Studi Analisis Algoritma CAST dan Implementasinya dalam PGP

Nabila As'ad

Program Studi Teknik Informatika
Alamat Jl. Ganeca 10 Bandung
e-mail: if17006@students.if.itb.ac.id

ABSTRAK

Kriptografi memiliki keterkaitan yang penting dalam dunia penyebaran informasi pada saat ini. Banyak terdapat para *intruder* atau *hacker* yang menyadap komunikasi antara seseorang dengan seseorang lainnya. Hal tersebut membuat informasi menjadi suatu hal krusial yang harus dijaga kerahasiaannya. Salah satu cara dalam bertukar informasi adalah dengan menggunakan pesan elektronik. Terdapat berbagai macam algoritma yang digunakan dalam menjaga keamanan informasi. Salah satu diantaranya adalah Algoritma CAST. Algoritma CAST merupakan algoritma *block cipher* yang telah digunakan dalam berbagai produk, seperti contohnya pada PGP (*Pretty Good Privacy*). PGP digunakan antara lain untuk menjaga kerahasiaan pesan elektronik (*electronic mail*) dan otentikasi *digital signature*. Versi-versi terbaru dari PGP menggunakan algoritma CAST sebagai algoritma kunci simetri dan algoritma Diffie-Hellman sebagai algoritma kunci publik.

Kata kunci: *Pretty Good Privacy*, Kriptografi Kunci Simetri, *Block Cipher*, Jaringan Feistel, CAST.

1. PENDAHULUAN

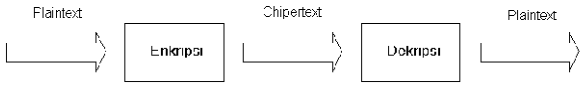
1.1 Kriptografi

Kriptografi berasal dari bahasa Yunani κρυπτός, *kryptos* yang berarti rahasia tersembunyi. Kriptografi sendiri merupakan sebuah studi tentang menyembunyikan informasi. Ilmu ini mempelajari mengenai penyembunyian huruf atau tulisan sehingga mebuat tulisan tersebut tidak dapat dibaca oleh orang yang tidak berkepentingan. Selain pengertian tersebut, juga terdapat pengertian bahwa kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data[9].

Kriptografi modern merupakan sebuah disiplin ilmu yang memadukan antara matematika, ilmu komputer, dan teknik. Contoh beberapa aplikasi yang menggunakan kriptografi adalah kartu ATM, *password* untuk komputer, dan *electronic commerce*. Pada jaman dahulu, kriptografi telah digunakan pertama kali pada jaman Julius Caesar dengan metode Caesar Cipher. Pada jaman sebelum era modern, kriptografi lebih *concern* kepada sebuah pesan saja.

Algoritma kriptografi dapat dibedakan menjadi kriptografi klasik dan kriptografi modern. Kriptografi klasik adalah kriptografi yang menggunakan algoritma berbasis karakter. Teknik ini sudah digunakan sejak jaman dahulu kala, sehingga algoritma ini sebenarnya tidak membutuhkan penggunaan komputer dalam implementasinya, cukup menggunakan pena dan kertas saja. Algoritma kriptografi klasik termasuk dalam kriptografi kunci-simetri. Terdapat dua jenis algoritma kriptografi klasik, yaitu *cipher* substitusi dan *cipher* transposisi. Inti dari *cipher* substitusi adalah mengganti suatu huruf dengan huruf lainnya dengan cara menggeser tiap huruf alfabet. Contoh *cipher* substitusi adalah Caesar Cipher. Di sisi lain, *cipher* transposisi bekerja dengan prinsip menukar posisi huruf-huruf yang ada dalam plainteks (seperti permainan anagram). Kedua algoritma ini dapat digabungkan menjadi Super-Enkripsi.

Pada kriptografi terdapat dua proses utama yang terjadi yaitu enkripsi dan dekripsi. Enkripsi merupakan proses merubah pesan asli (dinamakan plainteks) menjadi sebuah pesan terenkripsi yang tidak dapat diartikan seperti aslinya. Untuk dapat membaca pesan tersebut, pesan hasil enkripsi (yang dinamakan cipherteks) harus didekripsi dengan kunci yang sama. Berikut merupakan ilustrasi proses enkripsi dan dekripsi.



