Pengamanan Komputer Menggunakan Kriptografi *CIPHER BLOCK CHAINING* (CBC)

Alcianno G. Gani
localghost2000@yahoo.com

ABSTRACT

In cryptography, a mode of operation is an algorithm that uses a block cipher to encrypt messages of arbitrary length in a way that provides confidentiality or authenticity. A block cipher by itself is only suitable for the secure cryptographic transformation (encryption or decryption) of one fixed-length group of bits called a block. A mode of operation describes how to repeatedly apply a cipher's single-block operation to securely transform amounts of data larger than a block.

Most modes require a unique binary sequence, often called an Initialization Vector (IV), for each encryption operation. The IV has to be non-repeating and, for some modes, random as well. The initialization vector is used to ensure distinct ciphertexts are produced even when the same plaintext is encrypted multiple times independently with the same key. Block ciphers have one or more block size(s), but during transformation the block size is always fixed. Block cipher modes operate on whole blocks and require that the last part of the data be padded to a full block if it is smaller than the current block size. There are, however, modes that do not require padding because they effectively use a block cipher as a stream cipher; such ciphers are capable of encrypting arbitrarily long sequences of bytes or bits.

Historically, encryption modes have been studied extensively in regard to their error propagation properties under various scenarios of data modification. Later development regarded integrity protection as an entirely separate cryptographic goal. Some modern modes of operation combine confidentiality and authenticity in an efficient way, and are known as authenticated encryption modes.

Keywords : cryptography, cipher, cipher blok chaining, security, network, security network

PENDAHULUAN

Komputer merupakan suatu sistem yang saling berkaitan antara input, proses, dan output. Oleh Karena itu jika salah satu saja mengalami kerusakan atau dalam keadaan sakit, Komputer tersebut akan mengalami gangguan bahkan dapat pula komputer tersebut tidak bisa digunakan. Terutama sekali dengan peralatan proses (CPU) yang menjadi tumpuan dari sebuah komputer, Untuk menanggulangi hal tersebut, dalam bab ini akan diuraikan tentang pengamanan komputer, ada baiknya kita mengetahui definisi dari pengamanan komputer.

Pengamanan komputer atau yang sering disebut dengan sekuriti komputer adalah pencegahan atas penggunaan data atau program dari masalah yang akan dihadapi, atau lebih mudahnya adalah segala sesuatu yang menyangkut masalah keamanan sistem komputer. Jadi, pengamanan komputer adalah segala sesuatu baik berupa alat-alat, program komputer sampai dengan manusianya untuk saling menjaga dan mencegah komputer dari kerusakan. Karena itu, ketiga komponen tersebut harus saling mengisi satu dengan yang lainnya. Ini disebabkan keterbatasan kemampuan salah satu komponen tersebut. Namun demikian, komponen manusia merupakan komponen terbesar, ini karena alat-alat dan program komputer dapat berfungsi dengan baik jika dioperasikan dengan benar. Selain itu, lingkungan yang paling dekat dengan komputer alah dari faktor manusia, sehingga manusia haruslah dapat

berperan banyak dalam pengamanan komputer.

KEAMANAN JARINGAN KOMPUTER

Keamanan jaringan komputer sendiri sering dipandang sebagai hasil dari beberapa faktor. Faktor ini bervariasi tergantung pada bahan dasar, tetapi secara normal setidaknya beberapa hal dibawah ini diikutsertakan :

- *Availability* (ketersediaan).
- *Confidentiality* (kerahasiaan).
- *Integrity* (integritas).

Availability (ketersediaan)

Ketersediaan data atau layanan dapat dengan mudah dipantau oleh pengguna dari sebuah layanan. Yang dimana ketidaktersediaan dari sebuah layanan (*service*) dapat menjadi sebuah halangan untuk maju bagi sebuah perusahaan dan bahkan dapat berdampak lebih buruk lagi, yaitu penghentian proses produksi. Sehingga untuk semua aktifitas jaringan, ketersediaan data sangat penting untuk sebuah system agar dapat terus berjalan dengan benar.

Confidentiality (kerahasiaan)

Ada beberapa jenis informasi yang tersedia didalam sebuah jaringan komputer. Setiap data yang berbeda pasti mempunyai grup pengguna yang berbeda pula dan data dapat dikelompokkan sehingga beberapa pembatasan kepada pengunaan data harus ditentukan. Pada umumnya data yang terdapat didalam suatu perusahaan bersifat rahasia dan tidak boleh diketahui oleh pihak ketiga yang bertujuan untuk menjaga rahasia perusahaan dan strategi perusahaan.

Kerahasiaan dapat ditingkatkan dan didalam beberapa kasus pengengkripsian data atau menggunakan VPN. Topik ini tidak akan, tetapi bagaimanapun juga, akan disertakan dalam tulisan ini. Kontrol akses adalah cara yang lazim digunakan untuk membatasi akses kedalam sebuah jaringan komputer. Sebuah cara yang mudah tetapi mampu untuk membatasi akses adalah dengan menggunakan kombinasi dari *username-dan-passwords* untuk proses otentifikasi pengguna dan memberikan akses kepada pengguna (*user*) yang telah dikenali. Didalam beberapa lingkungan kerja keamanan jaringan komputer, ini dibahas dan dipisahkan dalam konteks otentifikasi.

Integrity (integritas)

Jaringan komputer yang dapat diandalkan juga berdasar pada fakta bahwa data yang tersedia apa yang sudah seharusnya. Jaringan komputer mau tidak mau harus terlindungi dari serangan (*attacks*) yang dapat merubah data selama dalam proses persinggahan (*transmit*). *Man-in-the-Middle* merupakan jenis serangan yang dapat merubah integritas dari sebuah data yang mana penyerang (*attacker*) dapat membajak "*session*" atau memanipulasi data yang terkirim.

