blok yang berisi kode ASCII, untuk mengakomodasi batasan ini GPG mengkonversi teks ke bentuk aliran biner 8-bit ke bentuk kode ASCII yang dapat dicetak. Oleh karena itu teknik yang digunakan adalah konversi radix base 4 bit, dari setiap blok yang terdiri dari tiga octet base 24 bit yang akan dipetakan menjadi empat karakter ASCII base 32 bit, dan format ini juga menambahkan CRC untuk mendeteksi kesalahan dalam transmisi.

Penggunaan radix base 64 bit akan menambah ukuran pesan sebanyak 33,33%. 24 bit dikonversi menjadi 32 bit ASCII, tetapi kunci sesi dan bagian dari tanda tangan pesan berlagsung secara bersamaan, apalagi pesan sudah dikompres.

3.5 Manajemen Kunci Pada Gnu Privacy Guard

Enkripsi kunci umum adalah inti dari GPG. Hal ini digunakan untuk dua tujuan :

1) Pengirim menggunakan kunci rahasianya untuk mengganti *digital signature* dalam pesan yang keluar.
2) Dan pengirim menggunakan *kunci umum* dari penerima untuk melakukan enkripsi kunci sesi (*session key*) yang bersifat rahasia.

3.6 Algoritma Pada Gnu Privacy Guard

Gnu Privacy Guard mendukung berbagai jenis algoritma yang dapat dipilih dan digunakan. Hal ini sesuai prinsip GPG, yaitu pengguna dapat menggunakan algoritma kriptografi tertentu yang dianggap aman. Beberapa algoritma diantaranya :

- Algoritma kunci-publik (RSA, RSA-E, RSA-S, ELG-E, DSA) : Algoritma kunci umum memiliki dua jenis kunci, yaitu kunci kunci umum dan kunci rahasia. Algoritma ini biasanya lambat dan juga rentan terhadap beberapa serangan kriptanalis tertentu. Oleh karena itu algoritma ini hanya digunakan untuk mengenkripsi bagian tertentu data, seperti kunci sesi (session key) dari blok kode atau untuk mengenkripsi nilai Hash tertentu. Algoritma kunci umum biasa digunakan untuk melakukan enkripsi dan tanda tangan digital.
- Algoritma simetrik (3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH, CAMELLIA128, CAMELLIA192, CAMELLIA256) : algoritma enkripsi simetri melakukan enkripsi secara normal di mana kunci yang sama digunakan untuk melakukan enkripsi dan dekripsi.
- Algoritma Fungsi Hash (MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224) : Fungsi Hash menerima masukan berupa panjang seluruh dokumen atau pesan untuk kemudian menghasilkan nilai Hash atau disebut juga fingerprint (sidik jari) dari dokumen dengan ukuran tertentu, biasanya 1 atau 20 byte. Sidik jari memiliki dua atribut penting, yaitu seseorang tidak akan dapat menemukan dokumen hanya dengan melihat sidik jari dokumen tersebut dan pada kenyataannya tidak ada dua dokumen yang memilki sidik jari yang sama.

Selain algoritma kriptorafi di atas, GPG juga mendukung beberapa algoritma kompresi, antara lain : ZIP, ZLIB, BZIP2.

Program enkripsi GnuPG (GNU Privacy Guard) melayani Anda dengan aman, sederhana dan bebas metode enkripsi email. Hal ini dapat digunakan secara pribadi atau komersial tanpa pembatasan. Teknologi enkripsi yang digunakan oleh GnuPG sangat aman dan tidak dapat rusak dengan menggunakan teknologi terkini.

GnuPG adalah perangkat lunak bebas. Ini berarti bahwa setiap orang dapat menggunakan perangkat lunak untuk pribadi atau untuk tujuan komersial, serta menganalisis atau mengubah source code (misal. perintah pemrograman yang sebenarnya), dan mendistribusikan sama. Transparansi kode sumber membentuk bagian penting dari perangkat lunak keamanan, karena merupakan satu-satunya cara untuk memverifikasi program yang dapat dipercaya. GnuPG didasarkan pada standar internasional OpenPGP (RFC 2440), adalah sepenuhnya kompatibel dengan PGP dan menggunakan infrastruktur yang sama (server kunci dll). Sejak GnuPG versi 2 yang kriptografi standar S/MIME (CMS/RFC3852 dan X.509) juga didukung.

3.6.1 Enkripsi dan Dekripsi RSA

Diberikan teks-asli : TAKEHOMETES

Teks-asli	Heksadesimal	Biner
T	54	01010100
A	41	01000001
K	4B	01001011
E	45	01000101
H	48	01001000
O	4F	01001111
M	4D	01001101

E	45	01000101
T	54	01010100
E	45	01000101
S	53	01010011

Teks-asli biner :
01010100010000010100101101000101010010000100111101001101
01000101010101000100010101010011

Teks-asli biner dibagi dalam 11 bit per blok sehingga diperoleh 8 blok bit-asli beserta interpretasi desimalnya :

Teks-asli	Desimal	1024	512	256	128	64	32	16	8	4	2	1
Blok 1	674	0	1	0	1	0	1	0	0	0	1	0
Blok 2	82	0	0	0	0	1	0	1	0	0	1	0
Blok 3	1674	1	1	0	1	0	0	0	1	0	1	0
Blok 4	1156	1	0	0	1	0	0	0	0	1	0	0
Blok 5	1958	1	1	1	1	0	1	0	0	1	1	0
Blok 6	1301	1	0	1	0	0	0	1	0	1	0	1
Blok 7	648	0	1	0	1	0	0	0	1	0	0	0
Blok 8	1363	1	0	1	0	1	0	1	0	0	1	1

Enkripsi RSA :

User A akan Mengirim User B pesan :

Teks-asli : TAKEHOMETES

User A memilih p = 41, q = 53 dan b =623

Rumus enkripsi RSA :

$Y = E(X) = \text{Teks-kode\_Desimal} = (X)^b \bmod n$

Di mana :

Y : Teks-kode_Desimal

X : Teks-asli_Desimal

b : Kunci umum User A = 623

$n = p \times q$ = 2173

$\Phi(n) = (p-1) \times (q-1)$ = 2080

$\text{Gcd}(\Phi(n), b) = \text{gcd}(2080,623)$ = 1

Blok	X	Y = E(X)	Teks-kode_Biner (11 bit)
1	674	1650	11001110010
2	82	1230	10011001110
3	1674	1655	11001110111
4	1156	143	00010001111
5	1958	353	00101100001
6	1301	1495	10111010111
7	648	349	00101011101
8	1363	590	01001001110

Teks-kode biner :

11001110010100110011101100111011100010001111001011000011

01110101110010101110101001001110

Teks-kode_biner 8 bit	Heksadesimal	Karakter
1100 1110	CE	
0101 0011	53	S
0011 1011	3B	;
0011 1011	3B	;
1000 1000	88	Ê
11110010	F2	≥
1100 0011	C3	
0111 0101	75	U
1110 1010	EA	Ω
0100 1110	4E	N

Menentukan eksponen dekripsi rahasia (a)

a.b = k.mod Φ(n) + 1

a.623 = k.mod 2080 + 1 atau b-1 = 207 mod 2080

gcd (2080, 207) = 1

jadi eksponen dekripsi rahasia User A = 207

Dekripsi RSA :

