Pengamanan Surat Elektronik dengan *PGP/OpenPGP*

Chan Lung dan Arisat Fajar H.P.

Departemen Teknik Informatika
Institut Teknologi Bandung
Jalan Ganesha 10 Bandung 40132

E-mail : if11039@students.if.itb.ac.id, if11073@students.if.itb.ac.id

Abstrak

Surat elektronik atau yang lebih dikenal dengan *EMail*, telah menjadi sebuah media komunikasi yang sangat umum digunakan sekarang ini. Sebagian besar orang tentu sudah mengerti dan menggunakan fasilitas ini, namun tidak semua orang sadar dengan bahaya yang ada. Sebuah *EMail* sangat rentan terhadap kejahatan dunia cyber (*cyber crime)* seperti contohnya penyadapan, pengubahan isi *EMail* sampai dengan pemalsuan *EMail*. Seorang ahli kriptografi bernama Phil R Zimmermann berupaya untuk memecahkan masalah ini dengan mengembangkan sebuah teknologi yang disebutnya PGP *(Pretty Good Privasi)*. Sedangkan *OpenPGP* adalah teknologi yang sama yang dikembangkan atas dasar *Public GNU*. *OpenPGP* dengan kemampuannya mengenkripsi pesan mampu menjamin bahwa sebuah pesan baru dapat dipecahkan isinya dengan metode *brute force* setelah percobaan selama jutaan tahun. Selain itu adanya tanda tangan digital *(digital signature)*, fasilitas lain di dalam *OpenPGP*, para pengguna dapat memastikan keaslian sebuah dokumen dan kebenaran nama pengirimnya. Bagian terbaik adalah teknologi tersedia secara gratis untuk masyarakat non komersial. *OpenPGP* masih dikembangkan oleh para ahli hingga saat ini, mereka berharap dapat membuat sebuah teknologi kriptografi yang sempurna dan tidak terpecahkan dengan tingkat efektifitas yang tinggi.

***Kata kunci**: EMail, PGP, OpenPGP*

1. Pendahuluan

Internet saat ini menjadi sebuah teknologi yang membawa perubahan besar di dunia saat ini. Sebuah informasi dapat disebar dan dibaca oleh seluruh orang di bumi ini hanya dengan hitungan detik. Kemudahan ini membuat banyak orang kini menggunakan teknologi internet dalam kehidupan sehari.

Banyak sekali hal yang ditawarkan oleh Internet, mulai dari penyediaan informasi, perdagangan sampai dengan hiburan. Teknologi-teknologi pendukung nya pun kini banyak bermunculan, salah satunya yang sangat populer adalah surat elektronik atau yang biasa dikenal dengan sebutan *EMail*.

EMail seperti namanya mirip dengan fungsi surat pada umumnya, yaitu untuk mengirimkan berita, kabar atau informasi dari pengirim kepada satu orang atau lebih penerima. Perbedaannya bila surat biasa tertulis secara fisik, maka *EMail* tertulis secara digital dan didistribusikan melalui media Internet dan bukan diantar oleh seorang kurir surat.

Beberapa keunggulan dari *EMail* dibandingkan surat tertulis fisik adalah:

1. Penyebarannya yang cepat, bahkan orang yang berbeda negara dapat menyampaikan pesannya dalam hitungan detik.
2. *EMail* dapat menyertakan data berupa gambar, suara, video atau file lain di dalam isinya.
3. Biayanya murah dan sangat efisien dibandingkan surat fisik.

Namun *EMail* tidak hanya menawarkan kebaikan dan sisi positif saja. Banyak ancaman yang bisa saja muncul sehubungan dengan penggunaan *EMail* ini, antara lain yang berhubungan dengan masalah keamanannya adalah:

1. Kerahasiaan isi pesan atau *EMail* apabila pesan tersebut berhasil disadap atau diketahui oleh orang lain.

2. Keaslian isi pesannya. Apakah *EMail* tersebut masih asli dan tidak mengalami perubahan oleh pihak ke tiga.
3. Keabsahan pengirim, yaitu memang orang yang tertanda di *EMail* tersebut yang mengirimnya dan bukan orang lain.
4. Anti penyangkalan, artinya orang yang mengirim tidak bisa menyangkal bahwa dia yang mengirimkan.