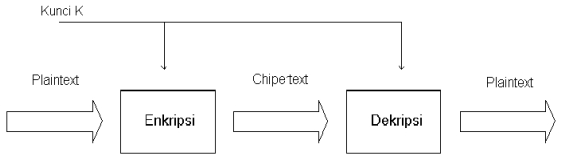
Gambar 1. Proses Enkripsi dan Dekripsi Secara Umum

Pada gambar tersebut pesan plainteks akan dienkripsi (masuk ke blok enkripsi) menjadi sebuah cipherteks yang kemudian didekripsi (masuk ke blok dekripsi) untuk menjadi plainteks kembali.

1.2 Kriptografi Kunci Simetri

Terdapat dua model algoritma enkripsi pada kriptografi yang menggunakan kunci, yaitu kriptografi kunci simetrik dan kunci asimetrik. Algoritma CAST yang akan dibahas lebih lanjut pada makalah ini merupakan salah satu kriptografi kunci simetris.

Enripsi kunci simetrik biasanya disebut juga sebagai enkripsi kunci konvensional. Enkripsi kunci simetrik merupakan proses enkripsi yang menggunakan kunci yang sama pada saat melakukan enkripsi maupun pada saat melakukan dekripsi. Gambar berikut merupakan ilustrasi untuk enkripsi kunci simetrik.



Gambar 2. Proses Enkripsi Kunci Simetri

Pada gambar tersebut dapat terlihat bahwa untuk melakukan enkripsi dan dekripsi hanya menggunakan satu kunci yang sama. Dalam menggunakan metode ini dibutuhkan adanya komunikasi antara si pengirim dan si penerima. Komunikasi dimaksudkan untuk memberitahukan kunci kepada si penerima agar penerima dapat mendekripsi pesan.

Keamanan kerahasiaan suatu pesan tergantung pada keamanan kerahasiaan dari kunci itu sendiri. Oleh karena itu, apabila terdapat *intruder* (penyusup) yang mengetahui kunci tersebut, kerahasiaan pesan akan terbongkar dan pesan dapat dibaca oleh pihak yang tidak berkepentingan.

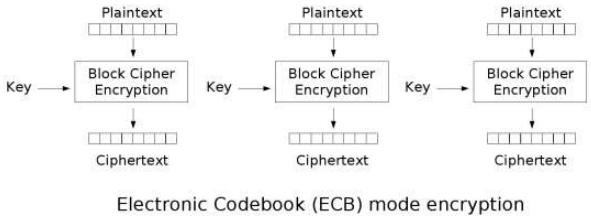
Terdapat dua metode yang menggunakan enkripsi kunsi simetrik, yaitu metode *stream cipher* dan metode *block cipher*. CAST merupakan algoritma yang menggunakan metode *block cipher*.

1.3 *Block Cipher*

Metode *block cipher* merupakan algoritma kunci yang akan membagi-bagi plainteks yang akan dikirimkan ke penerima menjadi ukuran-ukuran tertentu (yang disebut blok) dengan panjang t, dan setiap blok dienkripsi dengan menggunakan kunci yang sama. Pada umumnya, *block cipher* memproses plainteks dengan panjang blok yang relatif lebih panjang dari 64 bit, hal tersebut untuk mempersulit penggunaan pola-pola serangan yang ada untuk membongkar kunci.

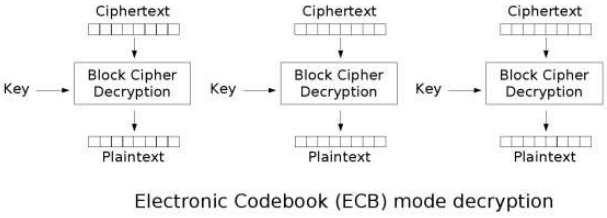
Metode *block cipher* ini memiliki empat tipe enkripsi dan dekripsi, yaitu:

1. ECB (*Electronic Code Book*)
   Skema dari proses **ENKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut.



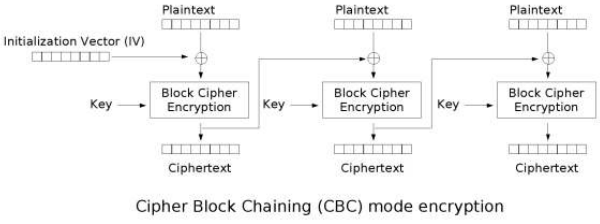
Gambar 3. Mode Enkripsi ECB

   Skema dari proses **DEKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



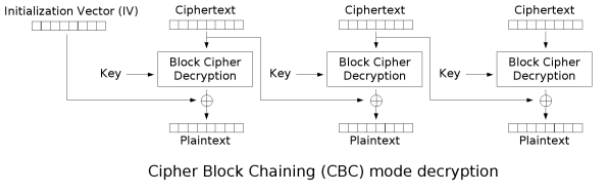
Gambar 4. Mode Dekripsi ECB

2. CBC (*Cipher Block Chaining*)
   Skema dari proses **ENKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



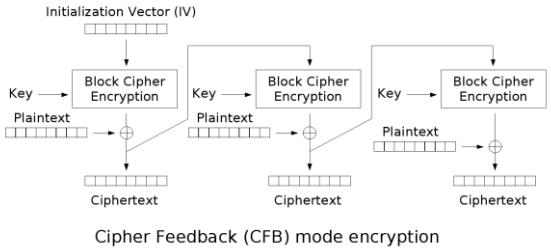
Gambar 5. Mode Enkripsi CBC

Skema dari proses **DEKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



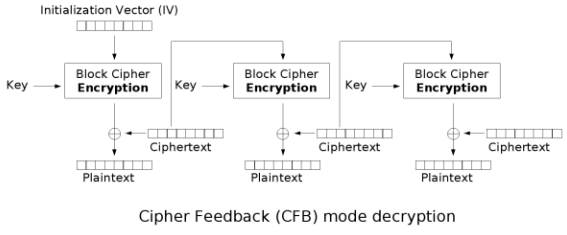
Gambar 6. Mode Dekripsi CBC

3. CFB (*Cipher Feedback*)
   Skema dari proses **ENKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



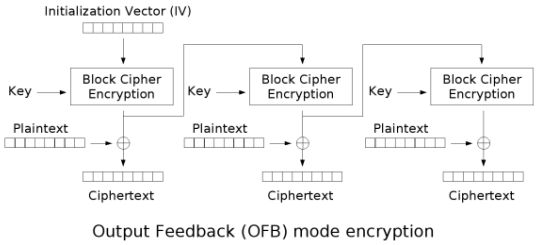
Gambar 7. Mode Enkripsi CFB

Skema dari proses **DEKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



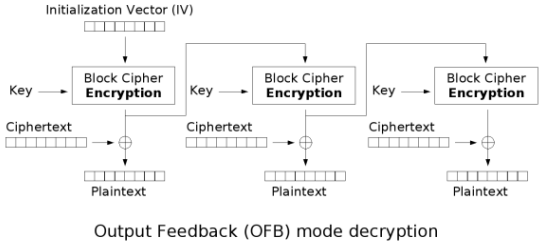
Gambar 8. Mode Dekripsi CFB

4. OFB (*Output Feedback*)
   Skema dari proses **ENKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



Gambar 9. Mode Enkripsi OFB

Skema dari proses **DEKRIPSI** menggunakan mode operasi ECB adalah sebagai berikut :



Gambar 10. Mode Dekripsi OFB

2. *Pretty Good Privacy*

2.1 Sejarah *Pretty Good Privacy*

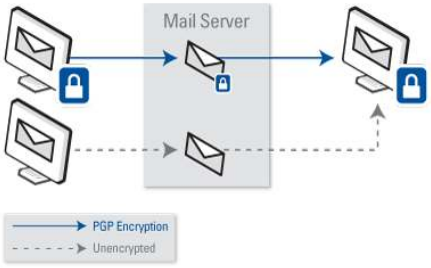
Pretty Good Privacy (PGP) merupakan salah satu cara untuk berkomunikasi melalui *email* (*electronic mail*) yang dipelopori oleh Philip R. Zimmerman pada tahun 1991. *Email* yang kita kirim akan dienkripsi isinya dan hanya dapat dibuka oleh penerima yang berhak.