Rumus dekripsi RSA :

$X = D(Y) = \text{Teks-asli\_desimal} = (Y)^a \bmod n$

Di mana :

X : Teks-asli_Desimal

Y : Teks-kode_Desimal

a : Kunci rahasia User A = 207

n = p x q = 2173

Φ(n) = (p-1) x (q-1) = 2080

gcd(Φ(n), a) = gcd(2080,201) = 1

Blok	Y	X = D(Y)	Teks-asli_Biner (11 bit)
1	1650	674	01010100010
2	1230	82	00001010010
3	1655	174	11010001010
4	143	1156	10010000100
5	353	1958	11110100110
6	1495	1301	10100010101
7	349	648	01010001000
8	590	1363	10101010011

Teks-asli biner :

01010100010000010100101101000101010010000100111101001101
01000101010101000100010101010011

Teks-asli biner 8 bit	Heksadesimal	Karakter
0101 0100	54	T
0100 0001	41	A
0100 1011	4B	K
0100 1011	45	E
0100 1000	48	H
0100 1111	4F	O
0100 1101	4D	M
0100 0101	45	E
0101 0100	54	T
0100 0101	45	E
0101 0011	53	S

3.6.2 Enkripsi dan Dekripsi ElGamal

Enkripsi ElGamal diasumsikan sebagai berikut :

p = 11 (bilangan prima)

q = 4 (bilangan acak g < p)

x = 8 (kunci rahasia x < p)

kemudia dihitung :

$y = g^x \pmod p \equiv 4^8 \pmod{11} = 9$

kunci public adalah y = 9, q = 4 dan p = 11. Kunci rahasia x = 8. Untuk mengenkripsi pesan m = 5, pertama pilih bilangan acak k = 7, gcd (k,p-1) = gcd (7,10)=1 dan perhitungannya :

$r \equiv g^k \pmod p \equiv 4^7 \pmod{11} \equiv 5$

$s \equiv (y^k \pmod p)) (m(\text{mod } p-1))$

$\equiv (9^7 \pmod{11})) (5(\text{mod } 10) \equiv 4 \text{ x } 5 \equiv 20$

Untuk mendekripsi pesan m, pertama dihitung :

$rx \pmod p \equiv 58 \pmod{11} \equiv 4$

dan rasionya :

$m \equiv s/rx \pmod p \equiv 20/4 \equiv 5$

pesan m sudah selesei didekripsi dengan menggunakan algoritma enkripsi ElGamal.

3.7 Kelemahan Gnu Privacy Guard

Berikut ini adalah beberapa kelemahan yang pernah ditemukan pada GPG :

1. Masalah pesan ganda

   Pada awal tahun 2007, Gerardo Richarte, dari Core Security Technologies, menemukan masalah ketika menggunakan GnuPG pada

mode *streaming*. Karena adanya *bug* tersebut, maka penyerang bisa menyisipkan teks tambahan sebelum atau sesudah tanda tangan digital. Dengan demikian, pengguna akan mengira bahwa teks tambahan tersebut termasuk tanda tangan digital. *Bug* ini telah diperbaiki dengan mengubah GnuPG sehingga pesan OpenPGP palsu dapat terdeteksi dan verifikasi akan gagal. Perbaikan ini diikutsertakan pada GnuPG 1.4.7.

2. Pointer fungsi yang bisa diatur

   Pada tahun 2006, Tavis Ormandy, dari tim keamanan Gentoo, menemukan *bug* yang bisa dieksploitasi pada pemrosesan paket yang terenkripsi pada GnuPG. Dengan memanfaatkan paket OpenPGP yang cacat, penyerang bisa mengubah referensi dari pointer fungsi pada GnuPG. Hal ini merupakan *bug* yang sangat kecil untuk dieksploitasi. *Bug* ini mempengaruhi penggunaan GnuPG di mana penyerang bisa mengatur data yang diproses GnuPG. Hal ini tidak terbatas pada data yang terenkripsi, tetapi data yang ditandatangani pun bisa dipengaruhi. *Bug* ini telah diperbaiki dengan dirilisnya GnuPG 1.4.6 dan *patch* untuk GnuPG 2.0.1.

3. Buffer Offerflow

   Bug ini ditemukan pada GnuPG 1.4 dan 2.0. Saat menjalankan GnuPG secara interaktif, pesan-pesan tertentu bisa digunakan untuk membuat GPG mengalami *crash*. Sejak versi 1.4.7 dan 2.0.3, bug ini telah diperbaiki.

4. Verifikasi tanda tangan False Positive

   Pada awal tahun 2006, Gentoo project menemukan bug pada GnuPG, yaitu mungkin terjadinya verifikasi tanda tangan digital yang *false positive*. Tanda tangan digital yang seharusnya benar, ternyata gagal diverifikasi. Verifikasi yang *false positive* ini hanya terjadi pada tanda tangan digital yang tidak dilekatkan pada pesan, yang dilakukan oleh

script atau program email. Penggunaan GnuPG dengan cara interaktif tidak terpengaruh oleh *bug* ini. *Bug* ini telah diperbaiki pada GnuPG 1.4.2.1.

5. GnuPG tidak mendeteksi injeksi data tidak bertandatangan

   Bug ini muncul akibat kesalah saat memperbaiki *bug* verifikasi *false positive* sebelumnya. Karena *bug* ini, verifikasi tanda tangan yang dilekatkan pada pesan bisa memberikan hasil yang positif, tapi saat saat mengekstrak datayang tidak bertandatangan, data tersebut mungkin ditambahi data tambahan di luar tanda tangan digital. Karena itu, hal ini memungkinkan penyerang untuk mengambil pesan yang bertandatangan, lalu memasukkan data tambahan yang sewenang-wenang. *Bug* ini telah diperbaiki pada GnuPG 1.4.2.2.

6. Kelemahan kunci ElGamal ditemukan

   Pada tahun 2003, Phong Nguyen menemukan *bug* pada cara GnuPG membuat dan menggunakan kunci ElGamal untuk tanda tangan digital. Hal ini merupakan kegagalan keamanan yang signifikan yang bisa menyebabkan hampir semua kunci ElGamal yang digunakan untuk menandatangani bisa dipecahkan dalam hitungan detik. Semua kunci ElGamal untuk tanda tangan dan enkripsi mungkin untuk dipecahkan. Solusi yang diberikan GnuPG adalah untuk tidak menggunakan kunci ElGamal untuk tanda tangan + enkripsi. Pada GnuPG versi selanjutnya (setelah GnuPG 1.0.2), kemampuan GnuPG untuk membuat kunci tanda tangan digital+enkripsi ElGamal dihilangkan.