Didalam jaringan komputer yang aman, partisipan dari sebuah "transaks*i*" data harus yakin bahwa orang yang terlibat dalam komunikasi data dapat diandalkan dan dapat dipercaya. Keamanan dari sebuah komunikasi data sangat diperlukan pada sebuah tingkatan yang dipastikan data tidak berubah selama proses pengiriman dan penerimaan pada saat komunikasi data. Ini tidak harus selalu berarti bahwa "*traffic*" perlu di enkripsi, tapi juga tidak tertutup kemungkinan serangan "*Man-in-the-Middle*" dapat terjadi.

Nonrepudiation (tanpa penyangkalan)

Setiap tindakan yang dilakukan dalam sebuah system yang aman telah diawasi (*logged*), ini dapat berarti

penggunaan alat (*tool*) untuk melakukan pengecekan system berfungsi sebagaimana seharusnya. "*Log*" juga tidak dapat dipisahkan dari bagian keamanan "*system*" yang dimana bila terjadi sebuah penyusupan atau serangan lain akan sangat membantu proses investigasi. "*Log*" dan catatan waktu, sebagai contoh, bagian penting dari bukti di pengadilan jika *cracker* tertangkap dan diadili. Untuk alasan ini maka "*nonrepudiation*" dianggap sebagai sebuah faktor penting didalam keamanan jaringan komputer yang berkompeten.

ITU-T telah mendefinisikan "*non-repudition*" sebagai berikut :

1. Kemampuan untuk mencegah seorang pengirim untuk menyangkal kemudian bahwa dia telah mengirim pesan atau melakukan sebuah tindakan.
2. Proteksi dari penyangkalan oleh satu satu dari entitas yang terlibat didalam sebuah komunikasi yang turut serta secara keseluruhan atau sebagian dari komunikasi yang terjadi.

Jaringan komputer dan system data yang lain dibangun dari beberapa komponen yang berbeda yang dimana masing-masing mempunyai karakteristik spesial untuk keamanan. Sebuah jaringan komputer yang aman perlu masalah keamanan yang harus diperhatikan disemua sektor, yang mana rantai keamanan yang komplit sangat lemah, selemah titik terlemahnya. Pengguna (*user*) merupakan bagian penting dari sebuah rantai. "*Social engineering*" merupakan cara yang efisien untuk mencari celah (*vulnerabilities*) pada suatu system dan kebanyakan orang menggunakan "*passwords*" yang mudah ditebak. Ini juga berarti meninggalkan "*workstation*" tidak dalam keadaan terkunci pada saat makan siang atau yang lainnya.

Layanan pada "*server*" memainkan peranan penting dalam keamanan. *Developer* perangkat lunak mengumumkan celah keamanan pada perangkat lunak dengan cepat. Alasan yang digunakan adalah celah ini kemungkinan akan digunakan oleh pihak yang tidak bertanggung jawab untuk menyusupi sebuah sistem ataupun setiap pengguna komputer. Pengelola atau pengguna *server* dan *workstation* harus melakukan pengecekan untuk "*update*" masalah keamanan secara regular.

Pemilihan jenis metode transmisi juga mempunyai peranan penting didalam masalah keamanan. Setiap informasi rahasia tidak boleh di transmisikan secara *wireless*, setidaknya tidak tanpa menggunakan enkripsi yang bagus, sehingga setiap orang dapat menyadap komunikasi *wireless* yang terkirim. Sangat dianjurkan untuk menggunakan *firewall* untuk membatasi akses kedalam jaringan komputer ke tingkat yang dibutuhkan. *Firewall* juga dapat menjadi titik terlemah, yang mana dapat membuat perasaan aman. *Firewall* harus mengizinkan arus data kedalam sebuah jaringan komputer jika terdapat juga arus data keluar dari jaringan komputer tersebut melalui firewall dan ini dapat menjadi titik terlemah. Fakta penting lainnya bahwa tidak semua serangan dilancarkan melalui *firewall*.

KEAMANAN KOMPUTER (*COMPUTER SECURITY*)

Knowing the Laws of Security (Russel, Ryan):

1. *Client-Side Security doesn't work.*
2. *You cannot securely exchange encryption keys without a shared piece of information.*

3. *Malicious code cannot be 100 percent protected against.*
4. *Any malicious code can be completely morphed to bypass signature detection.*
5. *Firewalls cannot protect you 100 percent from attack.*
6. *Any intrusion detection system (IDS) can be evaded.*
7. *Secret cryptographic algorithm are not secure.*
8. *If a key isn't required, you do not have encryption - you have encoding.*
9. *Passwords cannot be securely stored on the client unless there is another passwords to protect them.*
10. *In order for a system to begin to be considered secure, it must undergo an independent security audit.*
11. *Security through obscurity does not work*.

The Ten Immutable Laws of Security (www.microsoft.com/technet/ columns/ security/10imlaws.asp) :

1. *If a bad guy can persuade you to run his program on your computer, it's not your computer anymore.*
2. *If a bad guy can alter the operating system on your computer, it's not your computer anymore.*
3. *If a bad guy has unrestricted physical access to your computer, it's not your computer anymore.*
4. *If you allow a bad guy to upload program to your website, it's not your website anymore.*
5. *Weak passwords trump strong security.*
6. *A machine is only as secure as the administrator is trust-worthy.*
7. *Encrypted data is only as secure as the decryption key.*
8. *An out-dated virus scanner is only marginally better than no virus scanner at all.*
9. *Absolute anonymity isn't practical, in real life or on the web.*
10. *Technology is not a panacea.*

Mengamankan jaringan komputer membutuhkan tiga tingkatan proses. Untuk mengamankan jaringan komputer kita harus dapat melakukan pemetaan terhadap ancaman yang mungkin terjadi.

Prevention

Kebanyakan dari ancaman akan dapat ditepis dengan mudah, walaupun keadaan yang benar-benar 100% aman belum tentu dapat dicapai. Akses yang tidak diinginkan kedalam jaringan komputer dapat dicegah dengan memilih dan melakukan konfigurasi layanan (*services*) yang berjalan dengan hati-hati.

