Sistem Keamanan "Pesan" Berbasis *Pretty Good Privacy*

[1]Matius Irsan Kasau, [2]ST. Aminah Dinayati Ghani, [3]Rudy Donny Liklikwatil
[1]Dosen LLDIKTI IX DPK pada Universitas Dipa Makassar, [2,3]Universitas Dipa Makassar
Program Studi Teknik Informatika, Universitas Dipa, Makassar
Jalan Perintis Kemerdekaan KM. 9 Makassar, Telp. (0411)587194 – Fax. (0411) 588284
e-mail: [1]irsan.kasau@dipanegara.ac.id, [2]dinayati.amy@dipanegara.ac.id, [3]rudy.donny@dipanegara.ac.id

Abstrak

Tulisan ini akan memaparkan secara sederhana dan rinci mengenai metode *Pretty Good Privacy* (PGP) sebagai salah satu implementasi keamanan jaringan untuk pengiriman pesan yang sangat sulit diterobos. Pengiriman pesan melalui jaringan ke suatu tujuan harus dipastikan aman terhadap upaya penyadapan dan sabotase dari pihak yang tidak berhak selama pesan itu mengalir pada jaringan. Salah satu metode yang cukup handal yang biasa digunakan adalah *Pretty Good Privacy* (PGP). Metode ini menerapkan kunci asymetris ditambah sebuah kunci sessi. Pertama-tama pesan asli (*plaintext*) dienkripsi dengan kunci sessi, lalu *ciphertext* hasil enkripsi yang diperoleh dienkripsi lebih lanjut dengan menggunakan *public key* orang yang akan dikirimkan pesan. Sesampai pada penerima, ciphertext yang dua kali dienkripsi ini dapat dibuka dengan melakukan dekripsi menggunakan kunci private penerima. Hasilnya adalah sebuah *ciphertext* yang masih terkunci oleh kunci sessi pengirim. Akibatnya, penerima tidak dapat berbuat apa apa sebelum pengirim memberikan kunci sessinya. Oleh karena itu langkah selanjutnya adalah pengirim mengenkripsi kunci sessi itu dengan p*ublic key* penerima, lalu mengirim ke penerima. Tentu saja penerima dengan mudah membuka dan memperoleh kunci sessi itu sebab ia memiliki kunci privatenya. Dengan demikian *ciphertext* yang menyembunyikan pesan asli dapat dibuka oleh penerima menggunakan kunci sessi tersebut. Dari contoh kasus yang dijadikan illustrasi, terlihat bahwa betapa sulitnya mengetahui hubungan antara *plaintext* dengan c*iphertext* yang terjadi dari kunci sessi dan *public key*.

Kata kunci : **GP, enkripsi/deskripsi, *plaintext/ciphertext*, public key /kunci rahasia, kunci sessi.**

Abstract

This paper will explain in a simple and detailed manner the PGP method as an implementation of network security for sending messages that are very difficult to break. The delivery of messages through the network to a destination must be ensured that it is safe against interception and sabotage attempts from unauthorized parties as long as the message flows on the network. One fairly reliable method that is commonly used is Pretty Good Privacy (PGP). This method applies an asymmetric key plus a session key. First, the original message (plaintext) is encrypted with the session key, then the encrypted ciphertext is further encrypted using the public key of the person to be sent the message. Arriving at the recipient, this double encrypted ciphertext can be opened by decrypting it using the receiver's private key. The result is a ciphertext that is still locked by the sender's session key. As a result, the recipient is unable to do anything before the sender provides the session key. Therefore the next step is the sender to encrypt the session key with the recipient's public key, then send it to the recipient. Of course the recipient can easily open and get the key to the session because he / she has the private key. Thus the ciphertext that hides the original message can be opened by the recipient using the session key. From the examples of cases that are used as illustrations, it can be seen that how difficult it is to know the relationship between plaintext and ciphertext that occurs from the session key and the public key.

Keywords : ***PGP, encryption / description, plaintext / ciphertext, public key / secret key, session key***.

1. PENDAHULUAN

Pengiriman pesan baik berupa *file, e-mail* atau yang lainnya dari seorang ke seorang melalui jaringan dapat mengalami berbagai masalah seperti terinterupsi, termodifikasi, terintersepsi, terfabrikasi dari aliran normalnya oleh penyusup jaringan yang melakukan sabotase dan penyadapan sebelum pesan itu sampai pada tujuan.

Perlu dijelaskan bahwa keamanan komputer memiliki syarat utama terhadap data pesan yang ditransmisikan yaitu kerahasiaan (secrecy), integritas (integrity), dan ketersediaan (availability). Ketersediaan diperlukan agar data hanya bisa diakses dan dibaca oleh pihak pihak yang berwenang atau memiliki hak akses yang terotorisasi terhadap data pesan. Jenis hak akses yang dimaksud meliputi: mencetak, membaca, dan berbagai bentuk akses lainnya, seperti membuka suatu obyek. Integritas diperlukan agar data pesan hanya dapat dimodifikasi oleh pihak pihak yang berwenang. Modifikasi yang dimaksud meliputi menulis, mengubah, mengubah status, menghapus, membuat baru, menambah atau mengurangi isi data pesan. Sedangkan ketersediaan diperlukan agar data data selalu tersedia untuk pihak pihak yang berwenang.