3.8 Perintah-Perintah Dasar Gnu Privacy Guard Berbasis Windows

Tabel 3.3 Perintah Dasar GPG Berbasis Windows

COMMAND	KETERANGAN
-s, --sign [file]	Membuat sebuah tanda tangan
--clearsign [file]	Membersihkan/menghapus tanda tangan
-b, --detach-sign	Membuat tanda tangan yang objektif/tidak memihak
-e, --encrypt	Enkripsi data
-c, --symmetric	Enkripsi hanya dengan cipher simetrik
-d, --decrypt	Dekripsi data (default)
--verify	Memverifikasi tanda tangan
--list-keys	Melihat daftar kunci
--list-sigs	Melihat daftar kunci dan tanda tangan
--check-sigs	Melihat dan memeriksa tanda tangan kunci
--fingerprint	Melihat daftar kunci dan fingerprint (sidik jari)
-K, --list-secret-keys	Melihat daftar kunci private/pribadi
--gen-key	Membuat pasangan kunci baru
--delete-key	Menghapus kunci public
--delete-secret-key	Menghapus kunci pribadi
--sign-key	Menandatangani kunci
--lsign-key	Menandatangani kunci lokal
--edit-key	Merubah kunci
--gen-revoke	Membatalkan sertifikat kunci
--export	Mengekspor kunci
--send-key	Mengekspor kunci ke server kunci
--recv-key	Mengimpor kunci dari server kunci
--search-key	Mencari kunci pada server kunci
--refresh-key	Mengupdate semua kunci dari server kunci

--import-key	Mengimpor/menggabungkan kunci
--card-status	Mencetak status kartu
--card-edit	Mengubah data pada kartu
--change-pin	Mengubah PIN kartu
--update-trustdb	Mengupdate database
--print-md algo [files]	Mencetak Message Digests
Options :	
-a, --armor	Membuat keluaran berekstensi ascii
-r, --recipient NAME	Enkripsi menggunakan User ID penerima
-u, --local-user	Menggunakan user-id lokal untuk tanda tangan dan dekripsi
-z N	Menset/menyetel kompres level N (0 disables)
--textmode	Menggunakan mode teks resmi
-o, --output	Digunakan sebagai keluaran(output) file
-v, --verbose	Verbose
-n, --dry-run	Jangan membuat banyak perubahan
-i, --interactive	Perintah sebelum menulis lebih
--openpgp	Penggunaan perilaku openpgp
--pgp2	Membangkitkan PGP 2.x untuk kompabilitas pesan

3.9 Perbandingan GPG dengan PGP

Awalnya Phill Zimmermann menulis PGP dibawah lisensi GPL (GNU Public Lisence) sebagai freeware hak cipata. Namun, karena masalah royalti paten dan biaya pembelaan hukum yang berkaitan dengan hukum ekspor Amerika Serikat, Ia membuat upgrade untuk menjadi sebuah program berpemilik, hak-hak yang telah diperdagangkan di sekitar. PGP Corporation sekarang memiliki hak untuk PGP (kecuali untuk baris

perintah versi, yang masih dimiliki oleh Network Associates, Inc. Beberapa versi PGP masih menggunakan algoritma enkripsi IDEA, yang masih dipatenkan di beberapa negara. Gnu Privacy Guard adalah re-implementasi dari PGP dengan kode yang dirilis di bawah GNU Public License, dan tidak menggunakan algoritma enkripsi IDEA, sehingga dapat benar-benar bebas. Algoritma dan format data yang digunakan dalam PGP, GPG, dan program yang kompatibel publik didokumentasikan oleh Aliansi OpenPGP. PGP versi terbaru dapat menggunakan NIST baru Advanced Encryption Standard (AES) di tempat IDEA, dan karena itu dapat beroperasi dengan GPG. AES lebih aman daripada IDEA, biaya rendah karena tidak dipatenkan dan karena itu bebas royalti. Ternyata Gnu Privacy Guard lebih kompatibel dengan standar OpenPGP daripada yang asli adalah Pretty Good Privacy. GnuPG tidak berbeda jauh dengan dengan PGP karena GnuPG merupakan re-implementasi dari PGP itu sendiri. Berikut ini beberapa perbandingan Gnu Privacy Guard dengan Pretty Good Privacy berbasis Windows :

Tabel 3.4 Perbandingan GPG dengan PGP berbasis Windows

	Gnu Privacy Guard	**Pretty Good Privacy**
GUI	√	√
Command Line	√	√
Software	Gratis, Full	Gratis, Trial
Lisensi	GPL (GNU Public Lisence)	PGP Corporation
Dukungan WebMail	Gmail, Yahoo, dll	Belum ada
Flatform	Crossflatform	Crossflatform
Dukungan Sertifikat	OpenPGP, X.509	OpenPGP, Web of Trust
Konfigurasi	Mudah	Mudah

BAB IV
PENGUJIAN GNU PRIVACY GUARD

4.1 Lingkungan pengujian

4.1.1 Perangkat Keras

- 1 buah PC/Laptop (dianalogikan menjadi 2 user).
- Internet Connection.

4.1.2 Perangkat Lunak

Berikut ini adalah software yang digunakan dalam pengujian Gnu Privacy Guard :

- Operating System

  Sistem operasi yang digunakan adalah Windows XP Service Pack 2 (SP2).
- GnuPG Software

  Software GnuPG yang digunakan adalah gnupg-w32cli-1.4.10b.
- Browser

  Browser yang digunakan adalah Firefox Mozilla versi 3.5.7 serta add-ons firegpg.
- Web Mail

  Web Mail yang digunakan Gmail. Karena Gmail telah mendukung penuh penggunaan GnuPG dalam pengriman pesan email.

4.2 Skenario Pengujian

Pengujian dilakukan dengan scenario sebagai berikut :

1. User A dan User B membuat atau membangkitkan pasangan kunci. Berupa kunci privat dan kunci publik.

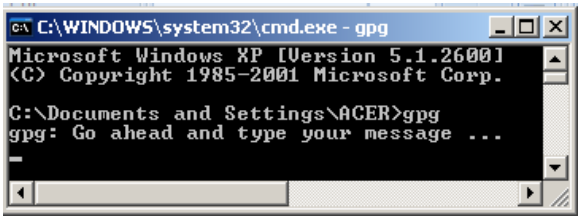
2. Masing-masing User memback-up kunci privat dan mengeksport kunci publik yang akan digunakan dalam pertukaran kunci ke rekan yang dituju.
3. Untuk mendapatkan kunci publik dari user lain. Masing-masing user melakukan impor kunci yang bisa dilakukan melalui email.
4. Pengirim email (User A) mengirim email menggunakan kunci publik si penerima (User B), ini diperlukan agar email bisa dienkripsi oleh pengirim dan didekripsi oleh si penerima email sebagai proses otentikasi.
5. Penerima email (User B) mendekrip isi email menggunakan kunci private yang dimiliki.

4.3 Membuat Pasangan Kunci (Key Pair)

Untuk melindungi email yang akan dikirimkan maka hal yang harus dilakukan oleh pengguna email adalah membuat sepasang kunci, private key dan public key. Langkah – langkah untuk membuat pasangan kunci GPG berbasis Widows dapat dilakukan sesuai dengan perintah sebagai berikut:

1. Klik **Start Menu → Run → cmd**

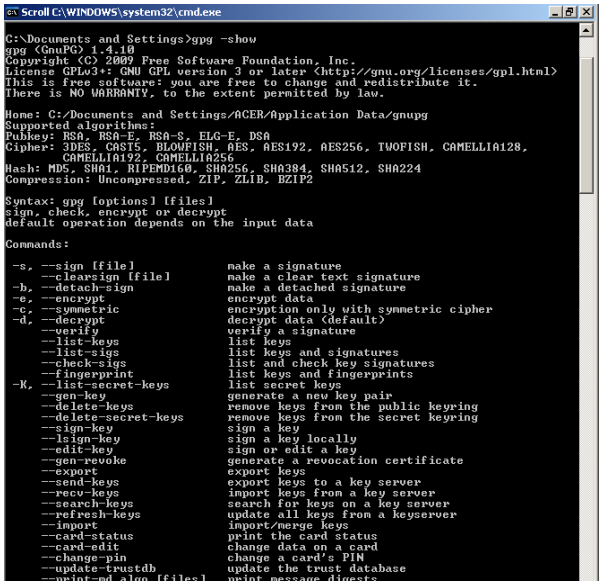
   Untuk menguji gpg sudah ada di dalam sistem windows.,ketik “gpg” pada command prompt, maka akan terlihat seperti gambar di bawah ini :



Gambar 4.1 Tes GPG pada Windows

Pada gambar di atas melihatkan bahwa GPG suda terpasang dan dapat digunakan pada sistem Windows.