Masalah-masalah keamanan di atas dapat dipecahkan dengan metode kriptografi yang tepat. Para ahli kriptografi pun mengembangkan sebuah teknologi yang mereka sebut dengan *OpenPGP* yang bisa digunakan untuk menjamin keamanan sebuah *EMali*, sehingga masalah-masalah di atas dapat dipecahkan dengan baik dan dengan biaya yang serendah mungkin.

2. Sejarah Kriptografi dan PGP

2.1 Sejarah Kriptografi

Kriptografi adalah seni dan ilmu untuk menjaga keamanan pesan. Teknik kriptografi ternyata telah dipakai dari sejak dulu. Pada zaman Julius Caesar misalnya, untuk menjaga kerahasiaan pesan yang akan dikirimkannya dia mengubah isi pesan untuk membingungkan musuh yang membacanya. Perhatikan contoh *Caesar Chipper* ini:

kata asli: AKU
disandikan: DNX

Dengan menukarkan kata asli, yang selanjutnya disebut plainteks, dengan 3 karakter di depannya maka akan didapatkan kata sandi, yang selanjutnya disebut sebagai chipperteks, seperti pada contoh.

Pada Perang Dunia II, pihak Jerman mengklaim telah membuat sebuah mesin penyandian yang disebut Enigma yang tidak mungkin dipecahkan tanpa kunci yang tepat. Mesin ini menggabungkan beberapa fungsi manipulasi karakter seperti substitusi dan transposisi.

Metode metode yang bisa digunakan untuk proses penyandian ini adalah:

1. Subtitusi, yaitu dengan menukarkan satu karakter dengan satu atau lebih interpretasi atau karakter yang lain.
2. Transposisi, yaitu dengan menukarkan keterurutan dari posisi karakter-karakter yang ada.
3. Perhitungan matematika, yaitu penjumlahan, perkalian, pengurangan, pembagian, modulus, perpangkatan dan mungkin penggabungan dari beberapa operasi.
4. Operasi boolean yang umumnya digunakan pada algoritma kriptografi modern, seperti operasi AND, OR, XOR, NOT.

Seiring dengan kemajuan zaman dan kemunculan komputer, teknik kriptografi pun berubah. Penyandian pesan tidak lagi dilakukan karakter per karakter tapi bit per bit (atau byte per byte) dengan metode ini tentu saja akan didapatkan hasil yang lebih baik tapi prosesnya pun menjadi lebih rumit. Para perancang kemudian membagi kriptografi ini menjadi 2 jenis besar yaitu kriptografi simetri dan kriptografi asimetri.

Dikatakan simetri karena untuk mengembalikan chipperteks menjadi plainteks, disebut proses dekripsi, cukup dibutuhkan kunci yang sama pada saat melakukan proses penyandian dari plainteks menjadi chipperteks, disebut proses enkripsi. Algoritma kriptografi simetri dapat dibagi menjadi 2 lagi berdasarkan proses yang dilakukan terhadap bit-bitnya, yaitu *stream chipper* (aliran) yaitu proses dilakukan terhadap satu-satuan bit, dan *block chipper* (blok) dimana operasi dilakukan per blok bit atau byte. Contoh dari algoritma simetri ini adalah *IDEA, DES, AES, tripleDES*, dll..

Sedangkan pada kriptografi asimetri, untuk melakukan dekripsi dibutuhkan kunci yang berbeda dengan saat melakukan enkripsi, maka perlu dibangkitkan lebih dari satu kunci yang saling berkoresponden untuk proses dekripsi dan enkripsi. Kompleksitas algoritma ini sangat tinggi karena biasanya melibatkan perhitungan matematika yang kompleks. Contoh kriptografi jenis ini adalah *Diffie-Helman* (*DH*), *RSA*, dll.. Umumnya algoritma jenis ini tidak digunakan untuk mengenkripsi pesan secara langsung tapi digunakan dalam prinsip *public key algoritma* dimana dua buah kunci dibangkitkan yang satu diumumkan secara meluaskan pada

khalayak ramai dan satu lagi disimpan sebagai kunci rahasia.

