ANALISIS KEAMANAN E-MAIL MENGGUNAKAN *PRETTY GOOD PRIVACY*

Nandang Iriadi
Program Studi Manajemen Informatika
Bina Sarana Informatika
Jl.Jatiwaringin RayaNo.18 Jakarta 17411
nandangiriadi@yahoo.com

Abstract

Electronic mail (e-mail) has haved important communication means and which. thereby, user and business has haved concerned with privasi e-mail they and turn away to second software enkripsi e-mail commercial kailable to achieve confidentiality. e-mail protocol enkripsi very susceptible towards attack ciphertext where receivers e-mail unconsciously act as" dekripsi oracle" , that attack enough proper and hence be serious attention. attack towards e-mail that sent shaped file or message without compression prety good privacy (pgp) so message or file is not guaranteed the data security, but if compression file that sent (for example, file zip that sent to use pgp), that attack stills to work and can be used to restore original data. on the other side, compression is done by software enkripsi that sendiri(bila compression file is send)ed that causes attack towards file or message that reside in safe email from attack pembajakan(hijacking) email message.

Keyword: pgp, compression pgp

I. PENDAHULUAN

Perkembangan data elektronik yang semakin pesat, mengakibatkan dibutuhkannya pengamanan data elektronik yang cukup handal. Saat ini, setiap orang sangat mudah bertukar informasi mengenai segala hal, termasuk berbagi ilmu mengenai bagaimana caranya mengambil data pada pengiriman *electronic mail* (e-mail) secara ilegal.. Namun orang yang berniat jahat dapat mencari jalan lain untuk mengambil data rahasia yang dikirimkan melalui e-mail tersebut, yaitu dengan menggunakan metode *spoofing* atau pembajakan e-mail ditengah jalan.Dengan adanya kemungkinan akses ilegal yang menggunakan metode *spoofing* atau pembajakan e-mail ditengah jalan tersebut, maka perlu dikembangkan suatu mekanisme yang dapat mengamankan data dari akses ilegal yang dilakukan orang tidak bertanggung jawab pada level data rahasia pada pengiriman e-mail tersebut. Ada banyak cara yang bisa dilakukan untuk enkripsi data yang disimpan dalam data rahasia pada pengiriman e-mail tersebut salah satunya adalah dengan menggunakan metode enkripsi yaitu dengan aplikasi *Prety Good Privacy* (PGP)

Pada penelitian ini, penulis mencoba mengusulkan teknik enkripsi data dengan mempergunakan kompresi PGP. PGP merupakan mekanisme enkripsi yang sangat baik dengan mempergunakan *public* dan *private key*. PGP lebih dikenal untuk enkripsi pada e-mail dan file teks, serta beberapa fitur lain. Data penting yang hendak dikirim dalam e-mail, sebelumnya dienkripsi dengan mempergunakan PGP. Setelah dienkripsi, barulah *ciphertext* ini dikirimkan dengan menggunakan e-mail..

II. TINJAUAN PUSTAKA

A. Sejarah PGP

PGP versi 1 pertama kali dibuat pada tahun 1991 oleh Phil Zimmermann, walaupun telah dikembangkan sejak pertengahan tahun 1980-an. Pada mulanya PGP dibuat dengan tujuan untuk mengamankan komunikasi pada *Buletin Board System* dan juga melindungi dokumen dan pesan pribadi. Pada saat itu, PGP merupakan program yang gratis jika digunakan untuk alasan pribadi dan bukan komersial. Program PGP ini didapatkan dengan cara di-*download* langsung melalui internet berikut dengan *source code* lengkapnya. Nama "*Pretty Good Privacy*" diambil dari nama toko kelontong yang

terdapat pada suatu kota fiksi dalam cerita radio yang bernama "*Ralph's Pretty Good Grocery*".Enkripsi pesan dengan menggunakan PGP menjadi populer di seluruh dunia setelah diluncurkan pertama melalui internet. Pengguna PGP banyak juga yang berasal dari para aktivis sosial dari berbagai belahan dunia yang membutuhkan keamanan saat berkomunikasi. Menurut Sukmaaji dan Rianto (2008:175) kriptografi adalah ilmu yang berguna untuk mengacak data sedemikian rupa,sehingga tidak bias dibaca oleh pihak ketiga. Tentu saja data yang diacak harus bias dikembalikan kebentuk semula oleh pihak berwenang. Data yang ingin diacak biasanya disebut tek asli (*plain text*). Data yang ingin diacak dengan menggunakan kunci enkripsi(*encryption key*).*plain text* yang telah diacak disebut *cipher text*. Kemudian proses untuk mengembalikan *cipher text* ke *plaint text* dusebut dekripsi(*decryption*).

Menurut simarmata (2006:243),*Pretty Good Privacy* (PGP) adalah salah satu algoritma keamanan komunikasi data melalui internet untuk komunikasi harian semacam *elektronic mail* .PGP merupakan gabungan antara sistem pembuatan *digest*, enkripsi simetris dan asimetris.*Software* pengaman kriptografi yang cukup tinggi performansinya. Menurut kurniawan (2004:251) PGP dibuat oleh Zimmerman yang melakukan usaha-usaha berikut:

1. Memilih algoritma kriptografi terbaik yang ada sebagai komponen dasar pembentuk pgp
2. Mengintegrasikan algoritma-algoritma ini kedalamaplikasi serba guna yang independent terhadap sistem operasi dan *processor* dan dijalankan dengan sekumpulan kecil intruksi yang mudah digunakan.
3. Membuat paket dan dokumentasinya, mencakup kode sumber,dapat diakses secara gratis melalui internet,*bulletin board*, dan jaringan komersial semacam *compuserve*.
4. Melakukan perjanjian dengan perusahaan (*viacrypt)* untuk memberikan kompatibiliatas yang penuh, versi komersial PGP yang berharga murah.

PGP mempunyai dua tingkatan kunci, yatu kunci rahasia (simetri) yang disebut juga *session key* untuk enkripsi data dan pasangan kunci privat-kunci publik untuk pemberian tanda tangan digital serta melindungi kunci simetri. Kunci simetri hanya dipakai sekali *(one-time)* dan dibuat secara otomatis dari gerakan *mouse* atau ketikan tombol *keyboard*.PGP pertama kali muncul sebagai aplikasi sederhana untuk melindungi komunikasi antarkomputer dengan menggunakan algoritma RSA dari serangan penyadap. Seiring dengan perkembangan internet, gangguan yang terjadi tidak hanya pada jalur komunikasi saja tetapi juga pada dokumen-dokumen yang terdapat pada komputer pribadi. Oleh karena itu, PGP menambahkan fitur-fitur baru pada aplikasinya, antara lain fitur untuk menghapus dokumen secara aman, enkripsi pada *hard disk*, dan enkripsi pada jaringan. Selain itu, dikembangkan juga antarmuka yang *user-friendly* untuk memudahkan penggunaan PGP. Perubahan juga terjadi pada target pasar pengguna PGP yang semula individu menjadi peusahaan karena hanya perusahaan yang bersedia untuk membayar program pengamanan seperti PGP.PGP menjadi terkenal dan banyak digunakan karena merupakan program pertama yang menawarkan penggunaan kriptografi secara kuat. Pada awalnya PGP merupakan program yang gratis, pengguna dapar melihat *source code* (kode sumber) dan mengembangkannya jika diinginkan, akan tetapi pada akhirnya PGP dibeli oleh perusahaan perangkat lunak dan menjadi tidak gratis .Secara garis besar PGP memiliki tiga fitur utama, yaitu:

1. Fitur untuk melakukan enkripsi dan menandatangani dokumen.
2. Fitur untuk melakukan dekripsi dan verifikasi tanda tangan.
3. Fitur untuk mengelola kunci PGP yang dimiliki oleh pengguna.