2. Untuk melihat menu perintah/options GPG pada Windows ketik perintah gpg –show pada command prompt, maka akan terlihat seperti gambar seperti ini dibawah ini :



Gambar 4.2 Menu GPG Pada Windows

Pada gambar di atas melihatkan versi GPG (GnuPG) yang digunakan yaitu versi 1.4.10. lisesnsi yang digunakan GPLv3+, yaitu lisensi GNU GPL versi 3 ke atas. Pada gambar juga melihatkan algoritma-algoritma yang didukung seperti algoritma PubKey (*Assymmetric)*, Cipher (*Symmetric)*, Hash, dan algoritma kompresi.

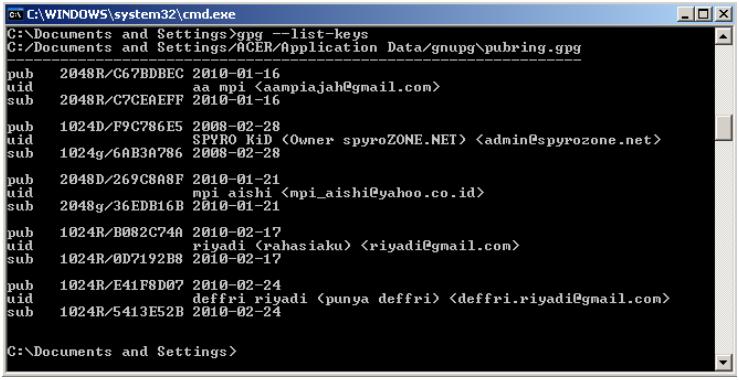
3. Untuk membuat pasangan kunci, kerik perintah “gpg --gen-key” pada command prompt :

Gambar 4.3 Proses Pembangkitan Kunci

Pada gambar di atas menjelaskan bagaimana langkah-langkah membangkitkan pasangan kunci, setelah mengetik perintah “gpg --gen-key” selanjutnya anda diminta memilih algoritma yang dipakai dalam pembangkitan kunci, selanjutnya memasukkan panjang panjang bit kunci antara 1024 bit sampai 4096 bit, semakin panjang bit kunci yang dipakai semakin aman dari serangan crytanalist. Kemudian pilih

berapa lama waktu berlaku kunci yang akan dibuat. Kemudian isikan User ID yang berupa nama, alamat email, komentar. Kemudian masukkan passphrase/password dua kali. Passphrase digunakan untuk melindungi kunci pribadi. Setelah selesai memasukkan passphrase tunggu beberapa saat proses pe-random-an atau pembangkitan kunci.

4. Melihat Pasangan Kunci. Untuk melihat pasangan kunci yang telah di buat gunakan perintah “gpg --list-keys” pada command prompt :



Gambar 4.4 Daftar Kunci

Pada gambar di atas terlihat pasangan kunci yang telah dibuat.

4.4 Pertukaran Kunci

Pertukaran kunci berupa Ekspor dan Impor kunci diperlukan untuk melakukan pertukaran kunci publik dengan rekan anda sehingga anda dan rekan anda dapat saling mengirim file secara aman.

4.4.1 Ekspor Kunci Publik

Ekspor kunci diperlukan bila anda menginginkan menyimpan sebagian kunci (kunci publik) untuk anda kirimkan ke rekan anda agar rekan anda tersebut dapat mengirimkan file terenkripsi ke anda, atau anda juga dapat menyimpan keseluruhan kunci (publik dan privat) kedalam file tertentu dengan ekstensi txt ataupun asc (file ascii).

Berikut adalah perintah untuk mengekspor kunci pada command prompt :

C:\Documents and setting>gpg --armor --export deffri.riyadi@gmail.com > usera.txt

Public key **usera.txt** :

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.10 (MingW32)

mI0ES4R89QEEALIjmGaC3FO78Zqu5fR5rI9G4IY0tHMXreUOPGSHTTS2ZQDiX2ud
XJYZza7MaYkAgut7tMPaBNN3YB7679c9kqHKuObyu5dZytCeB8qvU+KzzL4jg4SK
Wb/n0Yak5Oz7TeRXuxY++xc3iDk7zJPJJ9TeXwvXfLoX3A49NyXowiz9ABEBAAG0
NmRlZmZyaSByaXlhZGkgKHB1bnlhIGRlZmZyaSkgPGRlZmZyaS5yaXlhZGlAZ21h
aWwuY29tPoi4BBMBAgAiBQJLhHz1AhsDBgsJCAcDAgYVCAIJCgsEFgIDAQIeAQI
X
gAAKCRAC3Bf15B+NB7SIA/9rQKZzt/ZRIId9/XwlWysGR7UaZHnOJcG44yodsaUY
hSdNZmja/Ysc0HCVpzYltpSanbJfGNZU5IQyHB2db3fgSxB9tgaUep7Fr4hCREFA
3GqLPIQ8bxgpEDw18BEP0cMRS7VpiXtEulLFEFWf6p9QRNK4Tw9zdqeBnK1IUqYh
tLiNBEuEfPUBBADD9fg/QvJpWndABo384kXNMayaO6v1oiLovc2mpajvkz44Cijq
YniQlPXIAGnpyidEHnaaytP3GV9QBn4QkBvaZDD7lT2bxBQtS3PO8qav+ralPRGd
arDL3F9VCNQgrfwOPKkV8cNSiHlnjhrByWDqITT1JnTctWFrCzg+R+hXkwARAQAB
iJ8EGAECAAkFAkuEfPUCGwwACgkQAtwX9eQfjQcrhwP+LBxUUbbnm30YE3qpljU
A
0Ls5/4GFL7cXIsKLehZA5QPugzB0JCrIKYqMkk3prrVewbA4usDfi6l2kE7acED+

Hmds9prWnYDvVpRna1QBteQbPM8DUnC7F5/G7iEL38m9dDN5hc8Ov0CtTgNLW7tu
oLfw+Kgtq2vwQCjpE2mELvc=
=EiM6
-----END PGP PUBLIC KEY BLOCK-----

Baris pertama dan baris terakhir merupakan batas awal dan akhir pesan GPG yang dalam contoh diatas adalah kunci publik.

Pada baris kedua dari file tersebut terdapat versi GPG yang digunakan yaitu GnuPG versi 1.4.10.