Tidak berhenti pada algoritma kriptografinya saja, para peneliti juga mengembangkan algoritma – algoritma lain yang digunakan untuk mendukung proses kriptografi ini, seperti misalnya algoritma perhitungan bilangan yang besar (mencapai > 50 digit angka), pembangkitan bilangan acak, pembangkitan kunci secara acak atau menggunakan *keystroke* atau menggunakan *mouse gesture*, dan salah satu algoritma pendukung yang sangat penting adalah pembangkitan *message diggest*.

Message diggest, biasa disingkat *MD*, adalah sebuah string yang panjangnya tetap yang dihasilkan dari masukkan string yang panjangnya sembarang, dalam hal ini masukkan yang dimaksud bisa saja merupakan sebuah pesan. *MD* dibangkitkan berdasarkan apa yang disebut sebagai fungsi hash satu arah, artinya nilai *MD* yang dibangkitkan tidak mungkin dikembalikan lagi menjadi pesan semula. *MD* akan menjadi bagian penting dari algoritma kriptografi terutama pada proses pemberian tanda tangan digital. Algoritma yang umum digunakan untuk membangkitkan *MD* adalah *MD5* dan *SHA*.

2.2 Sejarah PGP

Pada tahun 1991 senat Amerika mengeluarkan aturan bahwa setiap perangkat lunak yang digunakan untuk proses enkripsi pesan harus memiliki *back door*, tidak pernah dijelaskan secara terperinci mengapa tapi alasan paling kuat saat itu adalah pemerintah ingin mengontrol semua aktivitas yang dilakukan organisasi-organisasi sehingga tidak bertentangan dengan kemauan senat. Akan tetapi seorang ilmuwan yang bernama Phil R Zimmermann berani untuk tidak memperdulikan peringatan itu.

Phil R Zimmermann kemudian menulis sebuah algoritma yang mengimplementasikan *RSA*, yang saat itu patennya masih dipegang oleh MIT, dikombinasikan dengan algoritma kriptografi simetri ciptaannya sendiri yang disebut *Bass-O-Matrix*. Phil juga menggunakan *MD4* untuk membangkitan *MD* dari pesan yang dienkripsi. Dia membagikan hasil kerjanya tersebut kepada teman-temannya untuk mendapatkan analisis yang akurat mengenai kehandalan perangkat lunak yang dibuatnya.

Salah seorang teman Phil, Kelly Goen, mendistribusikan penemuan Phil ini kepada khalayak umum hingga pada akhirnya Phil mendapatkan bahwa hasil kerjanya sudah terpasang di Internet dan mulai diteliti dan dipakai banyak orang. Sementara usaha Phil tidak berhenti di situ saja, selama PGP versi 1.0 nya mulai beredar, dia mulai menulis versi kedua nya.

Phil melakukan perbaikan-perbaikan seperti misalnya penggunaan algoritma simetri *IDEA* yang dinilai lebih baik daripada *Bass-O-Matrix*, penggantian *MD4* menjadi *MD5*. Phil juga menambahkan proses kompresi data dengan menggunakan ZIP sebelum pesan dikompresi. Bahkan pada versi-versi selanjutnya digunakan *radix-64* ASCII Armor untuk menangani pesan yang tidak hanya mengandung kata-kata teks saja.

Namun permasalahan justru datang karena proses penyebaran *PGP* miliknya. MIT yang memegang paten dari *RSA* merasa bahwa Phil menyalahi aturan penggunaan hak paten, tapi Phil bisa membantah hal tersebut dengan mengatakan bahwa masalah izin *RSA* seharusnya ditanggung sendiri oleh pihak pengguna seperti tertulis pada *User Agrement*. Selang beberapa tahun masalah ini dapat terselesaikan dan MIT akhirnya menjadi partner bagi Phil untuk mendistribusikan PGP.