Zimmerman menulis program enkripsi *email* PGP tersebut lalu disebar secara *freeware* ke internet. Namun Zimmerman ditahan selama tiga tahun karena dirasa telah melanggar peraturan pemerintah Amerika yang menyatakan dalam undang-undang bahwa dilarang menyebarkan teknologi enkripsi dan dekripsi. Dan selama proses investigasi PGP tersebut, PGP telah menyebar ke seluruh dunia hingga tahun 1996 pemerintah Amerika menutup kasus tersebut. Pada Desember 1997, PGP diakuisisi oleh *Network Associates* (NAI) dan PGP dikembangkan untuk menjadi sebuah proyek komersil.

Pada PGP terdapat dua versi kunci publik yaitu RSA (Rivest-Shamir-Adleman) yang dikembangkan sejak 1977 dan Diffie-Hellman. Pada PGP versi pertama menggunakan algoritma IDEA (*International Data Encryption Algorithm*) yang digunakan untuk men-generate kunci pendek dan mengenkripsi seluruh pesan kemudian mengenkripsi kunci pendek dengan algoritma RSA. Lalu versi kedua menggunakan algoritma CAST untuk meng-*generate* kunci pendek dari seluruh pesan untuk mengenkripsinya lalu kemudian menggunakan algoritma Diffie-Hellman untuk mengenkripsi kunci pendek tersebut.

2.2 Aplikasi *Pretty Good Privacy*

PGP biasanya digunakan untuk proses tanda tangan digital, selain untuk meng-enkripsi dan mendekripsi email untuk meningkatkan keamanan pada komunikasi melalui email. Jika biasanya dalam melakukan enkripsi pada konten email dan attachment menggunakan sebuah *desktop client*, produk PGP telah dikembangkan sejak tahun 2002 menjadi satu *set* aplikasi peng-enkripsi dimana bisa diatur dengan menggunakan sebuah *central policy server* yang fakultatif. Enkripsi PGP meliputi *email* dan *attachment*, tanda tangan digital, enkripsi *full disk* pada laptop, keamanan *file* dan folder, proteksi pada sesi IM, enkripsi transfer *batch file*, proteksi berkas dan folder yang disimpan pada server, dan mengenkripsi permintaan HTTP atau respon pada sisi *client* dan pada sisi *server*. Bahkan terdapat juga *plugin* pada Wordpress yang disebut wp-enigform-authentication.



Gambar 11. Enkripsi Email Menggunakan PGP

Terdapat juga PGP *Command Line*, dimana dapat digunakan untuk meng-*enable*-kan enkripsi berdasarkan *command line* dan informasi untuk penyimpanan, transfer, dan *backup*, seperti pada produk PGP Support Package untuk Blackberry dimana meng-*enable*-kan RIM Blackberry untuk menikmati enkripsi pengiriman data dari pengirim ke penerima. Hal inilah yang membuat Blackberry sering menjadi pilihan para pejabat tinggi negara sebagai alat untuk berkomunikasi karena terdapat proses enkripsi data. Meskipun belakangan beredar kabar bahwa proses enkripsi pada Blackberry telah dapat dipecahkan. PGP juga muncul pada sebuah lagu berjudul "*Guarded by Monkeys*" yang dinyanyikan oleh sebuah grup beraliran *rock* dari Amerika pada kalimat, "*i got your PGP key*". Enkripsi PGP juga digunakan dan direkomendasikan oleh karakter Lisbeth Salander di novel karangan Stieg Larsson yang berjudul *The Girl with The Dragon Tattoo*[8].

2.3 Cara Kerja *Pretty Good Privacy*

Pretty Good Privacy (PGP) menggunakan kombinasi serial dari metode *hashing*, kompresi data, kriptografi kunci simetri, dan kriptografi kunci publik. Setiap menggunakan salah satu dari algoritma yang mendukung. Setiap kunci publik terikat kepada sebuah *username* ataupun sebuah alamat *email*.

PGP ini nantinya akan menggunakan pasangan kunci privat dan kunci publik. Kunci privat merupakan kunci yang dipegang oleh oleh pengguna dan tidak boleh diketahui oleh orang lain. Sementara itu, kunci publik merupakan kebalikannya. Kunci publik ditujukan kepada publik terutama orang yang akan menerima pesan enkripsi tersebut.