Pada bagian tengah terdapat kunci publik yang telah dikonversi ke ascii, supaya dapat dimasukkan ke email tanpa harus dimasukkan ke dalam file attachment. Kunci publik ini aslinya adalah bilangan biner yang diperoleh dari pembangkit bilangan acak.

4.4.2 Impor Kunci Publik

Impor kunci diperlukan bila anda ingin memasukkan kunci publik rekan anda ke dalam Sistem GnuPG, sehingga pada kesempatan berikutnya anda dapat mengirim file kepada rekan anda secara aman. Tanpa memasukkan kunci publik ke GnuPG anda, anda tidak dapat mengamankan file yang anda kirim dengan GPG.

Impor kunci privat akan anda butuhkan pada saat anda menggunakan public key rekan anda untuk mengirim email yang terenkripsi.

Berikut adalah perintah untuk mengekspor kunci pada command prompt dengan perintah :

gpg –import nama public key

Gambar 4.5 Impor Kunci

Pada gambar di atas terlihat proses impor kunci public key milik SPYRO KiD berhasil di import kedalam kunci.

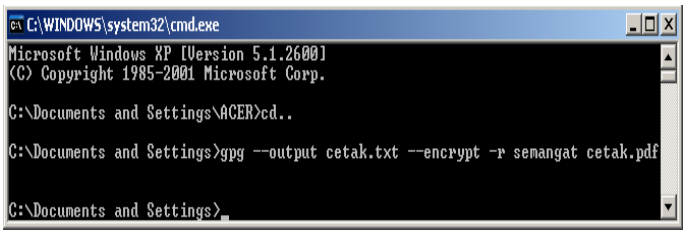
4.5 Enkripsi dan Dekripsi Menggunakan GnuPG Berbasis Windows

Pada bagian ini akan dijelaskan cara-cara melakukan enkripsi, dekripsi file menggunakan Gnu Privacy Guard.

4.6.1 Enkripsi

Enkripsi yaitu mengubah *plain-text*(teks-asli) menjadi *cipher text*. Perintah untuk menenkripsi suatu file pada Gnu Privacy Guard melalui command prompt sebagai berikut :

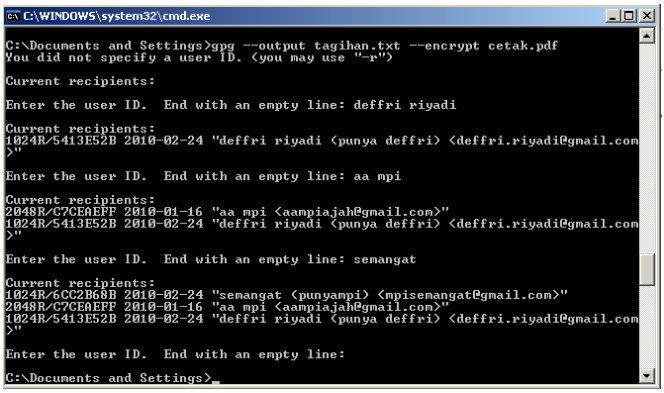
gpg --output namafilekeluaran --encrypt –r UserIDpenerima namafileasli



Gambar 4.6 Enkripsi File ke Satu Penerima Pesan

Atau :

gpg --output namafilekeluaran --encrypt namafileasli



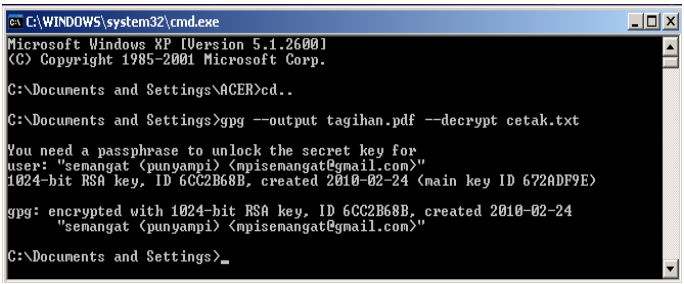
Gambar 4.7 Enkripsi File ke Banyak Penerima Pesan

Pada gambar 4.6 dan 4.7 di atas terlihat perintah Gnu Privacy Guard untuk mengenkripsi sebuah file ke satu user id penerima pesan dan banyak user id penerima pesan. File di enkripsi memakai kunci publik penerima pesan/email.

4.6.2 Dekripsi

Dekripsi kebalikan dari enkripsi, yaitu mengubah *ciphertext* menjadi *plaintext*. Untuk mendekripsi suatu pesan pada Gnu Privacy Guard menggunakan perintah sebagai berikut :

gpg --output namafilekeularan --decrypt namafileyangterenkripsi



Gambar 4.8 Dekripsi File

Pada gambar di atas terlihat perintah untuk mendekrip sebuah file. Untuk membuka file diperlukan passphrase kunci pribadi milik penerima pesan. Pada gambar terlihat passphrase yang digunakan untuk mendekripsi file milik UserID “semangat” yang menggunakan kunci RSA 1024 bit.

4.6 Pengujian Gnu Privacy Guard dalam Pengiriman Email

Sebelum melakukan pengiriman email yang terenkripsi pastikan PC/laptop pengirim dan penerima email sudah terpasang aplikasi gnupg yang diperlukan.

Jika belum terpasang download GPG untuk windows. Install juga firegpp di http://www.getfirgpg.org pilih menu install dan download. Pastikan juga browser yang dipakai menggunakan Mozilla Firefox versi 3 ke atas agar firegpg dapat berjalan di browser Mozilla user . setelah proses instalasi selesai restart browser.

4.6.1 FireGPG

FireGPG adalah ekstensi Firefox di bawah MPL (*Mozilla Public Lisence)* yang menyediakan interface yang terintegrasi untuk menerapkan

operasi GnuPG ke teks dari halaman web manapun, termasuk enkripsi, dekripsi, menandatangani, dan verifikasi tanda tangan digital.

FireGPG menambahkan beberapa fitur ke interface Gmail untuk membiarkan anda menggunakan fitur GPG langsung di webmail Anda. Lebih banyak aplikasi webmail yang mungkin akan didukung di masa depan. FireGPG adalah OpenPGP / mime compliant.

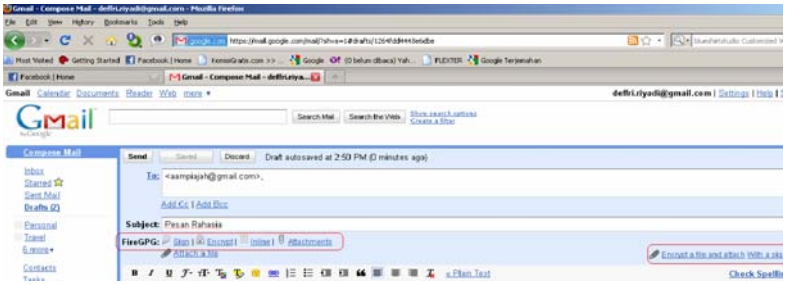
Macam-macam webmail yang didukung oleh FireGPG :

1. Gmail (Old Gmail Interface dan New Gmail Interface)
2. Yahoo
3. Roundcube
4. SquirrelMail
5. Horde (Imp/Dimp)
6. Dan lain-lain

Untuk sekarang ini Gmail yang mendukung penuh penggunaan GPG dengan dukungan OpenPGP didalamnya.