Tipe-tipe ancaman terhadap ancaman keamanan sistem komputer dapat dimodelkan dengan memandang sistem komputer sebagai penyedia sistem informasi. Berdasarkan fungsi ini, ancaman terhadap sistem komputer dikategorikan menjadi empat kategori seperti yang disebutkan tadi. Adapun Interupsi (*interruption*) adalah sabotase yang berupaya untuk menghancurkan sumber daya yang berakibat pada ketidak tersediaan data atau tidak berguna. Interupsi merupakan ancaman terhadap syarat ketersediaan. Sebagai contoh dari interupsi adalah; penghancuran bagian perangkat keras seperti hardisk, ketidak berfungsian alat proses, disk atau tape yang tidak terbaca, pemotongan kabel telekomunikasi. Intersepsi merupakan pihak tidak terotorisasi dapat mengakses sumber daya tanpa hak. Pihak tidak terotorisasi dapat berupa orang atau program komputer. Interupsi merupakan ancaman terhadap kerahasiaan. Sebagai contoh; penyadapan untuk mengambil data rahasia, mengkopy file tanpa otorisasi. Modifikasi merupakan pihak tidak terotorisasi tidak hanya mengakses tetapi juga merusak sumber daya. Modifikasi merupakan ancaman terhadap integritas. Sebagai contoh; mengubah nilai file data, mengubah program sehingga bertindak secara berbeda, memodifikasi pesan pesan yang ditransmisikan pada jaringan. Sedangkan Fabrikasi merupakan pihak tidak terotorisasi menyisipkan atau memasukkan obyek obyek palsu ke dalam sistem. Fabrikasi merupakan ancaman terhadap integritas. Sebagai contoh; memasukkan pesan pesan palsu ke jaringan, menambah record ke dalam file.

Masalah utama pada keamanan komputer selain kehilangan data (*data loss*) adalah penyusup atau instruder yang terbagi atas penyusup passif dan penyusup aktif. Adapun penyusup passif terjadi dalam *eavesdropping* atau memantau pengtranmissian. Tujuan penyusupan atau penyerangan adalah agar bisa memperoleh informasi yang sedang ditransmisikan. Dua jenis serangan pasif adalah membuka isi pesan dan analisis lalu lintas data. Membuka isi pesan bisa dipahami dengan mudah. Percakapan telepon, pesan electronik mail, dan file yang ditransfer kemungkinan berisi informasi informasi yang sensitif dan rahasia. Sedangkan analisais lalulintas, lebih halus. Anggap saja seseorang memiliki suatu cara untuk menutup isi pesan-pesan atau lalulintas data atau informasi lainnya sehingga penyerang, sekalipun mereka berhasil memperoleh pesan pesan tersebut, namun tetap tidak dapat mengutip informasi dari pesan tersebut. Penyerang atau penyusup juga dapat menentukan lokasi dan mengidentifikasikan pihak pihak yang melakukan komunikasi serta mampu mengamati frekuensi dan panjang pesan yang sedang dikomunikasikan. Informasi ini sangat berguna untuk memperkirakan kondisi komunikasi yang sedang dilakukan. Serangan passif sangat sulit dideteksi karena tidak melibatkan pengubahan data. Namun bisa dicegah agar penyusup ini tidak berhasil. Jadi penekanannya lebih pada pencegahan daripada pendeteksian.

Penyusup aktif melibatkan beberapa modifikasi aliran data atau informasi atau menciptakan aliran data yang menyesatkan yang terbagi dalam empat kategori yaitu penyamaran, jawaban, modifikasi pesan, dan penolakan layanan. Penyamaran dilakukan bila penyusup berpura-pura sebagai entitas yang berbeda. Serangan penyamaran biasanya mencakup salah satu dari beberapa bentuk serangan aktif. Sebagai contoh, rangkaian asli ditangkap dan dibalas setelah setelah rangkaian asli yang valid diganti, sehingga memungkinkan entitas yang berwenang dengan sedikit hak khusus memperoleh ekstra keistimewaan dengan cara menirukan entitas entitas yang memiliki keistimewaan tersebut. Jawaban melibatkan penangkapan secara passif suatu unit data berikut transmissi ulangnya berturut turut untuk mendapatkan efek yang tidak terotorisasi. Modifikasi pesan berarti beberapa bagian dari pesan asli diubah, atau pesan pesan tersebut ditunda, agar menghasilkan efek yang tidak terotorisasi. Sebagai contoh sebuah pesan yang artinya “Mengizinkan Muhammad Arsyad bin Abdullah membaca file rekening rahasia” dimodifikasi

dengan mengganti nama menjadi “mengizinkan Rudy Donny Likliwatil membaca file rekening rahasia”. Penolakan layanan yaitu mencegah atau menunjukkan penggunaan normal atau manajemen fasilitas fasilitas komunikasi. Serangan ini kemungkinan memiliki tujuan tertentu, misalnya penyusupan atau entitas bisa bisa menahan semua pesan yang menuju secara langsung ke suatu tujuan tertentu, misalnya layanan audit rahasia. Bentuk penolakan layanan lainnya ialah gangguan jaringan secara keseluruhan, baik dengan cara melumpuhkan jaringan atau dengan cara memenuhi jaringan dengan pesan pesan sehingga mengurangi kinerjanya. Serangan atau penyusup aktif menampilkan karakteristik yang berlawan dari serangan passif. Meskipun, serangan atau penyusup sulit untu dideteksi, ukuran-ukuran yang tersedia untuk mencegah kelancarannya. Sebaliknya, benar benar sulit mencegah serangan penyusup aktif secara mutlak, karena untuk melakukannya membutuhkan perlindungan keseluruhan fasilitas komunikasi dan jalur jalurnya dari segi fisik setiap saat. Karenanya, tujuannya hanya untuk mendeteksi serangan sekaligus perbaikan terhadap gangguan atau penundaan yang ditimbulkan. Adapun kategori penyusupan adalah misalnya: lirikan mata pemakai ninteknis, penyadapan oleh orang dalam, usaha hacker dalam mencari uang, spionase militer atau bisnis.