Di lain pihak, senat Amerika mulai tidak menyukai proses penyebaran dan karena dianggap menyalahi aturan akhirnya Phil dijatuhi dakwaan. Pemerintah Amerika melakukan penyelidikan terhadap nya selama 3 tahun sampai pada akhirnya dia diputuskan tidak bersalah.

Pemutusan tidak bersalah ini dinilai beberapa pihak sangat ganjil sehingga mereka mengambil keputusan bahwa Phil telah bekerja sama dengan pihak pemerintah Amerika untuk membuat *back door*. Tentu saja ini dibantah Phil dan dia mengatakan bahwa itu sama sekali tidak benar.

Saat ini telah bediri organisasi *OpenPGP* yang mengerjakan pengembangan *PGP* melalui lisensi *Public GNU*. Mereka memastikan bahwa keraguan dan tudingan dari orang-orang itu tidak benar. Bersama-sama dengan Phil sebagai guru besarnya, mereka membangun dan menyebarkan

PGP secara gratis. Bahkan mereka menyediakan *source code* yang bisa di download di www.openpgp.org sehingga orang bisa melakukan analisis dan meneliti apakah memang terdapat *back door.*

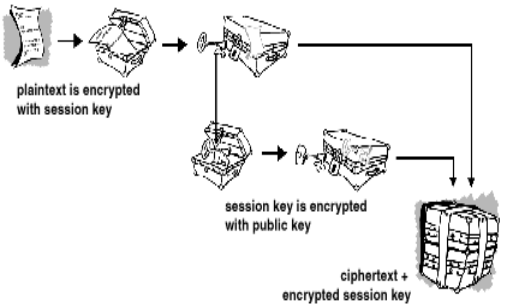
Saat ini *OpenPGP* telah berkembang pesat dan mampu terintegrasi dengan perangkat lunak yang pemroses *EMail* seperti Microsoft Outlook serta browser-browser seperti I.E,. Netscape, dll.. Pengenkripsiannya pun tidak terbatas hanya *EMail* teks saja, *PGP* yang terbaru bahkan mampu mengenkripsi *EMail* dengan *attachment* sesuai dengan standar *OpenPGP-MIME* seperti yang dideskripsikan di dalam *RFC2015*.

4. Prinsip Kerja *PGP*

PGP pada dasarnya merupakan sebuah *hybrid cryptosystem*, artinya *PGP* mendasarkan metode kriptografinya dengan menggunakan beberapa metode kriptografi yang ada, dalam hal ini *PGP* menggunakan metode kriptografi simetri dan asimetri. Algoritma yang digunakan bisa bermacam-macam, tergantung dari versi PGP yang bersangkutan. *OpenPGP* sendiri memungkinkan pengguna untuk memilih salah satu dari algoritma simetri berikut ini *IDEA, TripleDES (DES-EDE), AES, Reverse DES, CAST5, Blowfish* dan *SAFER-SK128*. Sedangkan untuk algoritma asimetrinya, pengguna bisa memilih untuk menggunakan algoritma *RSA, DSA* atau *Elgamal.*

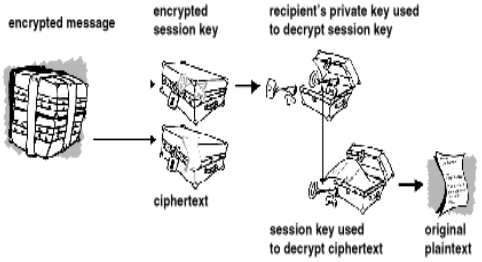
Selain daripada itu, untuk memperkecil dan mempercepat proses transfer pada saat melakukan pengiriman melalui internet maka data yang akan dienkrip dan dikirim terlebih dahulu di kompres dengan menggunakan teknik pengompresan, seperti contohnya ZIP, RAR, ACE atau lainnya. Pengompresan ini selain memperkecil ukuran file juga memiliki keunggulan lain seperti mengurangi adanya pola dalam plaintext sehingga akan lebih sulit untuk dipecahkan. Apabila setelah dilakukan kompresi ternyata file hasilnya berukuran lebih besar maka file tersebut akan dikembalikan seperti semula dan tidak akan mengalami proses kompresi saat di enkripsi. *OpenPGP* menyediakan 4 opsi untuk proses ini yaitu plainteks tanpa kompresi, kompresi dengan ZIP, kompresi dengan ZLIB atau kompresi lain yang sedang dikembangkan oleh organisasi *OpenPGP*.