FireGPG mampu mendeteksi PGP blok di halaman apapun (misalnya sebuah kunci publik), dan memungkinkan Anda dengan mudah mengelola blok yang berbeda ini.

FireGPG memiliki sebuah API yang memungkinkan Anda untuk merancang sebuah website yang menggunakan fitur GPG pada klien misalnya untuk pengguna autheticate untuk administrasi panel. FireGPG saat ini diterjemahkan ke dalam berbagai bahasa.

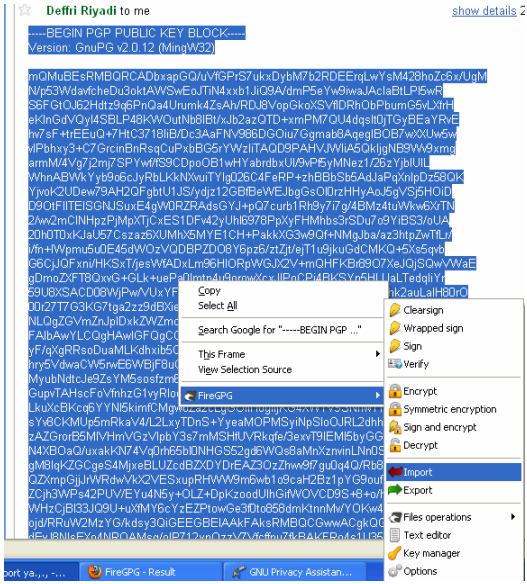


.Gambar 4. 9 Tampilan webmail Gmail yang sudah terpasang FireGPG

Pada gambar diatas terdapat Menu **FireGPG** dibawah **Subject** yang terdiri dari **Sign, Encrypt, Inline dan Attachment** serta tambahan menu **Encrypt a file and attach** dan **With a signature** yang berada di sampaing kanan menu **FireGPG.**

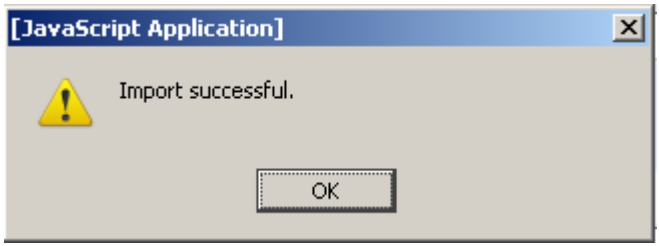
4.6.2 Import Public Key pada webmail

Import Public Key pemilik alamat email tujuan. Hal ini dibutuhkan supaya nantinya tanda tangan rekan Anda dapat dikenali. Untuk dapat melakukan Enkripsi terhadap pesan dan Attachment nantinya, Anda harus memiliki Public Key rekan Anda. Begitu pula sebaliknya, rekan Anda juga harus memiliki Public Key Anda. Ada banyak cara meng-import suatu Public Key. Anda dapat langsung meng-import Public Key tersebut langsung melalui Browser Mozilla Firefox Anda. Blok public Key target, Klik kanan, pilih menu *FireGPG* lalu pilih *Import.*



Gambar 4.10 Proses import publik key di email/browser

Jika berhasil akan muncul kotak dialog :



Gambar 4.11 Proses import kunci berhasil

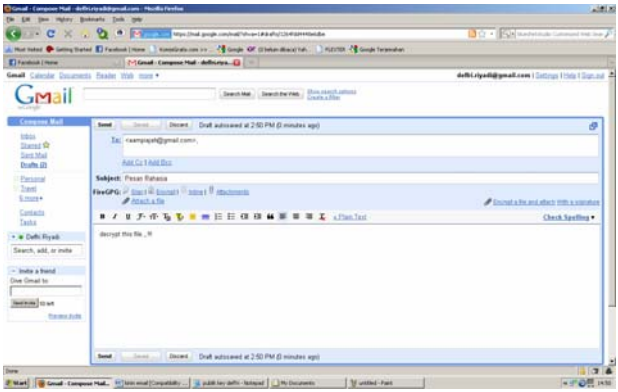
4.6.3 Pengiriman dan Penerimaan Email

Setelah import key berhasil dijalankan, baik pengirim dan penerima email, langkah selanjutnya bagaimana mengirim sebuah email yang terenkripsi dengan aman. Pengiriman email dapat dilakukan memakai

webmail apa saja yang sudah disupport oleh GPG, dalam hal ini penulis memakai Gmail sebagai pengiriman email menggunakan Gnu Privacy Guard.

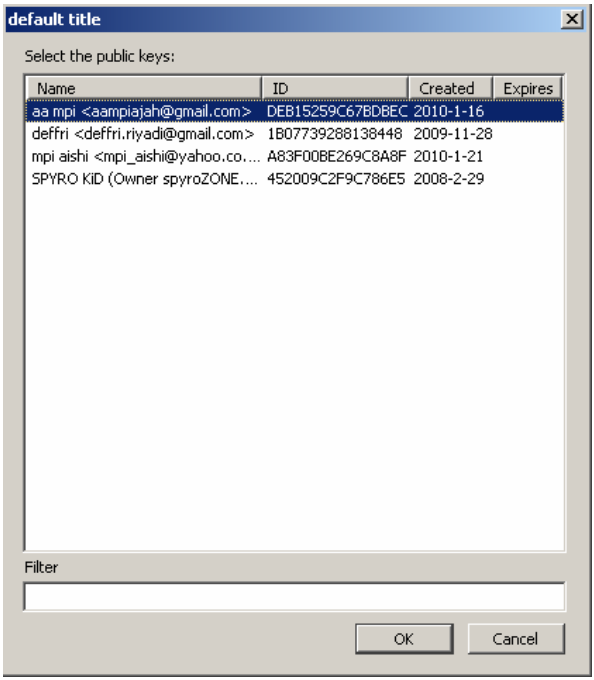
Langkah-langkahnya meliputi :

1. User A mengirim email ke User B yang sudah memiliki masing-masing public key.user bisa menulis email seperti biasa kemudian pilih menu *encrypt*, *sign atau encrypt a file and attach* serta *with a signature.* Pesan atau file yang dienkripi akan diubah menjadi file berkesktensi ascii (.asc).



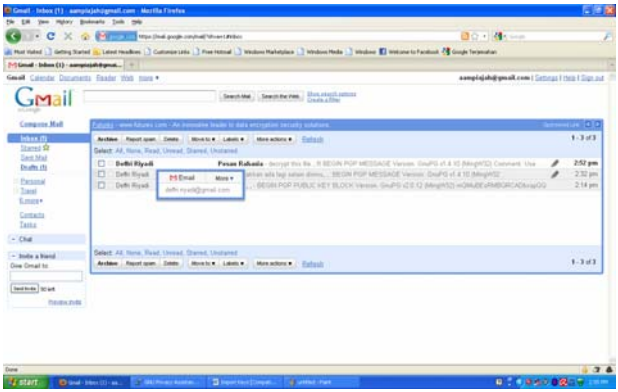
Gambar 4.12 Mengirim Pesan (User A)

2. Setelah memilih diantara menu tersebut, klik **send** maka anda akan diminta memilih public key email tujuan (USER B). disini saya memilih public key milik aa mpi sebagai tujuan pengiriman email.