Saltzer dan Schoolder (1975) memberi petunjuk mengenai prinsip-prinsip pengamanan sistem komputer yakni: rancangan sistem harusnya publik, dapat diterima, pemeriksaan otoritas saat itu, kewenangan serendah mungkin, dan mekanisme yang ekonomis. Keamanan sistem seharusnya tidak tergantung pada kerahasiaan rancangan mekanisme pengamanan. Mengasumsikan penyusup tidak akan mengetahui cara kerja dari sistem pengamanan hanya menipu atau memperdaya perancangan sehingga tidak membuat mekanisme proteksi yang bagus. Dapat diterima yaitu, skema yang dipilihharus dapat diterima secara psikologis. Mekanisme proteksi seharusnya tidak mengganggu kerja pemakai dan memenuhi kebutuhan otorisasi pengaksesan. Jika mekanisme tidak mudah digunakan, maka tidak akan pernah digunakan secara tidak benar. Pemeriksaan otorisasi saat itu yaitu sistem seharusnya memeriksa izin dan menyatakan pengaksesan diizinkan, serta kemudian menetapkan terus informasi ini untuk penggunaan selanjutnya. Banyak sistem memeriksa izin ketika file dibuka dan setelah itu tidak diperiksa. Pemakai yang membuka file dan lupa menutup file akan terus dapat diakses waktu pemilik fie telah mengubah atribut proteksi file. Kewenangan serendah mungkin yaitu program atau pemakai sistem seharusnya beroperasi dengan kumpulan wewenang serendah mungkin yang diperlukan untuk menyelesaikan tugasnya. Default sistem yang digunakan harus tidak ada akses sama sekali. Terakhir mekanisme yang ekonomis yaitu proteksi seharusnya kecil, sesederhana mungkin dan seragam sehingga memudahkan verifikasi. Proteksi seharusnya dibangun dilapisan terbawah. Proteksi merupakan bagian integral rancangan sistem, bukan mekanisme yang ditambahkan pada rancangan yang telah ada.

Salah satu cara tindakan pengamanan pesan terhadap upaya sabotase dan penyadapan adalah dengan menggunakan PGP *(Pretty Good Privacy)* yang merupakan salah satu implementasi dari teknik enkripsi-dekripsi dalam sebuah program. PGP dapat digunakan untuk mengkodekan suatu pesan sehingga hanya orang yang dituju atau pemilik pesan yang bisa mendeskripsikan dan membuat tanda tangan digital. Dengan demikian penerima pesan dapat meyakini bahwa pesan yang diterimanya benar benar berasal dari pengirim yang diharapkan, bukan hasil sabotase atau penyadapan dari seorang penyusup.

Berbeda dengan teknik enkripsi yang konvensional, PGP menggunakan tiga buah kunci untuk melakukan proses enkripsi dan dekripsi, dua diantaranya kunci asymetris dan satu lainnya untuk kunci penutup dan pembuka pesan. Implementasi PGP dapat dilakuka n pada berbagai jenis komputer dan sistem operasi yang digunakan seperti: *MS-DOS, UNIX, Machintosh, Windows* dan lainnya. Kunci Asymetris yang digunakan pada PGP didasarkan pada algoritma RSA atau *algoritma Diffie Hellman*. Dengan kedua algoritma ini PGP menjadi *software* enkripsi deskripsi yang kuat dan handal. Melalui proses enkripsi pesan diacak sehingga menjadi tidak terbaca oleh pihak penyusup. Kemudian hanya orang yang berhak yang memiliki pasangan kunci asymetris dan kunci tutup buka pesan yang terbungkus dalam public key yang dapat mengdeskripsinya ke bentuk pesan aslinya.

2. Landasan Teori

Pretty Good Privacy (PGP) adalah suatu metode program enkripsi pesan atau informasi yang memiliki tingkat keamanan yang sangat hadal, bersifat rahasia dengan menggunakan kunci asymetris yaitu sepasang kunci yang berbeda, satu bersifat bukan rahasia *(public)* dan satu bersifat rahasia *(private)*. Dalam mekanisme kerjanya, sebuah kunci yang lain yakni kunci tutup buka dienkrip secara terpisah dengan enkrip pesan sehingga seorang penyusup akan merasa sangat kesulitan untuk dapat melakukan sabotase atau penyadapan terhadap pesan terkode selama mengalir di jaringan. Sistem keamanan PGP pertama kali dikembangkan oleh Phill Zimmermann pada akhir tahun 1980. Pada mulanya PGP digunakan untuk melindungi e-mail (surat elektronik) dengan memberikan perlindungan kerahasiaan

secara enkripsi dan autentikasi dengan tanda tangan digital. Untuk itu Phill Zimmermann membuat sebuah program yang digunakan agar dapat melindungi pesan dengan kerahasiaan. Program yang dibuat oleh Phill Zimmermann memiliki dua versi yaitu "*USA Version*" dan "International Version". PGP Versi USA hanya dapat digunakan di wilayah USA dan oleh warga Negara USA saja. PGP Versi USA ini menggunakan algoritma RSA (yang telah menjadi hak paten) dalam enkripsinya. Sementara Versi International menggunakan algoritma MPILIB yang diciptakan khusus oleh Phill Zimmermann sendiri. PGP Internasional bisa digunakan oleh seluruh dunia.