Observation

Ketika sebuah jaringan komputer sedang berjalan, dan sebuah akses yang tidak diinginkan dicegah, maka proses perawatan dilakukan. Perawatan jaringan komputer harus termasuk melihat isi log yang tidak normal yang dapat merujuk ke masalah keamanan yang tidak terpantau. System IDS dapat digunakan sebagai bagian dari proses observasi tetapi menggunakan IDS seharusnya tidak merujuk kepada ketidak-pedulian pada informasi log yang disediakan.

Response

Bila sesuatu yang tidak diinginkan terjadi dan keamanan suatu system telah berhasil disusupi, maka personil perawatan harus segera mengambil tindakan. Tergantung pada proses produktifitas dan masalah yang menyangkut dengan keamanan maka tindakan yang tepat harus segera dilaksanakan. Bila sebuah proses sangat vital pengaruhnya kepada fungsi system dan apabila di-*shutdown* akan menyebabkan lebih banyak kerugian daripada membiarkan system yang telah berhasil disusupi tetap dibiarkan

berjalan, maka harus dipertimbangkan untuk direncakan perawatan pada saat yang tepat. Ini merupakan masalah yang sulit dikarenakan tidak seorangpun akan segera tahu apa yang menjadi celah begitu sistem telah berhasil disusupi dari luar.

Victims/statistic

Keamanan jaringan komputer meliputi beberapa hal yang berbeda yang mempengaruhi keamanan secara keseluruhan. Serangan keamanan jaringan komputer dan penggunaan yang salah dan sebegai contoh adalah virus, serangan dari dalam jaringan komputer itu sendiri, pencurian perangkat keras (*hardware*), penetrasi kedalam system, serangan *Denial of Service* (DoS), *sabotase*, serangan *wireless* terhadap jaringan komputer, penggantian halaman depan situs (*website defacement*), dan penggunaan yang salah terhadap aplikasi *web.*

Statistik menunjukkan jumlah penyusupan didalam area ini sudah cukup banyak berkurang dari tahun 2003, tipe variasi dari serangan, bagaimanapun juga, menyebabkan hampir setiap orang adalah sasaran yang menarik.

TIPE-TIPE ANCAMAN SERANGAN

Tujuan utama dengan adanya keamanan adalah untuk membatasi akses informasi dan sesumber hanya untuk pemakai yang memiliki hak akses.

Ancaman keamanan :

- *Leakage* (Kebocoran) : Pengambilan informasi oleh penerima yang tidak berhak.
- *Tampering* : Pengubahan informasi yang tidak legal.
- *Vandalism* (perusakan) : Gangguan operasi sistem tertentu. Si pelaku tidak mengharap keuntungan apapun.
- Serangan pada sistem terdistribusi tergantung pada pengaksesan ke saluran komunikasi yang ada atau membuat saluran baru yang menyamarkan (*masquerade*) sebagai koneksi legal.
- Penyerangan Pasif, Hanya mengamati komunikasi atau data.
- Penyerangan Aktif, Secara aktif memodifikasi komunikasi atau data.
- Pemalsuan atau pengubahan Email.
- TCP/IP Spoofing.

Ancaman Jaringan Komputer :

- FISIK
  - Pencurian perangkat keras komputer atau perangkat jaringan.
  - Kerusakan pada komputer dan perangkat komunikasi jaringan
  - Wiretapping.
  - Bencana alam.

- LOGIK
  - Kerusakan pada sistem operasi atau aplikasi.
  - Virus.
  - Sniffing.

BEBERAPA METODE PENYERANGAN

Scanning

"*Scanning*" adalah metode bagaimana caranya mendapatkan informasi sebanyak-banyaknya dari IP/Network korban. Biasanya "scanning" dijalankan secara otomatis mengingat "*scanning*" pada "*multiple-host*" sangat menyita waktu. "*Hackers*" biasanya mengumpulkan informasi dari hasil "*scanning*" ini. Dengan mengumpulkan informasi yang dibutuhkan maka "*hackers*" dapat menyiapkan serangan yang akan dilancarkannya.

Nmap, merupakan sebuah *network scanner* yang banyak digunakan oleh para professional di bidang *network*

security, walaupun ada tool yang khusus dibuat untuk tujuan *hacking*, tapi belum dapat mengalahkan kepopuleran *nmap.*

Nessus, juga merupakan *network scanner* tapi juga akan melaporkan apabila terdapat celah keamanan pada target yang diperiksanya. *Hacker* biasanya menggunakan *Nessus* untuk pengumpulan informasi sebelum benar-benar meluncurkan serangan.

Untungnya beberapa scanner meninggalkan "jejak" yang unik yang memungkinkan para *System administrator* untuk mengetahui bahwa *system* mereka telah di-*scanning* sehingga mereka bisa segera membaca artikel terbaru yang berhubungan dengan informasi log.

Passwords cracking

"*Brute-force*" adalah sebuah tehnik dimana akan dicobakan semua kemungkinan kata kunci (*passwords*) untuk bisa ditebak untuk bisa mengakses kedalam sebuah system. Membongkar kata kunci dengan tehnik ini sangat lambat tapi efisien, semua kata kunci dapat ditebak asalkan waktu tersedia.

Untuk membalikkan "*hash*" pada kata kunci merupakan suatu yang hal yang mustahil, tapi ada beberapa cara untuk membongkar kata kunci tersebut walaupun tingkat keberhasilannya tergantung dari kuat lemahnya pemilihan kata kunci oleh pengguna. Bila seseorang dapat mengambil data "*hash*" yang menyimpan kata kunci maka cara yang lumayan efisien untuk dipakai adalah dengan menggunakan metode "*dictionary attack*" yang dapat dilakukan oleh utility John The Ripper.