Setiap orang yang menggunakan PGP harus menerima kunci publik terlebih dahulu. Kunci publik tersebut didapatkan dengan cara mengirim email kepada rekan yang akan diajak berkomunikasi dengan memanfaatkan program PGP atau dengan terhubung secara langsung ke *server* yang memegang kunci publik. Setelah kunci publik didapat maka langkah berikutnya adalah melakukan verifikasi terhadap kunci tersebut. Proses verifikasi tersebut dapat dilakukan secara tidak langsung, hal ini merupakan fitur utama dari PGP.Jika sebuah pesan dikirimkan untuk beberapa pengguna pada satu buah *host*, pesan informasi tersebut hanya perlu dikirimkan satu kali. Jika pesan-pesan tersebut sudah siap untuk dikirimkan ke sebuah *host,* maka SMTP *sender* dapat membuka sebuah koneksi TCP,

mentransfer semua pesan tersebut, kemudian menutup koneksi, dari pada membuka dan menutup sebuah koneksi berulang-ulang untuk tiap-tiap pesan. Pengirim bisa memasukkan kembali surat tersebut ke dalam antrian untuk dikirimkan beberapa waktu kemudian, tetapi bila setelah periode waktu tertentu surat masih tidak dapat dikirimkan, maka surat tersebut tidak jadi dikirimkan agar tidak memenuhi antrian untuk jangka waktu yang tidak terbatas.

Kriptografi

B. Menurut Simarmata (2006:199), kriptografi adalah suatu ilmu yang mempelajari penulisan secara rahasia. Teknik ini digunakan untuk mengubah data kedalam kode-kode terteentu sehingga informasi yang disimpan atau ditransmisikan melalui jaringan yang tidak aman (misalnya saja internet) tidak dapat dibaca oeh siapapunkecuali orang-orang yang berhak.. dalam menjaga kerahasiaan data,kriptografi meantransformasikan data jelas(*plaintext*) kedalam bentuk data sandi(*ciphertext*) yang tidak dapat dikenali. *Ciphertext* inilah yang kemudian dikirimkan oleh pengirim (*sender*) kepada penerima(*receiver*). Setelah sampai dipenerima, *chiper text* tersebut ditransformasikan kembali kedalam bentuk *plaintext* agar dapat dikenali.

a) Tujuan kriptografi

Menurut Simarmata (2006:201)Tujuan kriptografi adalah sebagai berikut:

1. *Convidentiality*

   Memberikan kerahasiaan pesan dan menyimpan data dengan menyembunyikan informasi lewat teknik-teknik enkripsi. *Message integrity* memberikan jaminan bahwa setiap bagian tidak akan mengalami perubahan dari saat data dibuat atau dikirim sampai saat data itu dibuka.
2. *Non-repudation*Memberikan cara untuk membuktikan bahwa suatu dokumen datang dari seseorang apabila ia mencoba menyangkal memiliki dokumen tersebut.
3. *Authentication*

   Memberikan dua layanan,yang pertama dan memberikan jaminan ke otentikannya. Kedua, untuk menguji dentitas seseorang apabila ia akan memasuki sebuah sistem.

b) Macam-macam algoritma kriptografi

Menurut Kurniawan (2004:6) terdapat dua jenis algoritma kriptografi berdasar jenis kuncinya:

1. Algoritma simetri (konvensional)

   Algoritma simetri disebut juga sebagai algoritma konvensional adalah algoritma yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Disebut konvensional karena algoritma yang biasa digunakan orang sejak berabad-abad yang lalu adalah algoritma jenis ini. Algoritma simetri sering juga disebut sebagai algoritma kunci rahasia,algoritma kunci tunggal atau algoritma satu kunci dan mengharuskan pengirim dan penerima menyetujui suatu kunci tertentu sebelum mereka dapat berkomunikasi dengan aman. Keamanan algoritma simetri tergantung pada kunci, membocorkan kunci berarti bahwa orang lain dapat mengenkrip dan mendekrip pesan.
2. Algoritma asimetri

   Algoritma asimetrik (juga disebut algoritma kunci publik) didesain sedemikian sehingga kunci yang digunakan untuk enkripsi berbeda.

C. Operasi PGP

Menurut Kurniawan(2004:253) PGP melakukan operasi utama yaitu:

1. Otentifikasi
   Fungsi-fungsi dalam PGP dapat dilihat pada tabel 1.
2. Kerahasiaan
   PGP menggunakan IDEA dengan kunci 128 bit untuk menyandikan data dan menggunakan *Mode Cipher Feed Back* (CFB) dengan vektor awal. Hal yang selalu harus diperhatikan adalah distribusi kunci dalam PGP,setiap kunci konvensioanal hanya digunakan sekali,artinya,setiap kali ada pesan yang akan terenkrip,dibangkitkan kunci baru 128 bit secara acak. Meskipun dalam dokumentasi kunci ini disebut sebagai kunci sesi,kenyataanya adalah kunci satu-waktu (*one-time key*), karena hanya digunakan sekali,kunci sesi ini digabungkan dengan pesan yang sudah dienkrip dengan kunci tersebut kemudian dikirim bersama-sama. Untuk melindunginya, kunci sesi ini dienkrip dengan kunci publik penerima.

Tabel 1 Fungsi dalam PGP

Fungsi	Algoritma yang dipakai	Deskripsi
Enkripsi pesan	IDEA,RSA(*Rivest Shamir Adleman*)	Pesan yang dienkrip menggunakan IDEA dengan kunci sesi sekali pakai yang dibangkitkan pengirim kunci sesi dienkrip pengirim menggunakan RSA dengan kunci public penerima dan digabungkan kedalam pesan
Tanda tangan digital	RSA,MD5	Kode has pesan dibuat menggunakan MD5.hash ini dienkrip menggunakan RSA dengan kunci privatpengirim dan digabungkan kedalam pesan.
Kompresi	ZIP	Pesan dapat dikompres untuk disimpan atau dikirim dengan zip
Kompatibilitas *email*	Konversi *radix* (*base*) 64	Untuk mempermudah penggunaan dalam aplikasi email,pesan yang terenkrip,dapat dikonversi kedalam string ASCII menggunakan konversi radix (*base*) 64.
segmentasi	-	Untuk mengakomodasi batasan ukuran pesan maksimum,PGP melakukan segmentasi dan penyusunan ulang

Sumber:Kurniawan(2004:253)

3. Kompresi
   Secara default,PGP mengkompres pesan setelah dilakukan tanda tangan,namun sebelum enkripsi. Ini memberikan kaeuntungan penghematan ruang untuk pengiriman e-mail dan penyimpanann file.
4. kompatibilitas *e-mail*
   ketika PGP digunakan,paling sedikit satu blok yang dikirim dienkrip. Jika hanya layanan tanda tangan yang digunakan,maka *message digest* dienkrip (dengan kunci privat RSA pengirim). Bila layanan keamanan, pesan ditambah tanda tangan (jika ada) dienkrip (dengan kunci IDEA sekali pakai). Jadi sebagian atau seluruh blok atau seluruh blok yang dihasilkan PGP,terdiri dari aliran sejumlah octet 8-bit. Namun terdapat sistem e-mail yang hanya mengijinkan pengguna blok yang terdiri teks ASCII. Untuk mengakomodasi batasan ini, PGP memberikan layanan konversi aliran biner 8-bit menjadi karakter ASCII yang dapat dicetak.