Dasar dasar dari PGP dibuat berdasarkan konsep *Private Key Cryptography* sebagai dasar otorisasi. Kunci ini digunakan untuk mengenkripsi dalam suatu hubungan komunikasi antara dua mesin. Untuk menjaga kerahasiaan data, kriptografi mentransformasikan pesan plaintext ke dalam bentuk terkodekan atau sandi yang disebut *ciphertext* yang tidak dapat dikenali. *Ciphertext* inilah yang kemudian dikirimkan oleh pengirim *(sender)* kepada penerima *(receiver*). Pada saat membuat kunci, PGP akan menciptakan dua buah kunci yaitu *private key* (kunci rahasia) dan *public key* (bukan rahasia, sengaja diumumkan untuk diketahui orang banyak). Seseorang yang ingin mengirim pesan kepada seorang yang lain, maka ia terlebih dahulu harus mencari pada suatu situs *public key* orang itu. *Public key* itulah yang digunakan untuk mengenkripsi pesan. Karena hanya orang yang dialamatkan pesan tersandi itu yang memiliki *private key* untuk membukanya, maka selama pesan *ciphertext* mengalir pada jaringan, tidak seorangpun penyusup dapat membukanya, selain pemilik *public key* dan *private key* sendiri. Salah satu metode dalam menciptakan pasangan *public key* dan *private key* adalah metode RSA (Rivers, Shamir, Adleman) dengan prinsip dasar sebagai berikut:

- Memilih 2 buah bilangan prima besar, p dan q (biasanya lebih besar dari 10100)
- Menghitung n = p x q dan z = (p – 1)x(q – 1)
- Memilih sebuah bilangan d (private key) yang secara relative prima terhadap z
- Mencari nilai e (public key) sedemikian rupa sehingga e x d = 1 mod z = 1 + m x z, dengan m adalah bilangan bulat 0, 1, 2, 3, …. Terdapat harga m yang dapat menyebabkan harga e dan harga d sebagai pasangan public key berharga bulat.

Untuk sekedar contoh illustrasi, marilah memilih 2 buah bilangan prima besar p = 3 dan q = 11. Secara berturut turut diperoleh n = p x q = 3 x 11 = 33 dan z = (p – 1)x(q – 1) = 2 x 10 = 20. Selanjutnya pilih d = 7 (boleh pilih yang lain seperti 3, tetapi tidak boleh 5 sebab 5 mempunyai factor persekutuan dengan 20, artinya 5 tidak relatif prima terhadap 20), diperoleh e = (1 + m x z)/d = (1 + 20 m)/7. Untuk m = 1 diperoleh e = 3. Jadi telah diperoleh *private key* d = 7 dan *public key* e = 3. Rumus enkripsi dan dekripsinya adalah : $C = P^e mod\ n$ untuk enkripsi dan $P = C^d mod\ n$ untuk deskripsi. Sekarang ambil plaint text P = E (urutan abjad ke 5) diperoleh *ciphertext* $C = P^e mod\ n = 5^3 mod\ 33 = 125\ mod\ 33 = 26\ atau\ Z$. Pada ujung penerima akan memperoleh plaint text P = E kembali dengan *private key* d = 7 dengan cara $P = C^d mod\ n = Z^d mod\ n = 26^7 mod\ 33 = 803181076\ mod\ 33 = 5\ atau\ E$. Untuk pembahasan lebih lanjut dalam tulisan ini akan digunakan pasangan kunci yang sudah dihitung tersebut yakni *private key* atau kunci rahasia $K_{RB} = d = 7$ dan *public key* $K_{PB} = e = 3$. Index B menunjukkan penerima pesan.

3. METODE PENELITIAN

Tulisan ini menjelaskan metode mengenkripsi dan deskripsi PGP dengan dua cara yaitu melalui windows explorer dan melalui program PGP Mail. Caranya adalah sebagai berikut:

1. Melalui windows explorer.
   - Pertama, pilih file yang akan dienkripsi, klik kanan mouse dan pilih menu PGP.
   - Selanjutnya, dari menu PGP pilih Encrypt sehingga muncul sebuah tampilan.
   - Pada tampilan itu pilih Resipents yakni orang yang dikirimi file. Tekan OK, maka file sudah terenkripsi dengan public key penerima.
   - Hasil enkripsi dapat dipilih untuk disimpan sebagai arsip teks dengan ekstensi nama arsip.asc. Atau sebagai arsip biner dengan ekstensi nama arsip.pgp.
   - Untuk sebaliknya, mendeskripsi arsip, klik arsip yang terenkripsi kemudian klik kanan mouse dan pilih Decrypt.
   - Program PGP meminta untuk memasukkan passphrase untuk private key (harus sama dengan passphrase yang diisikan pada waktu pembangkitan pasangan kunci)

- Jika passphrase benar, maka file akan didekripsi dengan menggunakan private key yang berkorespondensi dengan kunci publiknya.

2. Melalui program PGP Mail
   - Aktifkan program PGP Mail, sehingga muncul tampilan yang berisi sejumlah pilihan.
   - Pilih ikon surat + gembok.
   - Lanjutkan tahapan enkripsi sama seperti melalui *windows explorer* diatas.