Proses enkripsi menggunakan PGP tidak memakan waktu lama dan tidak memakan utilitas CPU. PGP akan mengenkripsi pesan dengan menggunakan kunci publik pemerima lalu mengenkripsi sebuah kunci pendek untuk mengenkripsi seluruh pesan. Kemudian seluruh pesan yang telah terenkripsi dengan kunci pendek dikirimkan ke penerima. Penerima akan membuka pesan tersebut menggunakan kunci privat yang dimiliki penerima. Kunci privat yang dimilik oleh penerima digunakan untuk mendapatkan kunci pendek tadi untuk selanjutnya digunakan untuk mendekripsi pesan.

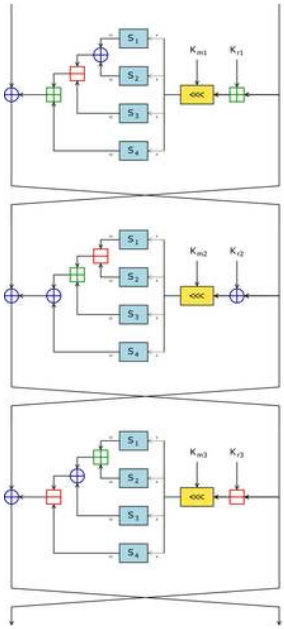
Untuk menyatakan keabsahan dari kunci dan pesan yang terenkripsi digunakan digital signature (tanda tangan digital). PGP meng-*generate* kode *hash* (kode yang menyatakan integritas sebuah data) dari informasi nama dan informasi lainnya. *Hash* yang dihasilkan kemudian dienkripsi dengan kunci privat. Penerima akan menggunakan kunci publik untuk mendekripsi kode *hash*. Jika terdapat kecocokan, maka kode *hash* tadi menjadi *digital signature* untuk pesan tersebut.

2. ALGORITMA CAST

Algoritma CAST merupakan algoritma yang serupa dengan algoritma Blowfish. Algoritma ini didesain oleh Stafford Adams dan Carlisle Adams, dan nama "CAST" merepresentasikan huruf pertama pada nama mereka.

Terdapat dua macam algoritma CAST, yaitu CAST-128 dan CAST-256. Algoritma CAST-128 menggunakan 12 atau 16- *round* jaringan Feistel dengan 64-bit ukuran blok dan ukuran kuncinya sekitar 40 hingga

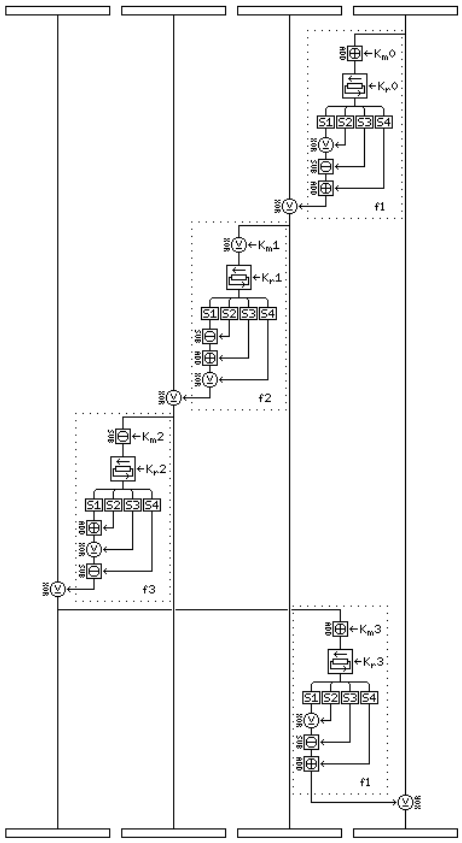
128 bit. 16 *round* penuh digunakan ketika kunci yang digunakan lebih panjang dari 80 bit. Komponen-komponennya termasuk di dalamnya sebuah 8x32-bit *S-boxes* besar yang berdasarkan *bent-functions*, rotasi kunci independen, *modular addition*, dan subtraksi, serta operasi XOR.