Garfinkel dalam bukunya yang berjudul "*PGP : Pretty Good Privacy*" mengemukakan bahwa keamanan komputer *(computer security*) melingkupi empat aspek, yaitu : *privacy, integrity, authentication*, dan *availability*. Selain keempat aspek tersebut, dua aspek lainnya yaitu *access control* dan *non-repudiation* (Raharjo, 2005:15-20) :

1. *Privacy/Confidentiality*
   Inti utama aspek *privacy* atau *confidentiality* adalah usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. *Privacy* lebih ke arah data-data yang bersifat privat, sedangkan *confidentiality* biasanya berhubungan dengan data yang diberikan ke pihak lain untuk keperluan tertentu (misalnya sebagai bagian dari pendaftaran sebuah servis) dan hanya diperbolehkan untuk keperluan tertentu tersebut.
2. *Integrity*
   Aspek ini menekankan bahwa informasi tidak boleh diubah tanpa seijin pemilik informasi. Adanya virus, *trojan horse*, atau pemakai lain yang mengubah informasi tanpa ijin merupakan contoh masalah yang harus dihadapi.
3. *Authentication*
   Aspek ini berhubungan dengan metode untuk menyatakan bahwa informasi benar-benar asli, orang yang mengakses atau memberikan informasi adalah benar-benar orang yang dimaksud, atau *server* yang terhubung adalah benar-benar server yang asli.
4. *Availability*
   Aspek *availability* atau ketersediaan berhubungan dengan aspek ketersediaan informasi ketika dibutuhkan. Sistem Informasi yang diserang dapat menghambat atau meniadakan akses ke informasi. Contoh hambatan adalah serangan yang sering disebut dengan "*denial of service attack*" (DoS attack), dimana server dikirimi permintaan palsu yang bertubi-tubi atau permintaan yang diluar perkiraan sehingga tidak dapat melayani permintaan lain atau bahkan sampai terjadi *down*, *hang*, *crash* pada sistem.
5. *Access Control*
   Aspek ini berhubungan dengan cara pengaturan akses informasi. Hal ini biasanya berhubungan dengan klasifikasi data (*public, private, confidential, top secret*) & user (*guest, admin, top manager*, dsb.), mekanisme *authentication* dan *privacy*. *Access control* seringkali dilakukan dengan menggunakan kombinasi *user-id/password* atau dengan menggunakan

mekanisme lain seperti kartu, *biometrics*, dan sebagainya.

6. *Non-repudiation*
   Aspek ini menjaga agar seseorang tidak dapat menyangkal telah melakukan sebuah transaksi.

D. Kunci-Kunci Kriptografi

Pretty Good Privacy menggunakan empat macam kunci yaitu, kunci konvensional sesi satu waktu (*one time key*), kunci publik, kunci privat dan kunci konvensional turunan *passphrase*.Setiap pemegang kunci PGP harus menjaga file berisi pasangan kunci publik/privat miliknya serta file yang berisi kunci publik relasinya. Menurut Kurniawan (2004:259) ada 4 macam kunci dalam PGP:

1. Kunci sesi dengan algoritma enkripsi IDEA digunakan untuk mengenkrip pesan untuk dikirim. Setiap kunci sesi hanya digunakan sekali dan dibangkitkan secara acak.
2. Kunci publik dengan algoritma enkripsi RSA digunakan untuk mengenkrip kunci sesi untuk dikirimkan bersama pesan. Pengirim dan penerima harus mendapatkan kunci publik rekan-rekannya.
3. Kunci privat dengan algoritma enkripsi RSA digunakan untuk mengenkrip sidik jari pesan untuk membentuk tanda tangn digital. Kunci privat hanya boleh diketahui oleh pemiliknya.
   4. Kunci turunan *passphrase* dengan algoritma enkripsi IDEA digunakan untuk mengenkrip kuinci privat yang disimpan oleh pemilik kunci privat.

E. Prinsip Kerja PGP

Menurut Moha (2006) prinsip kerja dari PGP adalah sebagai berikut :

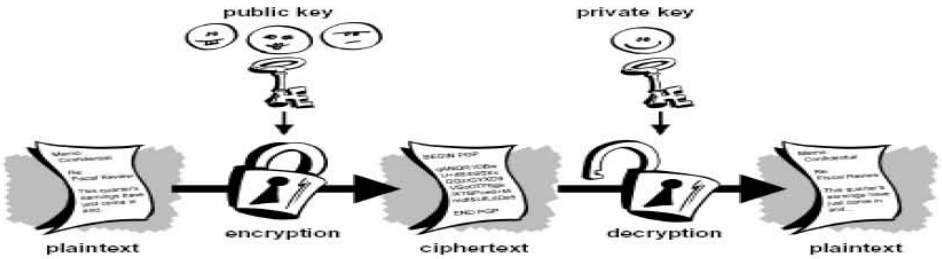
1. PGP menggunakan teknik yang disebut *public kec encryption* dengan dua kode. Kode-kode ini berhubungan secara intrinstik, namun tidak mungkin untuk memecahkan satu sama lain.
2. Ketika dibuat satu kunci, maka secara otomatis akan dihasilkan sepanjang kunci, yaitu kunci publik dan kunci rahasia, PGP dibuat berdasarkan metode *public key* yang menggunakan dua kunci. Satu kunci adalah *public key* yang disebarkan kesemua orang yang akan mengirim informasi. Kunci yang lain adalah *private key* yang digunakan untuk mendekripsikan informasi yang diterima.

PGP menggunakan dua kunci, *Pertama,* kunci untuk proses enkripsi (kunci publik). Disebut kunci publik karena kunci yang digunakan untuk enkripsi ini akan diberitahukan kepada umum.

Menurut wahana komputer (2005:246) secara sederhana *public key* berjalan sebagai berikut:

1. masing-masing sistem dalam *network* menciptakan sepasang kunci yang digunakan untuk enkripsi dan dekripsi dari informasi yang diterima.
2. Masing-masing sistem akan mempunyai *public key* dan memasangnya dalam register umum atau file, sedang pasangannya tetap dijaga sebagai kunci pribadi
3. apabila A ingin mengirim pesan kepada B, maka A akan mengenkripsi pesannya dengan *public key* dari B.
4. Ketika B menerima pesan dari A maka B akan menggunakan *private key* miliknya untuk mendekripsikan pesan dari A.