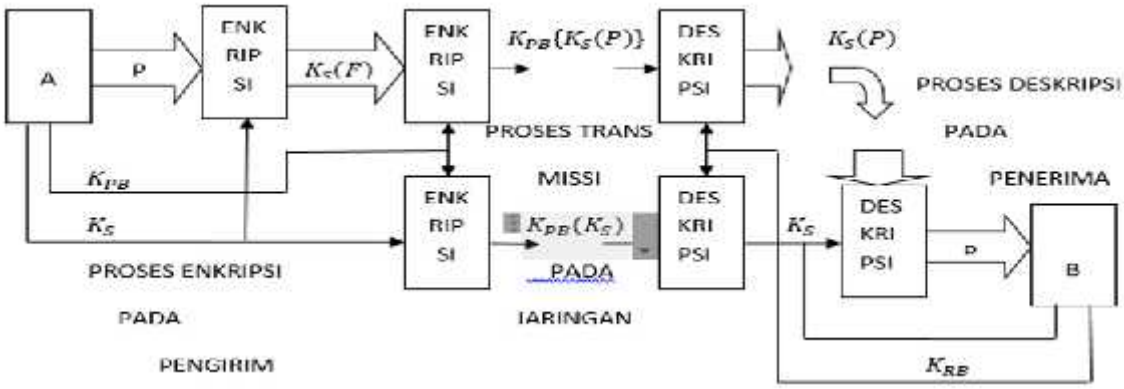
   Beberapa perintah dalam PGP :
   - "pgp –h" perintah untuk melihat keseluruhan argument yang tersedia dalam PGP
   - "pgp –kvc" perintah untuk membuka koleksi public key.
   - "pgp –kxa userid keyfile" perintah untuk mengekstaksi public key sehingga bisa dibagikan kepada orang lain. File yang berisi public key adalah "pubkey.asc".
   - "pgp –ka keyfile" perintah untuk menambahkan public key yang diterima ke dalam koleksi public yang sudah dimiliki.
   - "pgp –ks userid" perintah untuk menandai atau memberi tanda pada sebuah kunci.

   Sementara untuk membuat pasangan kunci pada PGP, secara sederhana dilakukan tahapan tahapan berikut :
   - Aktifkan PGPkey sehingga muncul sebuah tampilan
   - Pada tampilan yang muncul itu pilih Keys ➡ New Key
   - Selanjutnya akan ditampilkan wizard untuk membangkitkan pasangan kunci. Isilah beberapa isian yang disediakan.
   - Untuk melihat public key, atau member public key kepada orang lain, eksport kunci tersebut ke arsip (ekstensi arsip adalah .asc).
   - Public key orang lain dapat dimasukkan ke dalam daftar kunci dengan cara memilih menu *Key* ➡ *Import*.

Jika seseorang melakukan enkripsi dekripsi PGP sesuai dengan s*oftware* yang tersedia dengan mengikuti langkah langkah diatas, maka orang itu akan tercengang dan kagum melihat hasil enkripsi yang diperolehnya, tanpa mengetahui proses kerjanya. Untuk mengubah rasa kagum dan ketercengangan dari orang yang hanya mengetahui secara praktis saja, alangkah baiknya diungkapkan latar belakang teoritisnya sehingga seluruh proses yang terjadi dalam PGP dapat diketahui dengan jelas. Seseorang yang mengetahui latar belakang teoritis dari suatu masalah, akan menganggap masalah itu biasa biasa saja dan tidak lagi merasa tercengang dan kagum berlebihan pada masalah itu. Atas pertimbangan ini, penulis akan lebih menekankan pada penjelasan aspek teoritisnya dengan illustrasi praktis yang mudah dimengerti.

4. HASIL DAN PEMBAHASAN

Untuk dapat mengikuti dan mengerti proses kerja sistem keamanan pengiriman pesan berbasis *pretty good privacy (PGP*), alangkah baiknya terlebih dahulu digambarkan konsep dasar dan proses enkripsi dekripsinya. Perhatikan pada gambar 1 bahwa terdapat tiga kali enkripsi pada sisi pengirim. Demikian pula terdapat tiga kali dekripsi pada sisi penerima. Satu demi satu dari ketiga proses enkripsi dan ketiga proses dekripsi itu akan dijelaskan secara detail sebagai berikut :



Gambar 1. Blok diagram lengkap PGP

Perhatikan gambar, cara kerja dari PGP ini dapat dijelaskan sebagai berikut :