PGP kemudian akan membangkitkan sebuah *session key*, sejumlah kunci yang dibangkitkan untuk sekali penggunaan saja. Kunci ini dibangkitkan secara acak begantung dari gerakan mouse atau penekanan tombol pada keyboard. Plainteks kemudian dienkripsi dengan menggunakan kunci ini dengan menggunakan metode kriptografi simetri. Setelah pesan berhasil dienkripsi, maka selanjutnya *session key* akan dienkripsi dengan menggunakan kunci publik penerima. Pesan dan kunci ini lalu dijadikan satu dan dikirimkan melalui internet.



Gambar 1 Proses dekripsi pada PGP

Proses dekripsi merupakan kebalikan dari proses enkripsi. Pesan diesktrasi dan dengan menggunakan kunci privatnya, penerima berusaha mendekripsikan *session key* dan kemudian menggunakannya untuk mendekripsi pesan.



Gambar 2. Proses dekripsi pada PGP

Alasan mengapa digunakan algoritma simetri untuk mendekripsi pesan ketimbang asimetri adalah karena proses enkripsi dan dekripsi dengan algoritma enkripsi berjalan lebih cepat yaitu sekitar 1000 kali. Sedangkan algoritma asimetri digunakan untuk mengenkripsi kunci karena kunci sendiri tidak mungkin sangat panjang, biasanya terbatas,

lagipula dengan algoritma ini distribusi kunci bisa secara aman dilakukan meskipun dilakukan pada jaringan yang tidak aman.

4.1 Kunci

Kunci memegang peranan utama di dalam proses enkripsi dan dekripsi. Sebuah kunci untuk algoritma simetri dapat dibangkitkan secara acak dan bisa saja tidak berarti apa-apa. Hubungan antara kunci yang satu dan kunci yang lain pun sama sekali tidak ada. Hal ini jelas berbeda dengan prinsip kunci publik-privat.

Kunci publik dan privat tidak bisa dibangkitkan secara random begitu saja. Proses pembangkitan kunci ini membutuhkan sebuah operasi matematika yang rumit. Umumnya untuk menjaga agar sebuah proses enkripsi benar-benar aman untuk jangka waktu yang lama maka pasangan kunci yang dibangkitkan harus panjang. Perlu diingat bahwa pembangkitan kunci harus memperhatikan aspek berikut, kunci yang dibangkitkan harus cukup panjang untuk menjaga keamanan pesan tapi juga harus cukup pendek sehingga dapat diproses secara cepat. Kunci dengan panjang mencapai >10000 bit akan memberikan keamanan yang sangat tinggi, tapi tidak masuk akan karena komputasinya akan memakan waktu yang tidak sedikit, tentu saja ini dinilai tidak efektif.

Sehubungan dengan masalah manajemen kunci ini PGP juga memperkenalkan apa yang disebut *keyring. Keyring* adalah sebuah file yang berisi kumpulan kunci-kunci publik dan kunci privat. Kumpulan kunci-kunci publik berarti seorang pengguna dapat menambahkan dan menyimpan kunci publik dari pengguna lain yang akan dikirimi pesan.

Perlu untuk diketahui, bahwa kunci privat tidak hanya digunakan untuk mendekripsi *session key* tapi juga dapat digunakan untuk tanda tangan. Terkadang dalam sebuah perusahaan pengguna diharuskan untuk menggunakan kunci private yang sama untuk menandatangani keabsahan suatu pesan.