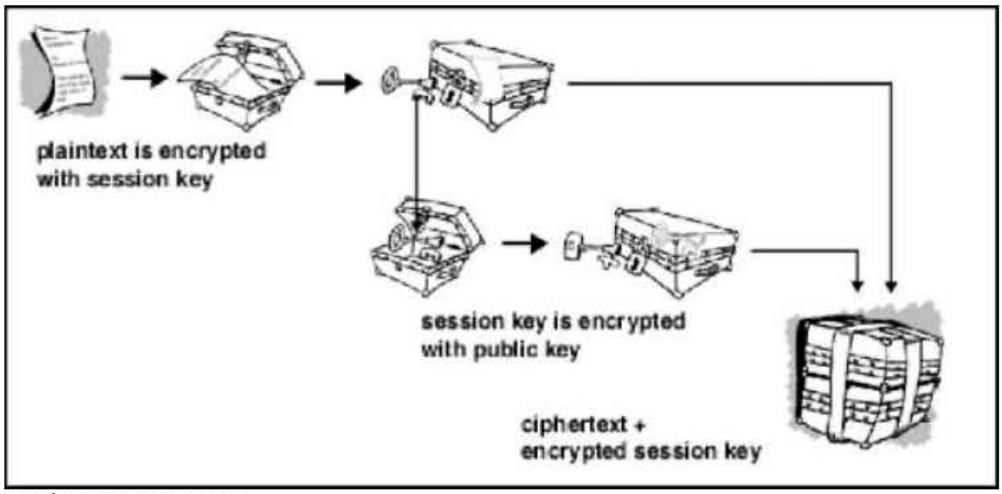
Kelemahan dari metode *public key cryptography* adalah jika dibandingkan dengan metode enkripsi konvensioanal, algoritma ini bersifat lebih kompleks. Akibatnya, untuk perbandingan ukuran dan harga perangkat keras, metode *public key* akan menghasilkan *perpormance* yang lebih rendah.



Sumber:wahana komputer(2005:246).

Gambar 1 Proses Enkripsi dengan PGP

Algoritma *public key* tampak seperti gambar 1 proses enkripsi dengan PGP. Pada algoritma tersebut, pengetahuan tentang algoritma enkripsi ditambah kunci enkripsi tidak cukup untuk menentukan kunci dekripsi.Orang yang akan mengirimkan *e-mail* rahasia kepada kita harus mengetahui kunci publik ini. *Kedua,* kunci untuk proses deskripsi (kunci pribadi). Disebut kunci pribadi karena kunci ini hanya diketahui oleh kita sendiri.Karena dengan *conventional crypto,* di saat terjadi transfer informasi kunci, diperlukan suatu *secure channel*. Jika kita memiliki suatu *secure channel,* mengapa masih menggunakan *crypto?* Dengan *public key system,* tidak akan menjadi masalah siapa yang melihat kunci milik kita, karena kunci yang dilihat orang lain adalah yang digunakan hanya untuk enkripsi dan hanya pemiliknya saja yang mengetahui kunci rahasia tersebut.Sebagai alat pengamanan *email* tentu saja tingkat keamanan PGP perlu diperhatikan. Sehingga data yang berupa informasi yang dikirimkan benar – benar aman dari ancaman para penyadap. Selain itu Keamanan PGP sangat bergantung pada algoritma yang digunakan. Semakin bagus kode – kode algoritma yang digunakan semakin susah para *hacker* atau *cracker* dalam memecahkan suatu algoritma yang digunakan berarti akan semakin mudah informasi (pesan) akan sampai ditangan orang yang berhak menerima isi informasi tersebut.pada gambar 2 adalah ilustrasi gambar dari sistem kerja PGP (*Pretty Good Private*) dari sebuah pesan *plaintext* yang kemudian dienkrip dengan kunci publik(public) menjadi ciphertext, setelah itu didekrip lagi menjadi plaintext dengan kunci pribadi (*private*) sehingga kembali menjadi *plaintext* yang sudah bisa dibaca. PGP mengkombinasikan fitur-fitur terbaik yang terdapat pada kriptografi konvensional dengan kriptografi kunci publik. PGP merupakan sistem kriptografi *hybrid*.Enkripsi pada PGP menggunakan kriptografi kunci publik dan juga sistem yang menggabungkan kunci publik 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 alternatif tersebut melalui manajemen server secara otomatis.Enkripsi e-mail pada PGP menggunakan algoritma enkripsi kunci asimetri dengan pasangan kunci publik-kunci privat. Pengirim email menggunakan kunci publik penerima untuk melakukan enkipsi kunci rahasia yang digunakan pada algoritma *cipher* simetri. Pada akhirnya kunci akan digunakan untuk melakukan enkripsi *plaintext*. Hampir semua knci publik pengguna PGP tersimpan pada *server* kunci PGP yang tersebar di seluruh dunia. Berikut ini adalah skema enkripsi dengan menggunakan PGP:

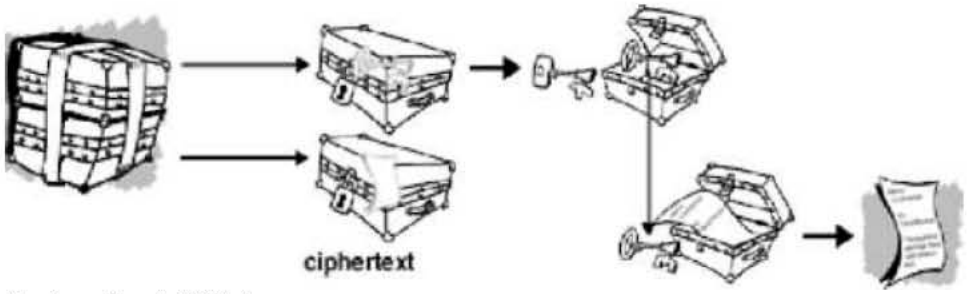


Sumber : Tanoto.2006:4

Gambar 2 Skema Enkripsi PGP

Penerima *email* yang terenkripsi tersebut menggunakan kunci sesi *(session key)* untuk melakukan dekripsi terhadap *email* tersebut.

Kunci sesi ini terdapat pada email yang terenkripsi tersebut dan diperoleh dengan cara mendekripsinya dengan menggunakan kunci privat.