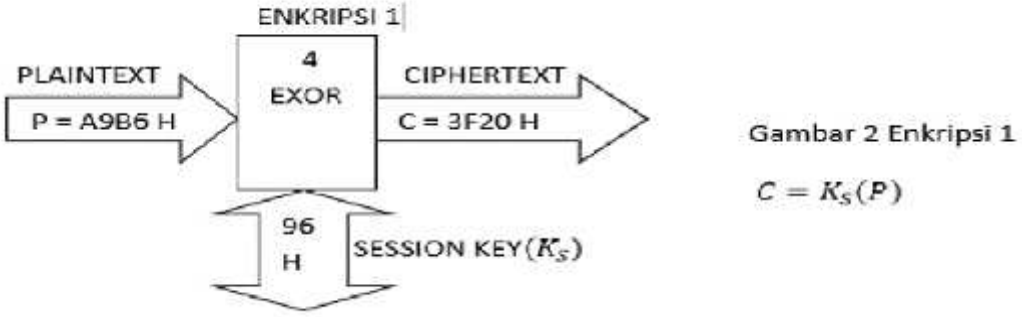
1. Pertama tama A mengenkripsi pesan F nya dengan menggunakan kunci sessi K_S, melakhirkan pesan tersandi dalam bentuk *ciphertext* $K_S(F)$. Tidak ada pihak lain yang dapat membuka pesan *ciphertext* ini, sebab K_S merupakan kunci rahasia yang hanya diketahui oleh A.
2. Kemudian A mengambil *public key* B pada web site yakni K_{PB} dan mengenkripsi sekali lagi ciphertext $K_S(F)$ menjadi $K_{PB}\{K_S(F)\}$ yang dikirim melintasi jaringan. Pesan yang terkunci sebanyak dua kali ini tentu saja sangat aman, sebab seorang penyusup jaringan haruslah mengetahui dua kunci rahasia yakni K_S yang dirahasiakan oleh A dan K_{RB} yang dirahasiakan oleh B baru dapat melakukan sabotase atau perekayasa pesan F yang terkunci dua kali itu.
3. Sekarang perhatikan, ketika pesan dobel ciphertext itu tiba atau diterima oleh B, maka B mengambil kunci rahasianya yakni K_{RB} dan melakukan deskripsi pertama $K_{RB}\{K_{PB}[K_S(F)]\} = K_S(F)$.
4. Sampai disini tentu saja B kesulitan untuk membuka $K_S(F)$ sebab dia terkendala tidak mengetahui kunci rahasia K_S yang dirahasiakan oleh A, sehingga tidak dapat tidak haruslah A mengirimkan kepadanya kunci rahasia K_S tersebut, tentu saja kunci rahasia ini harus dikirim dalam keadaan terkunci. Jadi kunci dikunci, namun kunci yang digunakan untuk mengunci atau mengenkripsi K_S ini adalah juga *public key* B yakni K_{PB}, mengalirlah *ciphertext* kunci pada jaringan $K_{PB}(K_S)$. Sekali lagi tentu saja hanyalah B yang dapat melakukan deskripsi terhadap *ciphertext* kunci terkunci ini dengan menggunakan kunci rahasianya yakni K_{RB}, diperoleh $K_{RB}\{K_{PB}(K_S)\} = K_S$
5. Dan lihatlah, kesulitan B untuk membuka $K_S(F)$ kini sudah menjadi sangat mudah, sebab kunci yang tadinya tidak diketahuinya telah diterimanya dari A. Dengan menggunakan kunci ini, maka B segera dapat melihat pesan dari A yakni $K_S\{K_S(F)\} = F$. Pesan F dari A pun diterima oleh B dengan sangat aman dalam proses pengirimannya. Itulah proses dari pengiriman pesan dengan metode PGP.

Dalam tulisan ini akan diperlihatkan secara bertahap dan sangat rinci proses kerja PGP. Para pembaca diharapkan akan dapat memahami dengan mudah melalui illustrasi menggunakan kunci sessi, *public key* dan kunci rahasia yang sederhana, namun *ciphertext* yang dihasilkannya begitu sangat sulit untuk dipecahkan. Ada sebanyak tiga kali enkripsi dilakukan pada sisi pengirim. Demikian juga ada sebanyak tiga kali deskripsi dilakukan pada sisi penerima. Penjelasan untuk masing masing enkripsi deskripsi tersebut akan dijelaskan sebagai berikut :

1. Anggaplah digunakan kunci sessi rahasia $K_S = 96\,H$ (biner 10010110) dengan metode "*one time pad*" untuk mengenkripsi pesan $F = A9B6\,H$ (biner 1010100110110110). Hasil enkripsi EXORnya dalam metode *"one time pad"* adalah:

$$F = A9B6\,H = 1010100110110110\,B$$
$$K_S = 9696\,H = 1001011010010110\,B$$
$$\text{EXOR}$$
$$0011111100100000\,B$$
$$3 \quad F \quad 2 \quad 0 \quad H$$

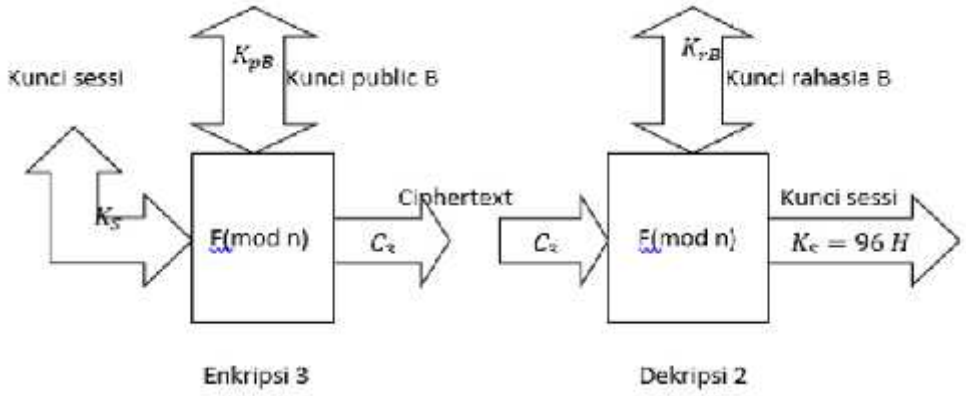
Jadi pesan F = A9B6 H dienkripsi dengan kunci K_S = 96 H (didobel), menghasilkan *ciphertext* C = 3F20 H. Tidak seorangpun yang dapat mengetahui asal dari C = 3F20 H ini kalau tidak diberikan kuncinya yakni $K_S = 96$ tersebut.