Gambar 12. CAST-128

Sementara itu algoritma CAST-256 adalah sebuah cipher simetri yang didesain berdasarkan prosedur mendesain CAST. Algoritma ini merupakan ektensi dari CAST-128 dan telah didaftarkan sebagai salah satu kandidat untuk NIST *Advanced Encryption Standard* (AES). Dalam mendesain algoritma CAST-256 ini, Howard Heys dan Michael Wiener juga turut berkontribusi.

CAST-256 menggunakan komponen-komponen yangs ama dengan CAST-128, termasuk *S-boxes* (yang diadaptasi dari *block* berukuran 128 bit). Panjang kunci yang dapat diterima adalah 128, 160, 224, atau 256 bit. CAST-256 menggunakan 48 putaran (*round*) yang sering disebut sebagai 12 "*quad-rounds*".



Gambar 13. Enkripsi Email Menggunakan PGP

3. ANALISIS

3.1 Prinsip Kerja Algoritma CAST dalam PGP

PGP versi 6.5 memiliki dua pilihan versi algoritma, semnetara di PGP versi 9.5 terdapat berbagai macam pilihan algoritma lain selain RSA dan CAST.

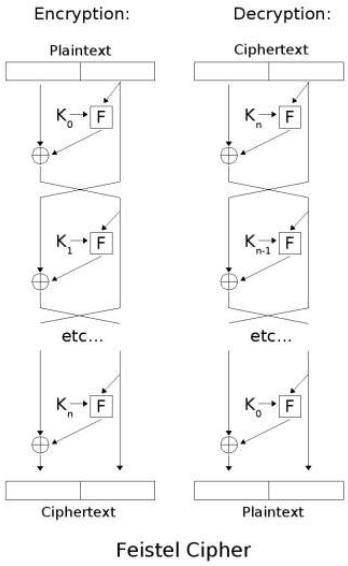
Panjang maksimum kunci yang digunakan oleh algoritma CAST dalam PGP adalah 128 bit. Algoritma dasar CAST adalah CAST menerima input plainteks m1 hingga m64, dan kunci k1 hingga k128. Dan CAST menghasilkan cipherteks c1 hingga c64 (menghasilkan 64 bit cipherteks).

INPUT : plainteks m1...m64;key K = k1...k128. OUTPUT : cipherteks c1...c64.

Pertama-tama, dilakukan penjadwalan kunci yang menghasilkan 16 pasang subkunci $\{K_{mi}, K_{ri}\}$ dari kunci K. K_{ri} berukuran 5 bit dan digunakan sebagai kunci untuk rotasi ke-i (*rotation key*) sementara K_{mi} digunakan untuk

rotasi ke i[10] (*masking key*). Adanya kunci *masking* dan kunci rotasi menaikkan entropi kunci dibandingkan dengan entropi data pada setiap putaran. Sehingga mempersulit jika terdapat serangan terhadap algoritma ini.

Kemudian dari blok-blok masukan plainteks-nya dibagi menjadi dua blok kanan dan blok kiri (R0 dan L0). Sebelah kiri L0= berisikan m1 hingga m32 dan sebelah kanan R0 berisikan m33 hingga m64. Pembagian blok ini bertujuan untuk membentuk jaringan feistel yang memudahkan adanya proses enkripsi dan dekripsi dan mengakibatkan susahnya konstruksi iteratif. Berikut merupakan skema jaringan feistel.



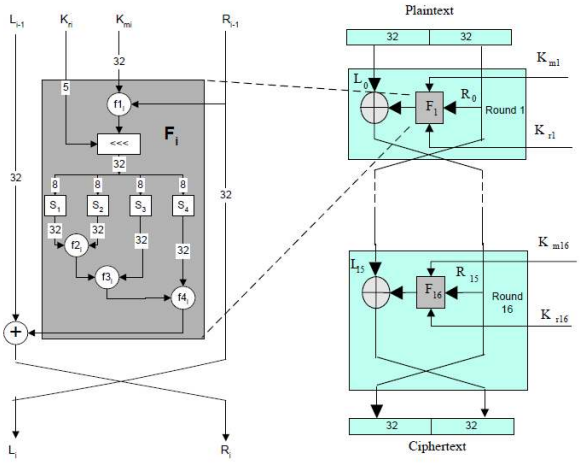
Gambar 14. Skema Jaringan Feistel

Selanjutnya dilakukan proses enkripsi sebanyak 16 putaran dan perhitungan untuk blok kanan dan kiri pada putaran ke I adalah sebagai berikut.