Sumber : Tanoto.2006:4

Gambar 3 Skema Dekripsi PGP

Strategi yang sama digunakan untuk mendeteksi apakah suatu pesan sudah mengalami perubahan atau belum dan juga untuk menentukan apakah pesan berasal dari pengirim yang sebenarnya. Pengirim menggunakan enkripsi PGP untuk memberikan tanda tangan digital *(digital signature)* pada pesan dengan algoritma RSA atau DSA. Untuk melakukannya, PGP melakukan komputasi nilai *hash (message digest)* dari *plaintext* untuk kemudian membubuhkan tanda tangan digital dari nilai *hash* tersebut dengan menggunakan kunci privat pengirim. Penerima pesan melakukan perhitungan nilai hash terhadap plainteks semula dan menggunakan kunci publik penerima untuk memberikan tanda tangan digital pada pesan tersebut. Jika tanda tangan hasil komputasi ini sesuai dengan tanda tangan yang terdapat pada pesan yang diterima maka pesan dapat diterima dengan tingkat kepercayaan tinggi sebagai pesan yang asli tanpa adanya perubahan.Baik ketika melakukan enkripsi pesan maupun pada saat melakukan verifikasi tanda tangan, sangatlah penting bahwa kunci publik yang digunakan untuk mengirim pesan pada seseorang juga merupakan "milik" penerima pesan. Dengan *men-download* kunci publik dari *server* kunci PGP, ada kemungkinan terjadinya penyadapan oleh pihak yang tidak bertanggung jawab. Akan tetapi, PGP telah menyediakan cara untuk mendistribusikan kunci publik dengan menggunakan sertifikat identitas yang dibangkitkan dengan algoritma kriptografi. Sejak versi pertama, PGP telah memberikan suatu skema pembuatan sertifikat secara internal yang disebut *web of trust*. Kunci publik yang diberikan dapat ditandatangani secara digital oleh pihak ketiga untuk menguji asosiasi antara seseorang dengan kunci tersebut. Skema ini juga memiliki beberapa tingkat kepercayaan untuk hasil pengujian tanda tangan.*Web of trust* adalah suatu model kepercayaan yang kumulatif. Sebuah sertifikat dapat dipercaya secara langsung atau dipercaya dengan melalui perantara sertifikat lainnya. PGP menggunakan tanda tangan digital sebagai bentuk perkenalan. Ketika pengguna menandatangani kunci lain maka pengguna tersebut akan menjadi pengenal *(introducer)* kunci tersebut. Dengan cara demikian maka akan terbentuk suatu jaringan kepercayaan yang disebut *web of trust*.

F. Algoritma pada PGP

Setiap algoritma kriptografi dapat memecahkan suatu permasalahan tertentu. Versi terbaru dari PGP mendukung berbagai jenis algoritma yang dapat dipilih untuk digunakan. Hal ini sesuai dengan prinsip disribusi PGP, yaitu setiap pengguna dapat menggunakan algoritma kripografi tertentu yang dianggap aman. Berikut ini adalah penjelasan mengenai tipe dan jenis algoritma kripografi tersebut (Tanoto,2006:8):

a. *Cipher* Blok
   Algoritma enkripsi simetri melakukan enkripsi secara normal di mana kunci yang sama digunakan untuk melakukan enkripsi dan dekripsi. *Cipher block* adalah cipher simetri yang beroperasi pada blok berukuran 8 atau 16 byte. Kunci *cipher block* juga memiliki ukuran tertentu, umumnya 16 atau 128 *byte*. *Cipher block* banyak digunakan karena algoritma ini cepat, mudah untuk diprogram, efisien, dan dapat menangani data dalam jumlah besar.
b. Algoritma Kunci Publik

Algoritma kunci publik memiliki dua jenis kunci, yaitu kunci publik dan kunci privat. Algoritma ini biasanya lambat dan juga rentan terhadap beberapa serangan kriptanalis tertentu. Oleh karena itu, algoritma ini hanya akan digunakan untuk melakukan enkripsi pada bagian tertentu pada data, seperti kunci sesi *(session key)* dari cipher blok atau untuk mengenkripsi nilai hash tertentu. Algoritma kunci publik umumnya digunakan untuk melakukan enkripsi dan tanda tangan digital.

c. Fungsi *Hash*

Fungsi *hash* menerima input berupa panjang seluruh dokumen atau pesan untuk kemudian menghasilkan nilai *hash* atau disebut juga *fingerprint* dari dokumen dengan ukuran tertentu, biasanya 16 atau 20 *byte*. *Fingerprint* memiliki dua atribut penting, yaitu seorang tidak akan dapat menemukan dokumen hanya dengan melihat *fingerprint* dokumen tersebut dan pada kenyataannya tidak ada dua dokumen yang memiliki *fingerprint* yang sama. Secara teori mungkin saja terdapat *fingerprint* yang sama untuk dua jenis dokumen, akan tetapi karena nilai hash yang sangat banyak maka akan dibutuhkan waktu yang sangat lama untuk membangkitkan *fingerprint* yang sama tersebut. Oleh karena itu, enkripsi cukup dilakukan terhadap nilai *hash* suatu dokumen bukan terhadap keseluruhan dokumen.

d. Algoritma *Secret Sharing*

Algoritma ini tidak dibutuhkan pada fitur dasar dari PGP tetpai dapat digunakan pada versi yang baru. Algoritma ini dapat membagi kunci privat menjadi sejumlah *m* blok sehingga jika beberapa orang menggabungkan sejumlah *n* blok secara bersamaan menjadi *m* blok maka akan didapatkan kembali kunci privat tersebut. Algoritma ini biasanya digunakan pada perusahaan yang memiliki tiga orang direktur, di mana dibutuhkan dua dari tiga orang untuk menandatangani suatu kontrak.

G. Dekripsi kunci privat

ketika pengguna akan memakai kunci privatnya,PGP akan memintanya memasukkan *passphrase*. Biasanya *passphrase* digunakan untuk melindungi kunci privat yang tersimpan dalam *disk*.pengguna mengetikkan *passphrase* yang digunakan untuk mengenkrip kunci privat,ketika sistem membangkitkan pasangan kunci publik atau privat baru dengan RSA,PGP menanyakan pasphrase pengguna. *Passphrase* dimasukkan MD5 untuk membangkitkan kode hash 128 bit.*passphrase* dibuang. Kunci privat dienkrip menggunakan IDEA dengan kunci yang berasal dari kode hash diatas. Kode *hash* dibuang, dan kunci privat yang terenkrip disimpan pada ring kunci privat. menurut Kurniawan (2004:262) PGP memiliki dua macam bilangan acak:

1. Bilangan acak yang sebenarnya
   a) Digunakan untuk membangkitkan pasangan kunci RSA
   b) Memberikan masukan awal (s*eed*) untuk pembangkit bilangan acak semu.
   c) Memberi masukan tambahan selama pembangkitan bilangan acak semu.
   d) Dinamakan bilangan acak sebenarnya karena tidak bisa dibangkitkan kembali.

   PGP menyimpan 256 *byte buffer* atau bit-bit acak. Setiap memerlukan bit-bit ini, PGP merekam waktu komputer dalam format 32 bit, lalu menunggu pengguna mengetikan ketikan acak. Ketika menerima ketikan pengguna, PGP merekam waktu pengetikan.
2. Bilangan acak semu
   a) Digunakan untuk membangkitkan kunci sesi
   b) Digunakan untuk membangkitkan vektor inisialisasi,untuk digunakan dengan kunci sesi dalam enkripsi mode CFB.
   c) Dinamakan bilangan acak semu karena dapat dibangkitkan kembali.

Bilangan acak semu menggunakan *seed* 24 *oktet* (1 *oktet* = 8 bit) dan menghasilkan kunci sesi 16 *oktet*,vektor inisiasi 8 *oktet* dan *seed* baru digunakan bagi pembangkitan bilangan semi acak berikutnya. Algoritma didasarkan pada algoritma ANSI X9.17 dengan IDEA sebagai algotitma intinya menggantikan 3DES untuk enkripsinya.algoritma menggunakan struktur data berikut:

1. Input

   File randseed.bin (24 oktet): bila kosong, diisi dengan 24 oktet bilangan acak yang sebenarnya.