Masalah bisa timbul apabila kunci tersebut dibagikan begitu saja satu persatu, karena seseorang bisa mengirim tanpa sepengetahuan yang lain dan lalu menyangkalnya.

PGP menyediakan kemampuan untuk memecah kunci menjadi bagian-bagian yang kemudian dibagikan kepada personil yang dipercaya. Tanpa dilakukan penggabungan terhadap kunci-kunci tersebut maka kunci privat tidak bisa digunakan. Dengan cara ini, maka dapat dilakukan kontrol terhadap keabsahan pengiriman, karena untuk menandatanganinya diperlukan izin dari semua pihak yang memegang kunci tersebut.

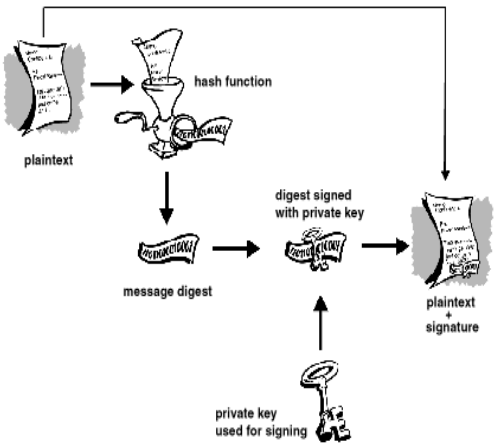
4.2 Tanda tangan digital

Seperti yang telah disebutkan di atas bahwa algoritma kunci publik memungkinkan adanya penandatangan dokumen. Sebuah tanda tangan digital yang diberikan pada pesan mempunyai 3 fungsi utama:

1. Keabsahan pengirim (*user authentication*). Hal ini berkaitan dengan masalah kebenaran identitas pengirim dan menjawab pertanyaan "apakah pesan yang diterima benar-benar berasal dari pengirim yang sesungguhnya?".
2. Keaslian pesan (*message integrity*). Berkaitan dengan keutuhan pesan. Menjawab pertanyaan "Apakah pesan yang diterima tidak mengalami perubahan (modifikasi)?"
3. Anti Penyangkalan (*nonrepudiation*). Pengirim tidak dapat menyangkal tentang isi pesan yang dia kirim.

Pada *PGP*, tanda tangan dilakukan dengan menambahkan *MD* yang dibangkitkan dari pesan yang bersangkutan dan telah dienkripsi dengan menggunakan kunci privat pemiliknya. Alasan digunakannya *MD* adalah karena *MD* dapat dibangkitkan dari fungsi hash satu arah dan menghasilkan keluaran dengan panjang tetap yang unik dan tidak mungkin sama. Kunci privat digunakan untuk membuktikkan bahwa memang benar dokumen itu dikirim oleh pemilik kunci privat tersebut dan bukan oleh orang lain, hal ini dapat dilakukan karena kunci privat dan kunci publik saling berkorespondenan.

Gambar berikut menggambarkan bagaimana proses penambahan tanda tangan pada *PGP*.



Gambar 3. Proses tanda tangan digital pada PGP

Proses otentikasi tanda tangan dilakukan dengan cara yang sama dengan saat penandatanganan. Terlebih dahulu tanda tangan pesan diekstrasi. Tanda tangan tersebut masih dalam bentuk tanda tangan yang terenkripsi, untuk mendekripsinya maka digunakan kunci publik dari pengirimnya, akan didapatkan *MD* dari pesan tersebut. Selanjutnya bangkitkan lagi *MD* dari pesan tersebut dan bandingkan dengan *MD* yang didapat dari proses dekripsi tadi. Apabila sama maka pesan tersebut masih asli dan tidak mengalami perubahan.

4.3 Sertifikat Digital

Menerima dan menggunakan kunci publik seseorang di dalam dunia Internet bukanlah sesuatu yang bisa begitu saja dilakukan. Diperlukan sebuah mekanisme yang baik sehingga pesan tersebut tersampaikan kepada orang yang benar. Contohnya adalah sebagai berikut Alice hendak berhubungan dengan Bob melalui Internet, untuk itu Alice membutuhkan publik key Bob. Seseorang entah itu siapa memberikan publik keynya dan menyatakan bahwa dirinya adalah Bob. Dari mana Alice tahu kalau itu memang benar adalah Bob?