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \, ^{f(R_{i-1}, K_{mi}, K_{ri})}$$

Dan tahapan terakhir adalah menukarkan blok final (telah dienkripsi sebanyak 16 putaran) L16 dan R16 kemudian mengkonkatenasi hasil cipherteks pada blok kanan dan blok kiri. Berikut merupakan skema enkripsi dengan algoritma CAST.



Gambar 15. Skema Enkripsi Algoritma CAST

CAST-128 didesain untuk mengijinkan adanya variansi ukuran kunci dari 40 bit hingga 128 bit. Spesifikasi ukuran kunci adalah:

1. Untuk ukuran kunci yang masih berada pada *range* 80 bit ke bawah, melibatkan 12 putaran proses enkripsi.
2. Untuk ukuran kunci yang lebih banyak dari 80 bit, menggunakan 16 putaran penuh proses enkripsi.
3. Dan untuk ukuran kunci kurang dari 128 bit, kunci tersebut di-*padding* dengan *zero bytes* (pada posisi *rightmost* atau *least significant*).

Seperti contohnya adalah sebagai berikut.

```
128-bit    key         = 01 23 45 67 12 34 56 78 23 45 67 89 34 56 78 9A
           plaintext   = 01 23 45 67 89 AB CD EF
           ciphertext  = 23 8B 4F E5 84 7E 44 B2


80-bit     key         = 01 23 45 67 12 34 56 78 23 45
                       = 01 23 45 67 12 34 56 78 23 45 00 00 00 00 00 00
           plaintext   = 01 23 45 67 89 AB CD EF
           ciphertext  = EB 6A 71 1A 2C 02 27 1B

40-bit     key         = 01 23 45 67 12
                       = 01 23 45 67 12 00 00 00 00 00 00 00 00 00 00 00
           plaintext   = 01 23 45 67 89 AB CD EF
           ciphertext  = 7A C8 16 D1 6E 9B 30 2E
```

Pada masing-masing rotasi, terdapat tiga tipe fungsi matematika yang digunakan, yaitu [5]:

- Tipe 1

$$I = ((k_{m_i} + D) \lrcorner k_{r_i})$$
$$O = ((S_1[I_a] \oplus S_2[I_b]) - S_3[I_c]) + S_4[I_d]$$

- Tipe 2

$$I = ((k_{m_i} \oplus D) \lrcorner k_{r_i})$$
$$O = ((S_1[I_a] - S_2[I_b]) + S_3[I_c]) \oplus S_4[I_d]$$

- Tipe 3

$$I = ((k_{m_i} - D) \lrcorner k_{r_i})$$
$$O = ((S_1[I_a] + S_2[I_b]) \oplus S_3[I_c]) - S_4[I_d]$$

Keterangan:

- D = Data *input* untuk operasi
- I_a-I_d = *byte* yang paling signifikan dari I
- Si = S-Box ke-i
- O = *Output* dari operasi
- +/- = penambahan dan pengurangan dengan modulo 232
- ⊕ = operasi bitwise *eXclusive*-OR
- ↵ = operasi *left-shift*

Pada PGP, algoritma CAST ini akan berhubungan dengan protokol. Dan jika akan berhubungan dengan protokol, diperlukan adanya OBJECT IDENTIFIERs (OIDs). Komponen *S-box* yang dirancang pada CAST dengan prosedur matematik menghasilkan adanya rendahnya distribusi perbedaan XOR dan pengurutan yang baik dengan menggunakan kriteria *bit* yang terpisah.

3.2 Tingkat Keamanan CAST

Algoritma CAST memiliki tingkat keamanan yang cukup tinggi dalam beberapa hal dibandingkan dengan algoritma yang digunakan pada PGP versi sebelumnya (IDEA). Hal tersebut dikarenakan algoritma CAST menggunakan kunci *masking* dan kunci *rotation*, menggunakan operasi aljabar yang berbeda dan menggunakan jaringan feistel. Dan terutama karena adanya *S-box*, yang sudah pasti akan menambah tingkat keamanan algoritma CAST ini.