2. Pesan
   Kunci sesi dan vektor inisiasi yang akan digunakan untuk mengenkrip pesan.
3. Output
   Struktur data internal.

H. Kompresi

Pretty Good Privacy mengkompresi pesan setelah dilakukan tanda tangan, namun sebelum enkripsi. Hal ini dilakukan demi penghematan ruang untuk pengiriman e-mail dan penyimpanan file.Tanda tangan dibangkitkan sebelum kompresi dengan alasan (Kurniawan, 2004:257):

1. Lebih disukai menandatangani pesan yang belum dikompres kita tidak perlu menyimpan pesan dalam keadaan terkompres untuk pengecekan tanada tangan berikutnya. Bila seseorang menandatangani pesan yang terkompres, maka diperlukan menyimpan pesan dalam keadaan terkompres atau mengkompres ulang pesan ketika akan melakukan verifikasi..
2. Algoritma kompresi tidak deterministik. Artinya hasil kompres terhadap pesan yang sama oleh *software* yang berbeda dapat memberikan hasil yang tidak sama. Hal ini dapat terjadi karena program kompresi memberikan pilihan antara kecepatan kompresi dengan rasio kompresi. Namun algoritma yang berda-beda dapat saling beroperasi bersama, karena setiap versi algoritma dapat dengan tepat mendekompres keluaran versi lainnya. Menjalankan fungsi hash dan tanda tangan setelah kompres akan membatasi seluruh implementasi PGP untuk menggunakan algoritma kompresi yang sama persis.
3. Enkripsi pesan yang dilakukan setelah kompresi untuk memperkuat keamanan kriptografi. Karena pesan yang dikompres memiliki sedikit redudansi dibanding *plaintext* aslinya, sehingga analisis *cipher*nya menjadi lebih sulit.

PGP secara *default* belum menyandikannya (kecuali data yang sudah dikompresi, seperti dijelaskan di atas).Algoritma kompresi diterapkan sebelum dan sesudah enkripsi dekripsi. Kesulitan-kesulitan muncul dari fakta bahwa program menggunakan *header* paket termasuk dalam data dienkripsi untuk diproses lebih lanjut. Modifikasi ini *header* biasanya akan menyebabkan serangan gagal karena program tidak dapat memproses *header error*. Selain itu, PGP menggunakan pesan untuk periksa dan melindungi integritas data., cek *digest* gagal dan menyebabkan program untuk *output* pesan peringatan yang mungkin akan menyebabkan pengguna menjadi curiga dan gagal untuk mengembalikan pesan *cipherteks* didekripsi yang dipilih.

III. PEMBAHASAN

Pada saat ini ada beberapa cara enkripsi data, seperti dengan *multi digest five* atau MD5 yang sangat populer pada aplikasi berbasis web dan PGP yang sering dipakai untuk enkripsi data dan e-mail. PGP memiliki kelebihan dengan kemampuannya untuk enkripsi dengan mempergunakan *publik/private key cryptosystem.* Pada penelitian ini, akan dipergunakan teknik enkripsi dengan mempergunakan PGP yang berjalan dibawah perintah *shell* DOS.

Untuk melakukan proses enkripsi dan dekripsi data, Aplikasi ini berjalan pada sistem operasi berbasis *Windows*. Berikut ini perintah enkripsi data yang disimpan dalam file teks dengan nama pln.txt dengan mempergunakan PGP yang berjalan dibawah shell DOS dan hasil enkripsi diletakkan kedalam file teks chp.txt :

```
C:\PGP>pgpe -c -aftz -o pln.txt > chp.txt
```

PGP menggunakan teknik kriptografi yang tersedia seperti, simetris dan asimetris enkripsi dan dekripsi dan tanda tangan digital untuk menciptakan rahasia dan dikonfirmasi oleh *e-mail*. Pada dasarnya PGP mencakup RSA, DSS, CAST-128, IDEA atau 3DES, SHA-1. yang dianggap aman. PGP adalah perangkat lunak *open source* yang tersedia bebas, sehingga menarik pengguna *internet* untuk mengolah pesan *e-mail* secara aman. Sebuah paket PGP memiliki bagian header dan bagian data. *Header* menyimpan informasi seperti jenis paket dan panjang, antara lain. Bagian data berisi *payload* dari paket, yang tergantung pada jenis paket. Spesifik dari format paket yang sedikit berbeda dalam berbagai versi PGP dan dalam spesifikasi, fokus di PGP 10. Pertimbangkan pesan *e-mail* atau file. PGP kompresi mengenkripsi pesan sebagai berikut:

1. Sebuah sesi acak atau "sesi tombol" K dihasilkan, yang dienkripsi dengan

penerima *pub-lic* pk kunci, dan dikemas dalam paket sesi publik Tombol dienkripsi
(PKESKP). Output dapat direpresentasikan sebagai berikut: (PKESKP HEADER, EPK (K).

2. Pesan M adalah dikemas dalam sebuah paket data literal (LDP), menghasilkan: LDP = (LP HEADER, M)
3. LDP dikompresi menggunakan algoritma *deflate* [4], dan menjadi *payload* dari paket data terkompresi (CDP): CDP = (CP HEADER, deflate (LDP).
4. CDP dienkripsi dengan algoritma enkripsi *symmetric-key* (yaitu, blok cipher) dan K kunci, cipher menggunakan umpan balik (CFB) mode..
   Hal ini memberikan *ciphertext* Ci, C2, C3.*Ciphertext* dirumuskan dalam simetris enkripsi data paket (SEDP) sebagaiberikut:
   (SEDP HEADER, C1, C2, C3 ,...).
5. Pesan berikut akan dikirim ke penerima: (PKESKP HEADER, EPK (K)) (SEDP HEADER, C, C2, C3 ,...). Penerima, membalikkan langkah di atas, menggunakan kunci pribadinya untuk menghitung K; diberikan K, penerima kemudian dapat menentukan pesan terkompresi yang didekompresi untuk mengembalikan pesan asli M.