Masih terdapat beberapa cara lainnya seperti "*hash look-up table*" tapi sangat menyita "*resources*" dan waktu.

Rootkit

"*Rootkit*" adalah alat untuk menghilangkan jejak apabila telah dilakukan penyusupan*. Rootkit* biasanya mengikutkan beberapa tool yang dipakai oleh *system* dengan sudah dimodifikasi sehingga dapat menutupi jejak. Sebagai contoh, memodifikasi "PS" di *linux* atau *unix* sehingga tidak dapat melihat *background process* yang berjalan.

BEBERAPA BENTUK ANCAMAN KEAMANAN (*SECURITY THREAT*)

- *Sniffer, p*eralatan yang dapat memonitor proses yang sedang berlangsung.
- *Spoofing*, penggunaan komputer untuk meniru (dengan cara menimpa identitas atau alamat IP).
- *Phreaking,* Perilaku menjadikan sistem pengamanan telepon melemah.
- *Remote Attack*, segala bentuk serangan terhadap suatu mesin dimana penyerangnya tidak memiliki kendali terhadap mesin tersebut karena dilakukan dari jarak jaruh di luar sistem jaringan atau media transmisi.
- *Hole*, Kondisi dari software atau hardware yang bisa diakses oleh pemakai yang tidak memiliki otoritas atau meningkatnya tingkat pengaksesan tanpa melalui proses autorisasi.
- *Hacking*, bentuk kegiatan yang secara diam-diam mempelajari sistem yang biasanya sukar dimengerti untuk kemudian mengelolanya dan men-*share* hasil ujicoba yang dilakukannya. *Hacking* tidak merusak sistem.
- *Cracking*, bentuk kegiatan yang secara diam-diam mempelajari sistem dengam maksud jahat salah satunya merusak.

- *Eavesdropping*, mendapatkan duplikasi pesan tanpa ijin.
- *Masquerading*, Mengirim atau menerima pesanmenggunakan identitas lain tanpa ijin mereka.
- *Message tampering*, mencegat atau menangkap pesan dan mengubah isinya sebelum dilanjutkan ke penerima sebenarnya. "*man-in-the-middle attack*" adalah bentuk *message tampering* dengan mencegat pesan pertama pada pertukaran kunci enkripsi pada pembentukan suatu saluran yang aman. Penyerang menyisipkan kunci lain yang memungkinkan dia untuk mendekrip pesan berikutnya sebelum dienkrip oleh penerima.
- *Replaying*, menyimpan pesan yang ditangkap untuk pemakaian berikutnya.
- *Denial of Service***,** membanjiri saluran atau sesumber lain dengan pesan yang bertujuan untuk menggagalkan pengaksesan pemakai lain.

Penilaian terhadap segala bentuk Ancaman (threat) :

– FISIK
- Hardware
- Perangkat Jaringan
- Perangkat komunikasi data

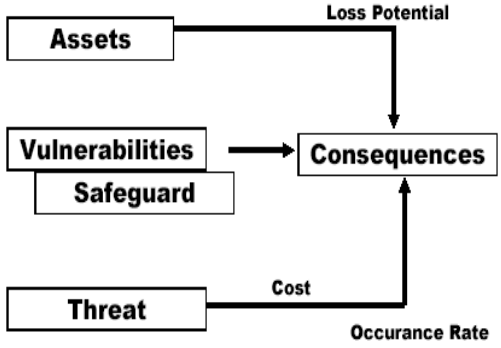
→
- Pencurian
- Kerusakan Fisik
- Wiretapping
- Bencana Alam

– LOGIK
- Aplikasi
- Sistem Operasi
- Data dan Informasi

→
- Kerusakan Logik
- Virus
- Sniffing
- Denial of Service

Pengumpulan Informasi
- Penilaian terhadap bagian yang berpotensi terkena gangguan (vulnerability).
- Penilaian terhadap perlindungan yang effektif (*safeguard*).
  – keamanan fasilitas fisik jaringan.
  – keamanan perangkat lunak.
  – keamanan pengguna jaringan.
  – keamanan komunikasi data.
  – keamanan lingkungan jaringan.

Analisis



Pengganggu komputer banyak terjadi karena faktor manusia, yaitu *Identification* dan *Authentication*. Untuk mengamankan keduanya, Kita dapat mengikuti cara berikut ini :

1. *Identification*

Passwords dapat diibaratkan seperti sikat gigi yang digunakan setiap hari. Oleh Karena itu, harus mengganti *passwords* tersebut secara periode dan jangan digunakan oleh orang lain. *Passwords* menjadi tanggung jawab setiap orang (Pemilik), sehingga kita dapat mengikuti cara-cara di bawah ini agar *passwords* lebih terjamin.

- Jangan biarkan login tanpa *passwords*, Jika kita bekerja dengan jaringan dan kita adalah seorang administrator sistem, pastikan setiap *account* mempunyai *passwords.*
- Jangan pernah membiarkan seseorang menggunakan *passwords* kita, Jika kita sudah terlanjur memberitahukan kepada orang lain, segeralah mengganti *passwords* dengan yang baru.
- Janganlah menulis *passwords* pada layar monitor, meja, atau sekitar ruang kerja.

- Jangan mengetik *passwords,* sementara dibelakang atau sekeliling komputer kita ada orang lain yang mengawasi.
- Jangan mengirimkan *passwords* secara *online* ke suatu tempat melalui *e-mail*, karena ada kemungkinan orang lain akan menyadap saluran *e-mail* anda.