Selain itu, pada PGP yang menggunakan asumsi bahwa keamanan panjang logika dikihat dari panjangnya kunci, maka algoritma CAST ini juga dianggap lebih aman dan menghasilkan pesan dan tandatangan yang lebih pendek dibandingkan algoritma versi sebelumnya.

Pada PGPDisk versi –versi terdahulu terdapat beberapa *bug* yang pada versi-versi terbaru telah diperbaiki. PGPDisk merupakan program yang memungkinkan pengguna untuk melakukan enkripsi pada *hard disk* tanpa mempunyai *passphrase*. *Bug* yang terdapat pada PGPDisk merupakan bug yang terletak pada fungsi jadwal kunci pada enkripsi CAST. Hal tersebut karena CAST dapat membangkitkan himpunan subkunci dari kunci dengan ukuran 1024 bit. Kunci tersebut hanya disalin ke buffer sehingga 128 bit pertama dari subkunci yang akan diinisiasi, sedangkan yang bit-bit lainnya dibiarkan bernilai 0. Oleh karena itu, enkripsi hanya baik untuk dua putaran CAST sehingga kunci relatif mudah ditebak jika seseorang mengetahui sebagian dari plainteks[4].

4. KESIMPULAN

Algoritma CAST merupakan algoritma kunci simetri yang menggunakan metode *block cipher* yang digunakan untuk meng-enkripsi dan men-*generate* kunci pada PGP. Terdapat dua macam algoritma CAST, yaitu CAST-128 dan CAST-256 yang merupakan ekstensi dari CAST-128. Algoritma CAST menggunakan 16 putaran dalam melakukan enkripsi. Panjang kunci yang diterima oleh PGP sepanjang 128 bit. Algoritma CAST menggunakan struktur jaringan feistel dalam pengimplementasiannya untuk mempermudah adanya proses enkripsi dan proses dekripsi serta untuk meningkatkan keamanan sehingga sulit untuk di-*hack*.

PGP menggunakan asumsi bahwa keamanan dinilai dari tingkat kerumitan panjang kunci. Oleh karena itu, algoritma CAST dinilai cukup aman dibandingkan dengan algoritma IDEA pada PGP versi sebelumnya.

REFERENSI

[1] Adri, Muhammad. "Block Cipher - lanjutan", 2008, halaman 9.

[2] Hirani ,Sohail. "Energy Consumption of Encryption Schemes in Wireless Devices", 2003, halaman 26.

[3] Munir, Rinaldi. 2004. "Kriptografi dalam Kehidupan Sehari-hari (Bagian 2)", 2004, halaman 13-24.

[4] Tanoto, Andri. "Analisis Keamanan pada *Pretty Good Privacy* (PGP)", 2003, halaman 12.

[5] Viqarunnisa, Pocut. "STUDI DAN ANALISIS PERBANDINGAN KEAMANAN PGP ALGORITMA IDEA-RSA DENGAN PGP ALGORITMA CAST-DH", 2003, halaman 9-10.

[6] http://www.encryptfiles.net/encryption/algorithm/cast.php , diakses pada Kamis, 18 Maret 2010 pukul 16.10 WIB.

[7] http://en.citizendium.org/wiki/CAST_%28cipher%29 , diakses pada Kamis, 18 Maret 2010 pukul 16.05 WIB.

[8] http://en.wikipedia.org/wiki/Cryptography, diakses pada Kamis, 18 Maret 2010 pukul 16.10 WIB.

[9] http://id.wikipedia.org/wiki/Kriptografi , diakses pada Kamis, 18 Maret 2010 pukul 18.00 WIB.

[10] http://jya.com/cast.html, diakses pada Jum'at, 19 Maret 2010 pukul 14.00 WIB.

[11] http://www.mccune.cc/PGPpage2.htm, diakses pada Jum'at, 19 Maret 2010 pukul 14.10 WIB.

[12] http://my.opera.com/adpermadi/blog/show.dml/2164995 , diakses pada Kamis, 18 Maret 2010 pukul 16.20 WIB.

[13] http://www.quadibloc.com/crypto/co040410.htm, diakses pada Kamis, 18 Maret 2010 pukul 16.22 WIB.

[14] http://www.scramdisk.clara.net/pgpfaq.html, diakses pada Kamis, 18 Maret 2010 pukul 15.45 WIB.