Sebuah modus enkripsi diperlukan pada langkah 4 (atas) untuk mengenkripsi CDPs lebih dari satu blok. PGP menggunakan variasi mode CFB Sebelum enkripsi, *plaintext* yaitu, CDP adalah *prepended* dengan *string* 10-oktet. 8 oktet pertama adalah acak, dan oktet 9 dan 10 adalah salinan dari oktet 7 dan 8. Data ini *prepended* berfungsi baik sebagai "memeriksa integritas" lemah dan sebagai vektor inisialisasi untuk cipher. Menunjukkan teks yang dihasilkan oleh R \, R2, Pi ,..., Id dimana R \ mewakili 8 oktet pertama *prepended* untuk CDP, R2 mewakili 2 oktet terakhir ("*octet* cek kunci") *prepended* untuk CDP, dan Pi, P2, ... , Id merupakan CDP itu sendiri. Enkripsi menggunakan mode CFB-seperti enkripsi kemudian hasil sebagai berikut: Enkripsi: Mengingat R1, R2, P1, P2, ... , Id, menghitung: Ci = Ri © EK (064) C2 = R2 © EK (C1) [0ji] (catatan: C2 dan R2 adalah 2 byte panjang) IV = Ci [2_7] o C2 C3 = Pi © EK (IV) untuk i = 2 sampai k: Ci +2 = Pi © EK (Ci + i) Output: C i, C2Ck +2

Dekripsi: Diberikan ciphertext Ci, C2 ,..., Ck +2, menghitung: Ri = Ci © EK (064) R2 = C2 © EK (Ci) [oji] (catatan: C2 dan R2 adalah 2 byte panjang) IV = Ci [2_7] o C2 Pi = C3 © EK (IV) untuk i = 2 sampai k: Pi = Ci +2 © EK (Ci + i) jika (== R2 Ri [6,7]) Output: P1, P2, P3 ..., Pk jika tidak *Error*

Tak terkompresi data,ketika data pesan tidak dikompresi oleh PGP sebelum enkripsi (misalnya, opsi kompresi dimatikan oleh pengguna), pesan terenkripsi hanya sebuah LDP dienkripsi. Perhatikan bahwa jika *plaintext* asli sudah dikompresi (yaitu, *plaintext* adalah file zip), ia akan diperlakukan sebagai data literal dan tidak akan kembali dikompresi oleh PGP. Menunjukkan di sini bahwa serangan dipilih dengan menggunakan *ciphertext* seperti yang dijelaskan dalam bagian sebelumnya tidak berhasil dalam memulihkan pesan *plaintext* dalam kasus ini. Pada tabel 2 merupakan pesan PGP tanpa kompresi. Dalam diagram, bagian yang diarsir merepresentasikan data dienkripsi. Angka-angka sepanjang diagram mewakili panjang (dalam *byte*) dari bidang yang sesuai. Sebuah ”nama” dan tanda "?" merupakan bidang panjang variabel (misalnya, panjang kolom "Nama" tergantung pada nilai yang terkandung dalam kolom "Nama Panjang").

CTB mempunyai ukuran data dalam byte berjenis *cipher*, PGP digunakan untuk membangun dan menentukan jenis paket yang sedang diproses. "Nama Panjang" adalah satuannya adalah *byte* tunggal yang menghalangi panjang dari bidang berikut "Nama"; bidang ini kedua berisi nama dari file *plaintext*. Perhatikan bahwa PGP tidak akan kembali lebih banyak data dari jumlah *byte* yang dijelaskan dalam kolom "Panjang" *header*, sementara memasukkan blok acak efektif menggandakan ukuran pesan. Oleh karena itu, implementasi langsung dari serangan itu akan membutuhkan musuh untuk memperoleh *decryption* dua *ciphertext* dalam rangka untuk memulihkan seluruh pesan (yaitu, musuh dapat memperoleh sekitar setengah *plaintext* dengan setiap *ciphertext* didekripsi diterima). Bergantian, musuh dapat mencoba untuk memodifikasi panjang *byte*. Panjang sebenarnya dari paket diketahui, sehingga dengan memanipulasi bit bidang panjang *byte* dalam data

dienkripsi musuh berpotensi dapat mengatur nilai panjang *cipherteks* dimodifikasi. Hal ini dimungkinkan, namun menyebabkan blok berikutnya data yang akan kacau tak terduga (sehingga mencegah musuh dari mempelajari pesan yang sesuai untuk memblokir). Selanjutnya, pendekatan ini akan efektif hanya jika panjang dari *ciphertext* dimodifikasi oleh *ciphertext* asli dalam kisaran yang sama, yaitu, panjang dijelaskan dengan jumlah yang sama *byte*. Jika tidak, serangan itu akan gagal karena musuh harus memasukkan byte ke dalam header dari sebuah paket terenkripsi PGP, yang akan menghasilkan data yang tak terduga.

Tabel 2 PGP Non-Kompresi

TB	ENGTH	PUBLIC KEY ENCRYPTED SESSION KEY DATA ?	TB	ENGTH	ANDOM‘ YTES	LAST TWO RANDOM BYTES 2	TB	ENGTH	ODE	AMELENGT	AME	STAMP	ATA	ATA	ATA	ATA	ATA	ATA
PKESKP PACKET					SEDP PACKET					LDP PACKET								

Sumber: Jallad,Kahil.et.all. 2002:7

Komplikasi lain yang kecil adalah bahwa PGP memeriksa nilai CTB setelah dekripsi. Jika nilai *byte* ini tidak valid, PGP akan keluar dengan pesan kesalahan. Dekripsi juga akan gagal jika "Mode" *byte* tidak diakui. Jadi, ketika membangun pesan *cipherteks* yang dipilih, musuh harus berhati-hati untuk tidak memutarbalikkan blok pertama dari pesan yang berisi header untuk paket data Literal. Hal ini diperhitungkan dalam tabel 3. dimana. , PGP akan membaca jumlah *byte* yang diberikan dalam kolom "Nama Panjang" dari kolom "Nama"; ini *byte* tidak akan muncul dalam teks keluaran biasa (digunakan untuk memperoleh nama untuk file yang akan berisi plaintext). Jika penyerang menyisipkan data dalam paket dienkripsi sebelum akhir nama file yang asli, atau jika nama file tidak berakhir pada batas blok, pesan didekripsi tidak akan benar sejajar dengan data acak penyerang. Hal ini terjadi karena awal dekripsi *ciphertext* yang dipilih akan mengandung bagian dari bidang nama file yang tidak biasanya output oleh PGP. Ini masalah kecil dapat dihindari dengan mengulangi *blok header* paket di *chosen ciphertext* dan kemudian menemukan keselarasan dengan menggeser pesan didekripsi,sehingga berulang kali mencoba menyerang sampai "Panjang kolom" (yang nilainya diketahui) ditemukan. Penyesuaian yang memungkinkan penentuan kolom "Panjang" juga memungkinkan sisa data yang akan ditentukan, dan tidak memilih *ciphertext* pesan tambahan yang diperlukan. Singkatnya, ketika pesan *plaintext* tidak dikompresi oleh PGP sebelum enkripsi atau ketika *plaintext* itu sendiri sebuah file terkompresi yang tidak lebih dikompresi oleh PGP, suatu *single-ciphertext* atau dua serangan *ciphertext* dapat digunakan untuk menentukan seluruh isi *original message*.

.

Tabel 3 Kompresi pesan PGP

PGP COMPRESSED MESSAGE									
TB	LENGTH	PUBLIC KEY ENCRYPTED SESSION KEY DATA	TB	LENGTH	RANDOM ' BYTES 8	LAST TWO RANDOM BYTES	CTB	ALG	COMPRESED
	1	?		1		2	1	1	DATA

Sumber: Jallad,Kahil.et.all. 2002:8

Modifikasi yang diusulkan memberikan keuntungan sebagai berikut dalam skema PGP:

1. Mengurangi jumlah kunci yang dibutuhkan untuk pesan transmisi *e-mail* aman dan *overhead* yang sesuai, maka akan meningkatkan kecepatan pemrosesan dari PGP pada pengirim pesan *e-mail* dan sisi penerima.
2. Pesan dienkripsi menggunakan kunci grup yang bisa dikirimkan ke beberapa PGP menggunakan kunci grup, di mana kunci kelompok dihasilkan menggunakan kelompok kunci. Dimana modifikasi pertama mengusulkan penggunaan kelompok kunci untuk mengenkripsi kunci sesi yang menghindari keterbatasan PGP untuk mengirim pesan yang dienkripsi yang sama untuk modifikasi beberapa pengguna *(receipients)*. Modifikasi kedua mengusulkan penggunaan kunci kelompok bukan kunci pribadi pengirim untuk generasi tanda tangan digital yang menghindari penggunaan kriptografi kunci publik di modifikasi PGP, baik untuk meningkatkan kecepatan pengolahan PGP.