Apabila diperbolehkan memilih *passwords*, pilihlah *passwords* yang sukar ditebak. Dibawah ini saran-saran untuk menentukan nama *passwords*, yaitu :

- Jangan menggunakan kata-kata dalam bahasa Inggris.
- Jangan menggunakan nama-nama, seperti nama sendiri atau keluarga, pahlawan fiktif, anggota keluarga, hewan piaraan dan lain-lain.
- Boleh juga menggunakan kata-kata yang tidak mempunyai arti, misalnya Jt93gpy.
- Sebaiknya gunakan gabungan huruf dan angka.
- Jangan menggunakan nomor telepon anda.
- Pilih *passwords* yang panjang, karena jika *passwords* anda hanya beberapa huruf atau angka atau kombinasi keduanya, akan mu-dah ditemukan. Gunakan minimal 6 - 8 karakter.
- Apabila anda bekerja dengan jaringan, sebaiknya bedakan *passwords* antara *host* (Komputer) yang satu dengan yang lain.
- *Passwords* yang baik adalah yang menggunakan kombinasi huruf besar dan kecil.

2. *Authentication*

- Proses pengenalan peralatan, sistem operasi, kegiatan, aplikasi dan identitas user yang terhubung dengan jaringan komputer.
- Autentikasi dimulai pada saat user login ke jaringan dengan cara memasukkan *passwords*.
- Jangan pernah meninggalkan kartu pengenal atau kunci di tempat terbuka, walaupun hanya sebentar.
- Tempatkan kartu pengenal atau kunci pada tempat yang sulit dijangkau oleh orang lain, atau letakkan pada tempat yang dapat anda kunci dari luar.
- Pada beberapa negara maju pengamanan kmputer telah menggunakan sensor untuk mengamankan komputer. Oleh karena itu, jangan pernah merekam sidik jari atau telapak tangan atau suara pada komputer anda karena akan mudah bagi oarang lain untuk membuat duplikatnya.

Bila anda seorang administrator sistem, sebaiknya anda membagi file-file tersebut menjadi beberapa tingkatan, yaitu :

- Siapa yang boleh membaca *file* anda.
- Siapa yang boleh mengubah *file* anda.
- Data anda di share (mendapat bagian yang sama) dengan *user* yang lain.

Dalam pengaturan akses terhadap file-file terdapat 2 tipe, yaitu :

1) Discretionary Access Control (DAC).
Pembatasan akses terhadap *file, directory* dan *device* berdasarkan user atau group. Pengaturan akses ini dilakukan oleh pe-miliknya. Pembahasan dengan tipe ini dibagi menjadi 3 bagian yang mendasar, yaitu *Read, Write,* dan *Execute*. Pembatasan akses kontrol dengan tipe *Discretionary Access Control* mempunyai beberapa jenis, yaitu :

a. *Ownership.*
   - Pembuatan *file* dilakukan oleh pemilik.

- Login atau beberapa pengenal disimpan dalam *file*.
- Apabila anda pemilik *file* tersebut, anda dapat membaca dan mengubah isi *file*.
- Jika anda bukan pemilik *file* tersebut, anda tidak dapat mengakses *file*-nya.

b. *File Types and File Protection Classes*
   - Metode ini lebih baik dibandingkan metode *Ownership*.
   - Sebuah *file* dapat didefinisikan sebagai *public, semi-public* atau *private file*. Untuk mendefinisikan, anda dapat menggunakan beberapa kode, yaitu:
     - o *Blank.* Digunakan untuk mendefinisikan *Public*, yaitu semua user dapat membaca atau menulis ke dalam file yang ber-sangkutan.
     - o @. Digunakan untuk mendefiniskan *Execute Only*, yaitu hanya pemilik *file, Administrator System* dan semua user saja dapat menjalankan file dan mengubah isi *file.*
     - o S. Digunakan untuk mendefiniskan *Read Only*, yaitu semua user dapat membaca dan menjalankan *file*, namun hanya pemilik *file* dan *Administrator System* yang dapat mengubah *file.*
     - o #. Digunakan untuk mendefinisikan *Private,* yaitu hanya pemilik *file* dan *Administrator System* yang dapat membaca, mengubah dan menjalankan *file.*
     - o A - Z. Digunakan untuk mendefiniskan *System Dependent,* yaitu *Administrator System* dapat men-*setting* sistem sehingga *user* yang mempunyai kelas tertentu yang dapat mengakses *file* tertentu pula, Misalnya kelas A dapat mengakses program Akuntansi, Kelas P dapat mengakses program *Payroll* dan lain-lain

c. *Self/Group/Public Controls*
   - Pengaturannya menggunakan 3 kategori, yaitu :
     - *Self,* digunakan oleh anda sendiri sebagai pembuat atau pemilik *file.*
     - Group, digunakan oleh sekelompok *user*.
     - Public, digunakan yang tidak termasuk *self* dan *group*.
   - Cara seperti ini digunakan pada sistem operasi UNIX, yang dikenal dengan nama UGO (*User/Group/Other*).
   - Setiap *file* mempunyai sekumpulan bit-bit yang disebut dengan *file permissions.*

d. *Access Data List*
   - Pembatasan *file* dengan ca-ra membuat daftar user-user dan group-group dengan haknya masing-masing.
   - Cara ini lebih fleksibel dibandingkan dengan cara sebelumnya.

2) Mandatory Access Control (MAC)
Pembatasan akses yang ditentukan oleh sistem. Pengaturan akses dengan menggunakan *Mandatory Access Control* lebih kompleks dibandingkan dengan menggunakan *Discretionary Access Control.* Pada umumnya, penggunaan dengan tipe ini dilakukan untuk memproses data yang bersifat sensitif, misalnya informasi pemerintah atau swasta, informasi badan intelejen, dan lain- lain.

Tahapan Autentikasi :

1) Autentikasi untuk mengetahui lokasi dari peralatan pada suatu simpul jaringan (data *link layer* dan *network layer*).
2) Autentikasi untuk mengenal sistem operasi yang terhubung ke jaringan (*transport layer*).
3) Autentikasi untuk mengetahui fungsi/proses yang sedang terjadi di suatu simpul jaringan (*session dan presentation layer*).