Gambar 2. Proses Enkripsi dengan kunci sessi

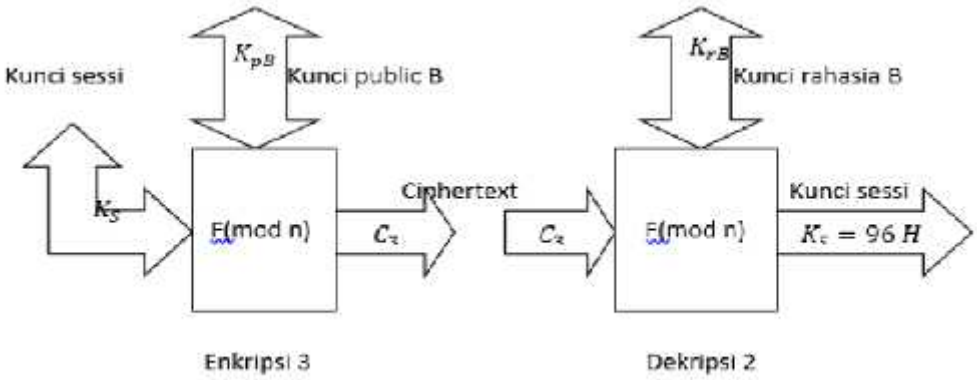
2. *Ciphertext C* = 3F20 H ini dienkripsi lebih lanjut menggunakan *public key* dari orang yang akan dikirimkan pesan P. Anggaplah menggunakan metode RSA untuk menentukan pasangan *public key* dan *private* (rahasia) dari orang yang akan menerima pesan P.

   Caranya adalah :
   - Ambil 2 bilangan prima, misalnya p = 3 dan q = 11.
   - Cari hasil kalinya dipeoleh n = p x q = 3 x 11 = 33 dan hasil kali untuk masing masing dikurangi 1, diperoleh z = (p – 1) x (q – 1) = 2 x 10 = 20
   - Pilih *private key* K_{rB} = d = 7 (7 ini relatif prima terhadap 20)
   - Hitung *public key* K_{pB} = e = (1 + m x z)/d = (1 + 20m)/7 = 3 (untuk m = 1).
   - Gunakan rumus $C_2 = (C_1)^e \, mod \, n$ untuk enkripsi 2 pada pengirim dan $C_1 = (C_2)^d \, mod \, n$, diperoleh :



Gambar 3. Proses enkripsi-dekripsi ciphertext ke ciphertext.

Di mana:
$C_1 = 3F20\,H$, dengan nomor urut hexadecimal (4), (16), (3), (1). Dengan *publik key* $K_{pB} = e = 3$, diperoleh $C_2 = (C_1)^e mod\, n = (4^3, 16^3, 3^3, 1^3) mod\, 33 = 31, 4, 27, 1$. Pada sisi penerima dilakukan dekripsi 1, diperoleh $C_1 = (C_2)^d \, mod \, n = (31^7, 4^7, 27^7, 1^7) mod\, 33 = 4, 16, 3, 1$ yang adalah $C_1 = 3F20\,H$ sendiri masih merupakan bentuk *ciphertext*. Sampai disini penerima B kesulitan untuk melakukan dekripsi lebih lanjut sebab ia tidak mengetahui kunci sessi yang digunakan oleh pengirim A. Ia harus menunggu dikirimkan kunci sessi itu dari A dengan proses pengiriman sebagai berikut :

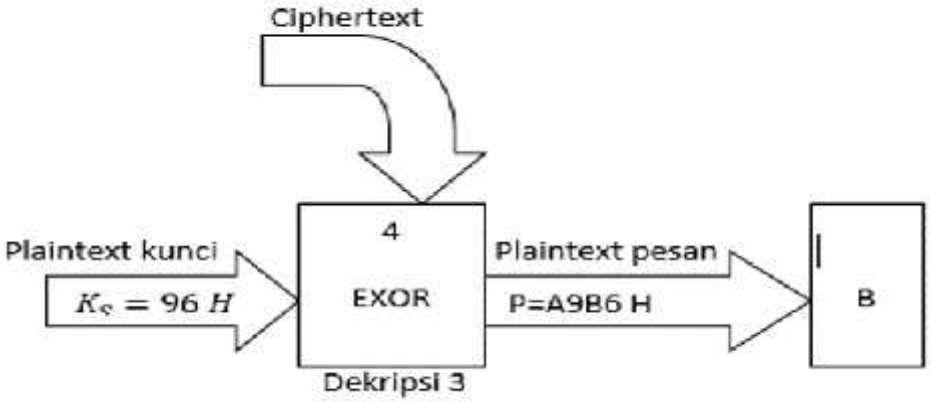


Gambar 4 Proses enkripsi-dekripsi dari kunci sessi

Di mana :

$K_S = 96\,H$ dengan nomor urut hexadecimal (10), (7). Hasil enkripsi 3 adalah $C_3 = (10^3, 7^3) mod\ 33 = 10,13$. Penerima B melakukan dekripsi 2 dengan kunci rahasia $K_{rB} = d = 7$, diperoleh $K_S = (10^7,\ 13^7) mod\ 33 = 10,7$ yang tidak lain adalah 96 H sendiri.

Proses terakhir yang dilakukan B adalah, membuka C_1 dengan menggunakan kunci sessi K_S = 96 H sebagai berikut :



Gambar 5. Proses dekripsi ciphertext dengan kunci sessi.