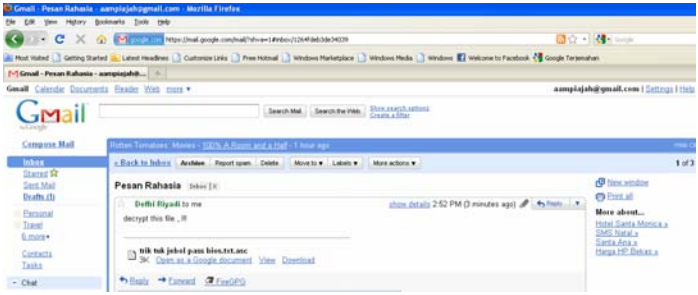
Gambar 4.13 Pilih Kunci Publik Tujuan

3. Tunggu beberapa saat email akan diproses untuk dikirimkan.
4. Untuk medekripsi email yang telah dikirim, buka account pada User B (penerima email).



Gambar 4.14 Account User B

5. Klik pada inbox atau kotak surat



Gambar 4.15 Isi Email User B

Pada gambar diatas file yang diterima telah diubah menjadi berekstensi ".txt.asc" ini terjadi karena file telah dikonversi menggunakan radix-64 pada saat proses enkripsi terjadi.

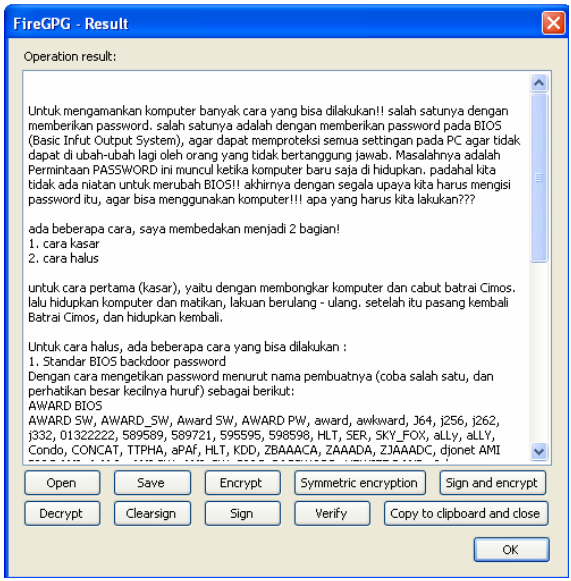
6. Untuk membuka file tersebut menjadi seperti semula ada beberapa cara bisa didownload terlebih dahulu terus didekripsi menggunakan GPG atau dengan cara menu view yang ada di Gmail kemudian blok

pesan tersebut dan klik kanan pada mouse pilih menu **FireGPG → Decrypt.**

7. Setelah itu akan muncul kotak dialog yang meminta anda untuk mengisi password kunci private.

Gambar 4.16 Kotak Dialog Private Key

8. Jika password key yang anda benar maka akan muncul hasil pesan yang berhasil di dekripsi, seperti dibawah ini :



Gambar 4.17 Hasil Dekripsi Pesan Email Dari User A

Pada gambar diatas terlihat hasil dekripsi pesan yang dikirim oleh user A.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melalui beberapa penelitian dan percobaan terhadap *Gnu Privacy Guard* dalam penyusunan Tugas Akhir ini khususnya pada windows, maka penulis dapat mengambil beberapa kesimpulan, yaitu :

- GPG merupakan sebuah teknologi pengenkripsian pesan atau *EMail*.. Fitur-fitur yang ada memudahkan pengguna untuk bisa menjaga kerahasiaan dan otentifikasi pesan. *OpenPGP* bahkan memberikan sesuatu yang lebih baik lagi, tidak hanya gratis dan bisa digunakan semua orang, pengguna pun diberikan kemampuan untuk memilih metode enkripsi yang digunakan ini berarti kesulitan ganda bagi orang yang berusaha memecahkan pesan sandi ini. Penyerang tidak hanya harus mencoba satu persatu kunci yang tepat tapi juga harus memilih metode atau algoritma yang digunakan.
- Tidak terbatas pada keamanan data saja, GPG juga memberikan kelebihan melalui pemberian tanda tangan digital nya. Melalui cara ini maka keaslian pesan, keabsahan pengirim dan anti penyangkalan dapat diperiksa dan dinyatakan kebenarannya. Dan sekali lagi, akan sangat sulit untuk mengubah tanda tangan digital ini, karena selain dibangkitkan dengan sebuah fungsi hash satu arah tanda tangan ini juga terenkripsi dengan menggunakan algoritma kunci publik.
- Mengenai masalah keamanan program, untuk menemukan kelemahan perangkat lunak (berupa *bug*) tidaklah mudah karena harus mendalami setiap potongan kode yang menyusun program tersebut. Untuk setiap *bug* GnuPG yang ditemukan, pihak pengembang biasanya selalu

segera memberikan solusinya berupa merilis *patch* untuk versi lama ataupun merilis program versi baru yang bebas dari *bug* tersebut.

- FireGPG, merupakan add-ons yang disediakan oleh Mozilla firefox yang digunakan untuk mendukung penuh penggunaan Gnu Privacy Guard lewat browser Mozilla versi 3 keatas, serta sudah terintegrasi dengan salah satu webmail terbesar seperti Gmail. Sehingga memudahkan pengguna dalam menggunaknnya khususnya bagi pengguna Windows.

5.2 Saran

Untuk meningkatkan efektifitas dari Gnu Privacy Guard (GPG), penulis memberikan beberapa saran sebagai berikut :

- Pengguna sebaiknya selalu waspada terhadap kemungkinan adanya *bug* pada setiap perangkat lunak, termasuk GnuPG. Dan jika menemukan *bug* tersebut, sebaiknya segera dipublikasikan, terutama ke pihak pengembang, supaya *bug* tersebut segera diatasi sehingga keamanan pesan tersebut bisa terus terjaga
- Bagi para pengguna Internet disarankan agar menggunakan program GPG dalam mengirim sebuah pesan karena kehandalan sistem ini dapat menjamin data atau informasi dari gangguan pihak-pihak yang tidak berhak.
- Untuk lebih mempermudah penggunaan GPG dalam keamanan mengirim sebuah email, untuk pengguna Windows disediakan pula GPG yang sudah berbasis GUI yaitu gpg4win yang bisa di download secra gratis di website resminya di http://www.gpg4win.org.

DAFTAR PUSTAKA

Abdullah, Lolly Amalia. *Tutorial Interaktif Keamanan sistem Transportasi Email Berbasil Open source (http://blog.poltek-malang.ac.id/media/3/20090528-g.%20Dokumentasi%20Email%20Exchange%20-%20A5.doc*, Waktu Akses : 16 Oktober 2009).

Anonym, *GNU Privacy Guard* (http://en.wikipedia.org/wiki/GNU_Privacy_Guard, Waktu Akses : 30 November 2009).

_____, IMAP (*http://id.wikipedia.org/wiki/IMAP*, Waktu Akses :12 Se;ptember 2009).

_____, *OSI Reference Model*. *http://id.wikipedia.org/wiki/OSI_Reference_Model*,, Waktu Akses : 16 Desember 2009.

_____, POP3. (*http://id.wikipedia.org/wiki/POP3*, Waktu Akses : 12 September 2009).

_____, Rfc4880. http://tools.ietf.org/rfc/rfc4880.txt, Waktu Akses : 16 Desember 2009.

_____, SMTP (*http://id.wikipedia.org/wiki/SMTP*, Waktu Akses :12 September 2009).

Ariyus, Dony. *Pengantar Ilmu Kriptografi Teori, Analisis, dan Implementasi*. Penerbit ANDI. Yogyakarta. 2008.