KEBIJAKAN DAN MEKANISME KEAMANAN

Pemisahan antara kebijakan dan mekanisme keamanan akan membantu memisahkan kebutuhan implementasinya :

- **Kebijakan** menspesifikasikan kebutuhan
- **Mekanisme** menerapkan spesifikasi kebijakan tersebut

Berdasar spesifikasi dari OSI, sebuah layanan (kebijakan) keamanan meliputi :

- ***Access Control***, Perlindungan terhadap pemakaian tak legak.
- ***Authentication***, Menyediakan jaminan identitas seseorang.
- ***Confidentiality***, Perlindungan terhadap pengungkapan identitas tak legak.
- ***Integrity***, Melindungi dari pengubahan data yang tak legak.
- ***Non-repudiation***, Melindungi terhadap penolakan komunikasi yang sudah pernah dilakukan.

Untuk mencapai layanan keamanan tersebut, mekanisme-mekanisme yang dapat diterapkan :

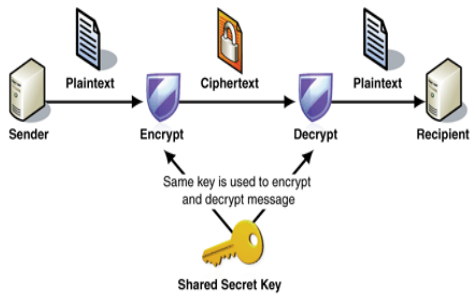
A. Enkripsi

- proses pengkodean pesan untuk menyembunyikan isi.
- Digunakan untuk menyediakan kerahasiaan, dapat menyediakan authentication dan perlindungan integritas.
- Algoritma enkripsi modern menggunakan kunci (key).
  - Pesan M (plaintext) di enkodekan dengan fungsi E dan sebuah kunci K untuk menjadi ciphertext. ***E(K,M) = {M}K***
  - Pesan didekripsi dengan menggunakan fungsi D dan kunci L. ***D(K,E(K,M)) = M***

Banyak algoritma enkripsi yang terkenal dan mereka semua memiliki fungsi yang berbeda-beda. Mereka memiliki dua karakteristik yaitu mengidentifikasi dan yang membedakan algoritma enkripsi antara satu dengan yang lain adalah kemampuan untuk melindungi data dari serangan dan kecepatan dan efisiensi dalam melakukan enkripsi.

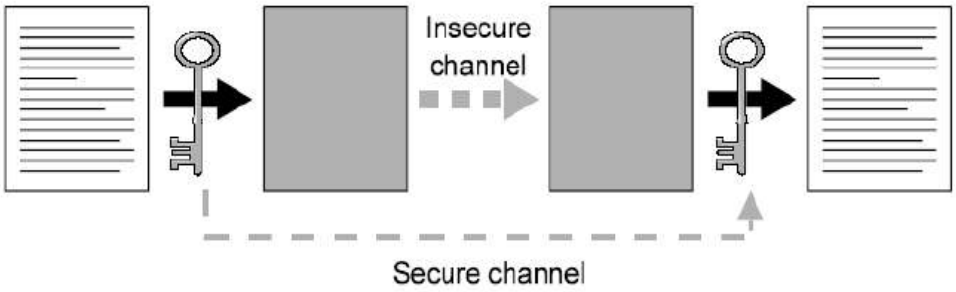
Semua algoritma enkripsi sebagian besar menggunakan dua jenis algoritma, yaitu:

- ***Algoritma Symmetric key***
  Menggunakan kunci enkripsi yang terkait atau identik untuk enkripsi dan dekripsi.

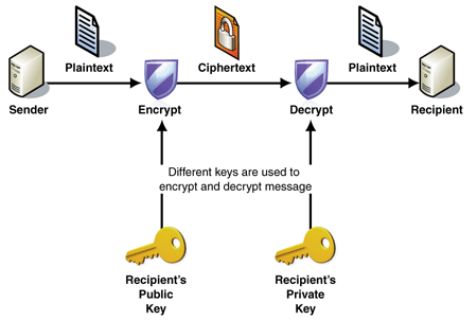


Ilustrasi : "*Alice menaruh sebuah pesan rahasia di dalam kotak dan mengunci kotak menggunakan gembok dan ia memiliki kuncinya. Kemudian dia mengirimkan kotak ke Bob melalui surat biasa. Ketika Bob menerima kotak, ia menggunakan kunci salinan sama persis yang dimiliki Alice untuk membuka kotak dan mem-*

baca pesan. Bob kemudian dapat menggunakan gembok yang sama untuk membalasa pesan rahasia". Dari contoh itu, algoritma sysmmetric-key dapat dibagikan kepada stream cipher dan block cipher. Stream cipher mengenkripsi satu per satu bit dari pesan, dan block cipher mengambil beberapa bit, biasanya 64bit dan mengenkripsi mereka menjadi satu bagian. Ada banyak algoritma berbeda dari symmetric termasuk Twofish, Serpent, AES (Rijndael), Blowfish, CAST5, RC4, TDES, and IDEA.



- ***Algoritma Asymmetric key***
  Menggunakan kunci berbeda untuk enkripsi dan dekripsi. Biasanya ini disebut sebagai *Public-key Cryptography*.



Ilustrasi ; "*Pertama Alice meminta Bob untuk mengirim gembok yang terbuka melalui surat biasa, sehingga ia tidak membagikan kuncinya. Ketika Alice menerimanya, ia menggunakannya untuk mengunci sebuah kota yang berisi pesan dan mengirimkan kotak dengan gembok terkunci tadi ke Bob. Bob kemudian membuka kotak dengan kunci yang ia pegang karena itu gembok miliknya untuk membaca pesan Alice. Untuk membalasnya, Bob harus meminta Alice untuk melakukan hal yang sama*". Keuntungan dari metode asymmetric key adalah Bob dan Alice tidak pernah berbagi kunci mereka. Hal ini untuk mencegah pihak ketiga agar tidak menyalin kunci atau memata-matai pesan Alice dan Bob. Selain itu, jika Bob ceroboh dan membiarkan orang lain untuk menyalin kuncinya, pesan Alice ke Bob akan terganggu, namun pesan Alice kepada orang lain akan tetap menjadi rahasia, karena orang lain akan memberikan gembok milik mereka ke Alice untuk digunakan.