Dekripsi 3 ini memperlihatkan bagaimana cara membuka *ciphertext* $C_1 = 3F20\,H$ dengan menggunakan kunci sessi $K_S = 96\,H$, caranya adalah :

$C_1 = 3F20\,H \rightarrow$ 0011111100100000 B
$K_S = 9696\,H \rightarrow$ 1001011010010110 B
EXOR
1010100110110110 B
A 9 B 6 H

yakni *Plaintext* yang dikirim oleh A telah berhasil dibuka oleh penerima B, setelah melalui 3 kali proses enkripsi dan 3 kali proses dekripsi.

Tabel 1 Enkripsi dan dekripsi pesan P.

Pesan P	$C_1 = P\ XOR\ K_S$	$C_2 = C_1^e\ mod\ n$	$C_1 = C_2^d\ mod\ n$	$P = C_1\ XOR\ K_S$
A9B6 H	3F20 H	31,4,27,1	4,16,3,1 (3F20 H)	A9B6 H

Tabel 2 Enkripsi dan dekripsi kunci sessi K_S

Kunci sessi K_S	$C_3 = K_S^e \ mod \ n$	$K_S = C_3^d \ mod \ n$	$P = C_1 \ XOR \ K_S$
96 H	10, 13	10, 7 (96 H)	A9B6 H

Catatan : 1 (0H), 2 (1H), 3 (2H),…., 16 (FH), publik key e = 3, private key d = 7 dan n = 33.

Perhatikan bahwa pada kedua table diatas sangat sulit diketahui hubungan antara *plaintext* A,9,B,6 H dengan *ciphertext* 3,F,2,0 H. Kemudian 3,F,2,0 H dengan 31,4,27,1 H, serta 31,4,27,1 H dengan *ciphertext* berikutnya 4,16,3,1 H (jika masing-masing dikurangi 1, maka diperoleh 3,15,2,0 = 3F20 H). Demikin pula halnya pada table 2, kunci sessi 96 H sangat sulit diketahui hubungannya bisa menjadi 10,13 yang kemudian berubah lagi menjadi 10,7 (jika masing masing dikurangi 1 diperoleh 96 H). Kunci sessi ini akan mengembalikan *ciphertext* 3F20 H menjadi *plaintext* A9B6 H seperti yang dikirim oleh pengirim.

5. PENUTUP

Berdasarkan uraian dan penjelasan dalam illustasi kasus menggunakan sistem keamanan *PGP (Pretty Good Privacy)* dalam mengirim pesan dapat ditarik beberapa kesimpulan dan saran sebagai berikut :

5.1. Kesimpulan :

1. Enkripsi pesan plaintext dengan kunci sessi menghasilkan *ciphertext* yang sangat sulit diketahui asal muasalnya, kecuali diberitahukan kunci sessinya.
2. *Ciphertext* yang diperoleh dengan kunci sessi itu dienkripsi lanjut menggunakan publik key yang ditujukan pesan. Adapun tujuannya adalah supaya semakin sulit diketahui oleh penyusup jaringan.
3. Penerima pesan mengalami kesulitan mendekripsi pesan yang diterimanya, sebab sekalipun ia dapat membukanya dengan *private key* miliknya, tetapi ia tidak mengetahui kunci sessi untuk membukanya lebih lanjut.
4. Hanya satu cara agar penerima dapat mendekripsi pesan yang terkunci dengan kunci sessi dari pengirim yakni kunci sessi itu dikirimkan dalam keadaan terenkripsi dengan publik key penerima sehinnga penerima dapat mengdekripsinya, mengambil kunci sessi itu lalu membuka *ciphertext* yang terkunci dengan kunci sessi.
5. Kekuatan dan kehandalan PGP terletak pada kesimpulan 1,2,3,4. Pesan terenkripsi dua kali dan terdekripsi dua kali. Kunci sessi yang diperlukan untuk mendekripsi terakhir juga dienkripsi satu kali dan didekripsi satu kali baru dapat digunakan.

5.2. Saran :

PGP (Pretty Good Privacy) merupakan sistem keamanan jaringan yang sangat kuat dan handal terhadap upaya sabotase dan penyadapan dari penyusup jaringan. Oleh karena itu disarankan agar pesan berharga atau rahasia yang akan dikirim melalui jaringan seperti sertifikat, surat perjanjian, pengiriman uang antar bank dan sebagainya dapat menggunakan sistem keamanan PGP ini.

DAFTAR PUSTAKA

Tulisan ini didasarkan pada buku dan bacaan referensi sebagai berikut :

[1] ftp://ftp.ifi.uio.no/pub/pgp/(primary)
[2] ftp://ftp.ox.ac.uk/pub/crypto/pgp/
[3] ftp://ftp.dsi.unimi.it/pub/security/crypt/pgp/
[4] ftp://ftp.informatik.uni-hamburg.de/pub/virus/crypt/pgp/
[5] Hans K.M. 2002. Keamanan Komputer, Diktat kuliah pada STMIK Dipanegara Makassar
[6] John D. Howard. 1995. Analysis Of Security Incidents On The Internet.
[7] Rachman I. 2010. PGP, Seminar Keamanan jaringan STMIK Dipanegara Makassar.
[8] Singh A, Ttiebel A. W, 2005. Network Security. John Wiley & Son
[9] Tanenbaum S.A. 2001. Komputer Network. Prentice-Hall, Inc.