Firdaus, Yulian. *Gnu Privacy Guard* (http://yulian.firdaus.or.id/gnupg.php, Waktu Akses : 10 September 2009).

Heinze Manfred J., Bihlmeier Karl, Kramer Isabel, Wray Francis, Bahn Ute, Koch Werner. *Gpg4Win Novice.* The Free Software Foundation. 2006.

Kid, Spyro. *Menggunakan GPG & FireGPG Untuk Mengenkripsi Email* (http://hack.spyrozone.net/0243_Menggunakan_GPG_n_FireGPG_Untuk_Mengenkripsi_Email_by_SPYRO_KiD_WWW.SPYROZONE.NET_24_April_2008.html, Waktu Akses : 4 Oktober 2009).

Kurniawan, Yusuf. *Kriftografi Keamanan Internet dan Jaringan Komunikasi.* Informatika Bandung. Bandung. 2004.

Mansfield, Nihll. *Practical TCP/IP : Mendalami, Menggunakan, dan Troubleshooting Jaringan TCP/IP di Linux dan Windows (Jilid 1).* Penerbit ANDI. Yogyakarta. 2004.

Prasetyo, Didit Dwi. *Belajar Sendiri : Mail Service Berbasis Java Pada Server Windows dan Linux.* Penerbit PT. Elex Media Komputindo Kelompok Gramedia. Jakarta. 2004.

Rahayu, Flourensia Sapty (http://bebas.vlsm.org/v06/Kuliah/MTI-Keamanan-Sistem-Informasi/2005/124/124P-04-final2.0-Cryptography.pdf, Waktu Akses : 7 Desember 2009).

Sugianto, Anggriawan. *Analisis Keamanan Gnu Privacy Guard* (http://www.informatika.org/~rinaldi/Kriptografi/2007-2008/Makalah2/MakalahIF5054-2007-B-034.pdf, Waktu Akses : 29 Oktober 2009).

Syafrizal, Melwin. *Pengantar Jaringan Komputer*. Penerbit ANDI. Yogyakarta. 2005.

Wulandari. *Penggunaan Kriptografi Dalam Sistem Pengamanan Email* (http://www.informatika.org/~rinaldi/Kriptografi/2008-2009/Makalah1/MakalahIF30581-2009-a038.pdf, Waktu Akses : 18 Desember 2009).

DAFTAR RIWAYAT HIDUP

Data Diri :

Nama Lengkap	: Deffri Riyadi
Jenis Kelamin	: Laki - laki
Tempat, Tgl. Lahir	: Tangerang, 28 Januari 1987
Kewarganegaraan	: Indonesia
Status Pernikahan	: Belum Menikah
Agama	: Islam
Alamat	: Kp. Blok Kelapa RT.02/02 No.124 Ds. Serdang Wetan Legok Tangerang
No. Handpone	: 0856-9154-8670
No. Telpon	: 021-598-5982
E - Mail	: deffri.riyadi@gmail.com

Latar Belakang Pendidikan

1993 - 1999	: SDN Komplek Api, Tangerang
1999 - 2002	: SMPN 1 Curug, Tangerang
2002 - 2005	: SMAN 1 Curug, Tangerang
2005 - 2010	: Universitas Indonusa Esa Unggul Jakarta Fakultas Ilmu Komputer Jurusan S-1 Teknik Informatika

LAMPIRAN

Instalasi Gnu Privacy Guard

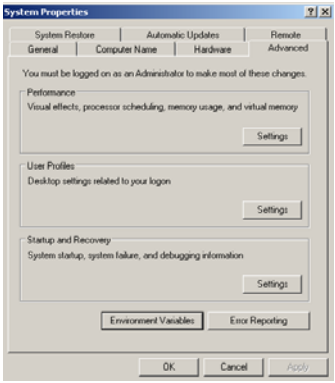
GNU Privacy Guard bisa anda download dari website resminya yaitu http://www.gnupg.org sebesar 1,5 MB yang merupakan versi 1.4.10 dan merupakan GPG berbasis *Command Line* (Perintah Baris).

Kita memulai proses instalasi dengan menjalankan *gnupg-w32cli-1.4.10b.exe.*

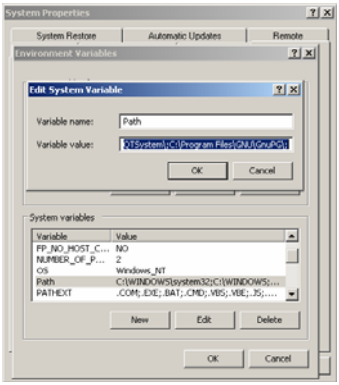
Setelah proses instalasi selesai, langkah selanjutnya setting GPG untuk memastikan GPG berjalan di Sistem Operasi Windows.

Setting GPG

Klik Start → My Computer klik kanan pilih menu Properties :



Pilih Tab Advanced pilih Environment Variables



Cari variable Path, kemudian edit dan tambahkan c:\Program Files\GNU\GnuPG\:\;

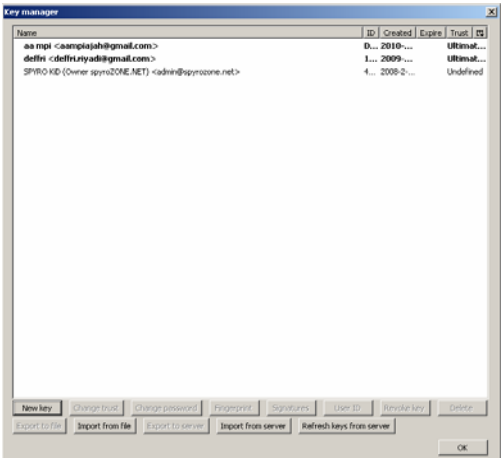
Konfigurasi FireGPG

Berikut ini adalah tambahan fitur GPG pada web browser Mozilla Firefox yang sudah terpasang FireGPG :

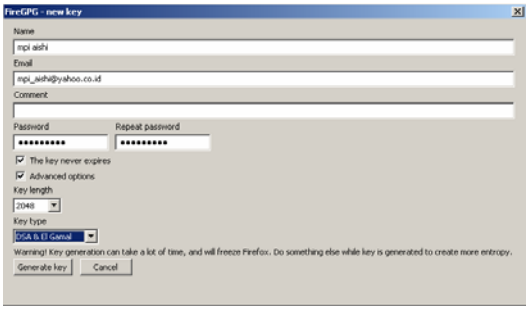
Membuat Pasangan Kunci pada FireGPG

Pada gambar diatas klik tools pilih menu FireGPG → Key Manager;

Tampilam Key Manager :



Pilih tombol “New Key”



Pada gambar di atas anda diminta mengisikan nama, email, password/passphrase. Panjang kunci dan algoritma yang dipakai untuk proses pembangkitan kunci. Setelah selesai klik “Generate Key”. Proses pembangkitan pasangan kunci akan berjalan beberapa saat.

TABEL ASCII CODE

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com