B. Kriptografi

Kriptografi atau "*Cryptography*" adalah ilmu dan seni (*art and science*) yang dapat menyembunyikan berita terang (*plaintext*) menjadi berita sandi (*ciphertext*) dengan formula atau algoritma tertentu dan biasanya menggunakan kunci juga. Biasanya kriptografi menyediakan

service CIA (*Confidensial, Integrity, Authentication*).

Ada beberapa macam kriptografi, diantaranya adalah :

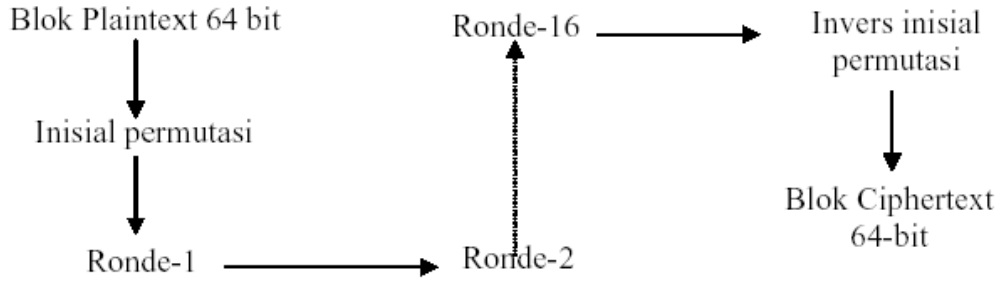
1. ***Data Encryption Standard (DES)***

   Adalah sebelumnya dominan algoritma untuk enkripsi data elektronik. Ini sangat berpengaruh dalam kemajuan modern kriptografi di dunia akademis. Dikembangkan pada awal tahun 1970 di IBM dan didasarkan pada desain sebelumnya oleh Horst Feistel, algoritma telah disampaikan kepada Badan Standar Nasional (NBS) mengikuti undangan badan untuk mengusulkan calon untuk perlindungan sensitif, unclassified data pemerintah elektronik.

   DES sekarang dianggap tidak aman untuk banyak aplikasi. Hal ini terutama disebabkan oleh 56-bit ukuran kunci yang terlalu kecil, pada bulan Januari 1999, *distributed.net* dan *Electronic Frontier Foundation* berkolaborasi untuk publik memecahkan kunci DES dalam 22 jam dan 15 menit Ada juga beberapa hasil analisis yang menunjukkan kelemahan teoritis dalam *cipher*, meskipun mereka tidak layak untuk me-*mount* dalam praktek. Algoritma diyakini praktis aman dalam bentuk Triple DES, meskipun ada serangan teoritis. Dalam beberapa tahun terakhir, cipher telah digantikan oleh Advanced Encryption Standard (AES).

   DES merupakan salah satu algoritma kriptografi cipher block dengan ukuran blok 64 bit dan ukuran kuncinya 56 bit. Algoritma DES dibuat di IBM, dan merupakan modifikasi daripada algoritma terdahulu yang bernama *Lucifer*. *Lucifer* merupakan algoritma cipher block yang beroperasi pada blok masukan 64 bit dan kuncinya berukuran 128 bit.

   Cara kerja DES secara sederhana dapat digambarkan sebagai berikut :



2. **Advance Encryption Standard (AES)**

   Merupakan pengganti dari DES setelah DES ditarik sebagai standar oleh Institut Nasional Standar dan Teknologi (sebelumnya National Bureau of Standards).

   AES adalah sebuah spesifikasi untuk enkripsi data elektronik yang ditetapkan oleh US *National Institute of Standar dan Teknologi* (NIST) pada tahun 2001 AES termasuk dalam standar ISO/IEC 18033-3. AES tersedia dalam berbagai paket enkripsi yang berbeda, dan merupakan

yang pertama diakses publik dan terbuka cipher disetujui oleh *National Security Agency* (NSA) untuk rahasia informasi bila digunakan dalam NSA disetujui modul kriptografi.

AES merupakan algoritma *cryptographic* yang dapat digunkan untuk mengamakan data. Algoritma AES adalah *blokchipertext* simetrik yang dapat mengenkripsi (*encipher*) dan dekripsi (*decipher*) infoermasi.

ALGORITMA KRIPTOGRAFI

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan kunci yang dipakainya :

1. Kriptografi Simetris

Kriptografi Simetris adalah: Kode Hill atau lebih dikenal dengan Hill cipher merupakan salah satu algoritma kriptografi kunci simetris dan merupakan salah satu kripto polyalphabetic. *Hill cipher* diciptakan oleh Lester S. Hill pada tahun 1929. Teknik kriptografi ini diciptakan dengan maksud untuk dapat menciptakan cipher yang tidak dapat dipecahkan menggunakan teknik analisis frekuensi.