Permasalahan ini bisa diatasi dengan menggunakan dua cara

1. Menyediakan server yang menyimpan basis data kunci publik dari penggunanya. Metode ini menimbulkan masalah baru yaitu bagaimana memastikan bahwa Alice benar-benar sudah terhubung dengan server bukan dengan orang lain. Beberapa protokol kriptografi mampu memecahkan masalah ini, tapi pada makalah kali ini pembahasan lebih lanjut mengenai protokol ini tidak dilakukan.
2. Menggunakan *Publik Key Infrastructure* (*PKI*). Sebuah PKI menyediakan kemampuan untuk mendistribusikan kunci publik ke setiap individu-individu yang memungkinkan tapi juga mampu menyediakan manajemen terhadap kunci-kunci tersebut. Dengan cara ini 2 orang dapat berhubungan secara langsung tanpa melalui perantara server. Pertanyaan awal yang timbul, bagaimana meyakini bahwa itu kunci publik yang benar dapat dijawab dengan melakukan apa yang disebut dengan *chain of trust*. *Chain of trust* merupakan suatu metode di mana seseorang yang ingin mengetahui keaslian kunci publik seseorang maka dia dapat melakukan pengecekan ke atas terhadap penyedia atau pemberi kunci publik tersebut. Kunci publik dinyatakan benar apabila tanda tangan yang disertakan pada kunci publik setelah di dekripsi dengan kunci publik penyedia ternyata sama. Proses ini dapat dilakukan sampai pada tingkat tertinggi yaitu di *root*, apabila kita mendapatkan bahwa *root* menyetujui kunci publik di bawahnya dan pengguna berpegang pada *root* yang sama maka kita dapat mempercayai bahwa semua tanda tangan di bawah *root* itu adalah benar.

Pada kehidupan nyata, ada lembaga-lembaga yang menyediakan servis pengadaan tanda tangan ini. Lembaga-lembaga ini disebut sebagai *Certification Authority*. Tentu saja untuk mendapatkan pelayanan seperti ini peminta diwajibkan membayar sejumlah uang. Umumnya perusahaan-perusahaan lah yang menggunakan jasa ini. Sertifikat yang diberikan oleh CA disebut sebagai X.509.

PGP menggunakan metode yang kedua untuk menyelesaikan permasalahan yang disampaikan pada awal-awal sub bab ini. Perbedaannya rantai kepercayaan ini tidak dibangun dengan menggunakan lembaga-lembaga penyedia jasa pemberi kunci publik. *PGP* membangun sendiri fasilitas ini, sehingga user bisa mendapatkan kunci publiknya secara gratis dan menyebarkannya dengan cepat.

Prinsip yang digunakan pun tidak jauh berbeda, user cukup melakukan pengecekan ke atas terhadap user lainnya. Sebagai contoh Alice dapat melakukan pengecekan terhadap orang lain berdasarkan key ring tersimpan di Bob. Dari situ setidaknya Alice akan mendapatkan orang lain, misalkan Trent, yang juga dipercaya oleh Alice. Karena Trent mempercayai kunci publik Bob maka Alice yakin kalau itu memang adalah Bob dan bukan orang lain.

Kalau Alice adalah orang baru yang belum memiliki jaringan sama sekali, maka dia dapat mereferensikan kunci publiknya pertama kali kepada pihak yang terkenal secara international. Seperti contoh Phil sendiri memberitahukan kunci publiknya melalui web sitenya. Organisasi *OpenPGP* pun memiliki kunci publiknya sendiri yang bisa dijadikan referensi pertama.

Kunci publik yang telah diterima dan digunakan dapat disimpan di dalam *key ring*. Proses penyimpanan ini akan memudahkan pengguna apabila ingin berkomunikasi lagi dengan pengguna yang lain.