PGP dan kriptografi tentu memiliki beberapa keterbatasan, berikut ini adalah ancaman yang harus diperhatikan oleh pengguna PGP:

1. *Dictionary attack*
   Seringkali pengguna PGP memilih *password* untuk melakukan enkripsi kunci privat yang mudah ditebak, seperti nama kerabat, tanggal lahir, nomor telepon, atau kata umum lainnya. Seorang hacker dapat dengan mudah menebak *password* tersebut dengan cara melakukan *dictionary attack,* yaitu menebak *password* dengan cara mencoba semua kata yang umum digunakan oleh pengguna PGP. Oleh karena itu, dibutuhkan *password* yang sulit ditebak seperti gabungan karakter antara alfabet dan numerik.
2. Penghapusan dokumen secara tidak "bersih",Proses menghapus dokumen dari komputer pribadi tidaklah semudah yang dikira. Sistem operasi hanya melakukan penghapusan indeks *file* dari *hard disk,* sedangkan *file* tersebut sebenarnya masih terdapat di sana. Seseorang dapat melakukan pemulihan *(recovery)* terhadap *file* tersebut dengan menggunakan berbagai aplikasi tertentu, salah satunya dengan Norton *Disk Doctor*. PGP menyediakan subrutin untuk melakukan *overwrite* tempat kosong pada *hard disk* sehingga *file-file* yang sudah dihapus tidak dapat di-*recover* lagi.
3. Virus dan *trojan horse*
   Seseorang dapat menyebarkan virus yang mencatat semua tombol *keyboard* yang ditekan oleh pengguna, program seperti ini biasa disebut *keylogger*. *Keylogger* dapat digunakan untuk mencuri *password* kunci privat pengguna PGP. Selain itu, terdapat pula program PGP palsu yang berupa *trojan horse* dan dapat menyebarkan kunci rahasia pengguna. Oleh karena itu, pengguna perlu melengkapi komputer

pribadinya dengan program *antivirus* dan juga *firewall.*

4. Ancaman keamanan fisik
   Selain ancaman pada sistem komputer, ancaman fisik seperti pencurian komputer dan data juga perlu diperhatikan. Oleh karena itu diperlukan juga keamanan secara fisik pada komputer milik pribadi dan juga perusahaan.
5. Analisis j aringan
   PGP dapat mencegah seseorang membaca pesan yang dikirimkan, tetapi tidak dapat mencegah seseorang yang melakukan *monitoring* terhadap jaringan komputer. Serangan seperti ini dilakukan dengan menangkap paket-paket data yang dikirim melalui jaringan kemudian melakukan analisis terhadap paket-paket data tersebut.
6. TEMPEST
   TEMPEST merupakan metode untuk mengumpulkan informasi rahasia dengan cara "mendengarkan" pada radiasi yang dipancarkan peralatan elektronik. Pada tahun 1960-an, perlengkapan militer dilengkapi alat untuk melindungi jenis serangan seperti ini. PGP juga menyediakan fitur untuk memilih jenis huruf khusus dengan ujung yang *smooth* dan karakter yang tidak biasa agar radiasi yang dipancarkan melalui monitor komputer tidak terdeteksi oleh serangan

V. KESIMPULAN

Dari penelitian diatas, dapat disimpulkan beberapa aspek tentang teknik enkripsi dalam pengiriman e-mail dengan mempergunakan PGP sebagai berikut :

1. Enkripsi dengan mempergunakan PGP kompresi, menghasilkan suatu hasil enkripsi yang relatif aman, sehingga data rahasia yang akan dikirim melalui e-mail akan terproteksi dari orang yang tidak bertanggung jawab yaitu *hacker* maupun *cracker*.
2. Teknik untuk enkripsi dan dekripsi mempergunakan bantuan file teks eksternal. Apabila pada waktu yang bersamaan terjadi proses enkripsi dan dekripsi lebih dari satu, perlu ada pengembangan lebih lanjut mengenai nama file yang dipergunakan, sehingga tidak terjadi pertukaran data.
3. Salah satu kelebihan PGP adalah enkripsi dengan mempergunakan kunci *public*. Sehingga perlu adanya pengembangan lebih lanjut tentang teknik enkripsi yang mempergunakan public dan *private key*. Dimana *private key* akan dipegang pemilik *record/account* tersebut.

DAFTAR PUSTAKA

Kurniawan,Yusuf . 2004, Kriptografi keamanan internet dan jaringan komunikasi penerbit Informatika, Bandung

Raharjo, Budi.2005.Keamanan Sistem Informasi Berbasis Internet. Jakarta :PT Indocisc dan Bandung: PT Insan Infonesia

Simarmata,Janner.2006.Pengamanan Sistem Komputer.Yogyakarta:Andi

Wahana komputer. 2005. Menjadi Administrator Jaringan Komputer. Yogyakarta:Andi

Referensi Web

Jiang Qinglin, Douglas S. Reeves ,et.all.2004. Improving Robustness of PGP Keyrings by Conflict detectional (http://reeves.csc.ncsu.edu/papers-and-other-stuff/2004-02-rsa-ct-robust-keyrings-paper.pdf) (diakses pada tanggal 20 januari 2011).

Jallad,Kahil.et.all. 2002. Implementation of Chosen - Ciphertext Attacks against PGP and GnuPG. http://www.cs.umd.edu/~jkatz/papers/pgp-attack.pdf (diakses pada tanggal 23 juli 2010)

Moha., Abdul Wahab 2004. Pretty Good Privacy (Pgp) Untuk Keamanan E-Mail.(http://www.cert.or.id/~budi/courses/ec7010/dikmenjur-2004/abdmoha-report.pdf) (diakses pada tanggal 23 Juli 2010).

Tilman Linneweh. 2002. Using PGP / GnuPG and S / MIME with Email (http://stud3.tuwien.ac.at/~e0025974/uni/crypto.pdf) (diakses pada tanggal 23 Juli 2010)

Tripathi,Sachin,G.PBiswas.2010. Secure E-Mail Messaging to Selected Group Members Using PGP Technique.http://www.ijcaonline.org/jo

urnal/number10/pxc387366.pdf(diakses pada tanggal 20 januari 2011)

Tanoto,Andri. 2006. Analisis Keamanan pada *Pretty Good Privacy* (PGP) (http://www.informatika.org/~rinaldi/Kriptografi/2006-2007/Makalah2/Makalah-022.pdf (Diakses pada tanggal 22 Juli 2010).