Perlu diperhatikan bahwa kunci publik ini perlu diperbaiki atau diganti setelah kurun waktu tertentu. *PGP* juga menyediakan fasilitas untuk mengecek apakah kunci publik tersebut masih valid penggunaannya.

Proses penjagaan kerahasiaan, integritas, dan anti penyangkalan dapat dijelaskan sebagai berikut:

1. Pesan yang akan dikirim terlebih dahulu di bangkitkan *MD* nya.
2. *MD* tersebut dienkripsi dengan menggunakan kunci privat pengirim. Hasilnya ditambahkan pada pesan.
3. Bangkitkan kunci sesi dengan cara seperti dijelaskan di atas.
4. Enkripsi pesan yang telah dibubuhi tanda tangan dengan menggunakan algoritma simetri dan kunci sesi yang telah dibangkitkan.
5. Ambil kunci publik penerima yang terdapat pada *ring key* dan gunakan untuk mengenkripsi kunci sesi. Tambahkan hasil enkripsi ini di pesan yang telah terenkripsi tadi.
6. Pesan dan *session key* yang telah terenkripsi tersebut dikirim melalui saluran transmisi.

Sedangkan fungsi deskripsi dan otentikasi dokumen dilakukan dengan langkah kebalikan dari proses di atas.

5. Kelebihan dan Kekurangan

5.1 Kelebihan

PGP yang menggunakan *hybrid cryptosystem* memiliki kekuatan yang sangat tinggi di dalam proses enkripsinya. Hal ini disebabkan karena algoritma yang digunakan *PGP* dianalisis oleh pakar-pakar dari seluruh dunia dan dapat dibuktikkan kehandalannya. *tripleDES*, *AES*, *RSA* adalah contoh-contoh algoritma yang sangat tinggi kompleksitasnya dan hingga saat ini belum ditemukan cara atau algoritma yang mangkus untuk memecahkan algoritma enkripsi di atas dengan cepat.

Kelebihannya yang kedua adalah, untuk perangkat lunak yang bermutu ini masyarakat umum dapat menikmatinya dengan gratis bahkan memperoleh kesempatan untuk berperan serta mengembangkan atau memberikan kritikan dan masukan yang membangun. *OpenPGP* yang dikembangkan di bawah lisensi *Public GNU* menjawab semua masalah harga dan hak cipta ini.

5.1 Kekurangan

Meskipun secara kriptografi proses enkripsi tidak terdapat kelemahan, setidaknya begitulah kesimpulan para ahli sampai saat ini, tapi pada saat pembangkitan kunci masih terdapat kelemahan mendasar. Pembangkitan kunci dengan menggunakan *keystroke* atau gerakan mouse masih bisa dideteksi dengan menggunakan perangkat lunak yang berjalan di *back ground* dan mencatat semua *keystroke* atau gerakan mouse yang terjadi. Bahkan ada beberapa yang mampu melakukan record dan perulangan gerakan-gerakan tersebut.

Kekurangan kedua adalah penggunaan teknik kompresi yang umum tidak akan banyak membantu proses enkripsi hanya akan menambah waktu proses saja, meski dalam hal ini mungkin hanya berselang beberapa menit. Akan lebih baik apabila dikembangkan algoritma lain yang sifatnya rahasia.

6. Kesimpulan

PGP merupakan sebuah teknologi pengenkripsian pesan atau *EMail* yang baik dan kuat. Sangat sukar untuk dapat memecahkan pesan ini hanya dengan mengandalkan metode *brute force*. *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, *PGP* 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.

[1] Callas, *"OpenPGP Message Format"*, RFC 2440, November 1998.

[2] *D*. Atkins, Stallings W. and P. Zimmermann, *"PGP Message Exchange Formats"*, RFC 1991, Agustus 1996.

[3] M. Elkins, *"MIME Security with Pretty Good Privacy (PGP)"*, RFC 2015, Oktober 1996.

[4] *OpenPGP Organization*, *www.openpgp.org*, diakses tanggal 8 Januari 2005 pukul 13:35.