Cara Enkripsi : Dengan mengkodekan atau mengubah setiap huruf abjad dengan integer sebagai berikut:

A = 0, B = 1, C = 2, D = 3, ……. , Z = 25

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Secara matematis, proses enkripsi pada hill cipher adalah:

C = K . P mod 26

C = Cipherteks | K = Kunci | P = Plainteks

Proses enkripsi pada hill cipher dilakukan per blok plainteks. Ukuran blok tersebut sama dengan ukuran matriks kuncinya. Contoh :

P = D O D I S P U T R A

dikodekan/diintegerkan menjadi :

P = 3 14 3 8 18 15 20 19 17 0

$$K = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$$

Karena matriks kunci K berukuran 2, maka plainteks dibagi menjadi blok yang masing-masing bloknya berukuran 2 karakter. Blok pertama dari plainteks P1,2 =[3;14] kemudian dienkripsi dengan kunci K dengan persamaan C = K . P mod 26. Karena perkalian tersebut menghasilkan lebih dari angka 25 maka dilakukan modulo 26 pada hasil yang lebih dari 25.

$$C_{1,2} = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} \begin{bmatrix} 3 \\ 14 \end{bmatrix} = \begin{bmatrix} 73 \\ 139 \end{bmatrix} (\text{mod } 26) \equiv \begin{bmatrix} 21 \\ 9 \end{bmatrix}$$

Karakter yang berkorespondensi dengan 21 dan 9 adalah V dan J. Setelah melakukan enkripsi semua blok pada plainteks P maka dihasilkan cipherteks C sebagai berikut:

P = D O D I S P U T R A

C = V J R N P W L U R X

Cipherteks yang dihasilkan oleh enkripsi hill chiper atau kode hill menghasilkan cipherteks yang tidak memiliki pola yang mirip dengan plainteks atau pesan aslinya.

2. Kriptografi Asimetris

Algoritma asimetris, sering juga disebut dengan algoritma kunci publik atausandi kunci publik, menggunakan dua jenis kunci, yaitu kunci publik (public key) dan kunci rahasia (secret key). Kunci publik merupakan kunci yang digunakan untuk mengenkripsi pesan. Sedangkan kunci rahasia digunakan untuk mendekripsi pesan.

Kunci publik bersifat umum, artinya kunci ini tidak dirahasiakan sehingga dapat dilihat oleh siapa saja. Sedangkan kunci rahasia adalah kunci yang dirahasiakan dan hanya orang-orang tertentu saja yang boleh mengetahuinya.

Cara Enkripsi:

A. Kunci Publik:
   1. Pilih bilangan prima p = 7 dan q = 11, n = 7.11 =77
   2. F(n)=(p-1).(q-1)=6.10= 60, artinya;
   3. F(n)={1,2,3,4,6,8,..,76}={x|gcd(x, n)=1}
   4. Pilih e dalam {x|gcd(x, 60)=1}, misalnya e=17
   5. Hapus p dan q dan Kunci Publik n=77, e=17

B. Kunci Rahasia:
   1. d = e-1 mod F(n), d .e = 1 mod 60, d =53
   2. 53 . 17 mod 60 = 901 mod 60 = 1 mod 60

3. Kriptografi Hibrid

Sistem ini mengggabungkan *chiper* simetrik dan asimetrik. Proses ini dimulai dengan negosiasi menggunakan *chiper* asimetrik dimana kedua belah pihak setuju dengan *private key/session key* yang akan dipakai. Kemudian session key digunakan dengan teknik *chiper* simetrik untuk mengenkripsi *conversation* ataupun tukar-menukar data selanjutnya. Suatu *session key* hanya dipakai sekali sesi. Untuk sesi selanjutnya *session key* harus dibuat kembali.

Contoh Kriptografi Hibrid : Metode hibrida terdiri atas enkripsi simetris dengan satu kunci (*Session Key*) dan enkripsi asimetris dengan sepasang kunci (*Public / Private Key*).

- Langkah 1 : Pengirim mengenkripsi teks dengan *Session Key*.
- Langkah 2 : Mengenkripsi *Session Key* dengan *Public Key*.
- Langkah 3 : Penerima men-*decrypt Session Key* dengan *Private Key*.
- Langkah 4 : Men-*decrypt* teks dengan *Session Key*.

KRIPTOGRAFI MENGGUNAKAN MODE *CIPHER BLOCK CHAINING* (CBC)

Cipher Block Chaining (CBC) adalah modus operasi untuk *blok cipher* (salah satu di mana urutan bit dienkripsi sebagai satu kesatuan atau blok dengan *kunci cipher* diterapkan pada seluruh blok). *Block chaining cipher* menggunakan apa yang dikenal sebagai *vektor inisialisasi* dari panjang tertentu. Salah satu karakteristik kunci adalah bahwa ia menggunakan mekanisme chaining yang menyebabkan dekripsi blok ciphertext bergantung pada semua blok ciphertext sebelumnya.

Akibatnya, seluruh validitas semua blok sebelumnya terkandung di *blok ciphertext* segera sebelumnya. Sebuah kesalahan bit tunggal dalam blok

ciphertext mempengaruhi dekripsi dari semua blok berikutnya. Penataan urutan blok ciphertext menyebabkan dekripsi menjadi rusak. Pada dasarnya, di blok chaining cipher, setiap *blok plaintext XOR* (lihat XOR) dengan blok ciphertext segera sebelumnya, dan kemudian dienkripsi.

CBC membutuhkan teks biasa empuk dengan ukuran blok cipher. Untuk informasi tambahan mengenai modus ini, lihat Blok *Cipher Mode* Operasi. *Blok ciphertext* yang identik hanya dapat terjadi jika *blok plaintext* yang sama dienkripsi menggunakan kedua tombol yang sama dan vektor inisialisasi, dan jika urutan ciphertext blok tidak berubah. Ini memiliki keuntungan atas modus Kode Buku Elektronik di bahwa XOR dalam proses menyembunyikan pola plaintext.

Idealnya, *vektor inisialisasi (IV)* harus berbeda untuk setiap dua pesan dienkripsi dengan kunci yang sama. Meskipun vektor inisialisasi tidak perlu rahasia, beberapa aplikasi mungkin menemukan ini diinginkan.

Mode operasi ini menerapkan mekanisme umpan balik (*feedback*) pada sebuah blok, yang dalam hal ini hasil enkripsi blok sebelumnya diumpan-balikkan ke dalam enkripsi blok yang current. Caranya, blok plainteks yang current di-XOR-kan terlebih dahulu dengan blok cipherteks hasil enkripsi sebelumnya, selanjutnya hasil peng-XOR-an ini masuk ke dalam fungsi enkripsi.

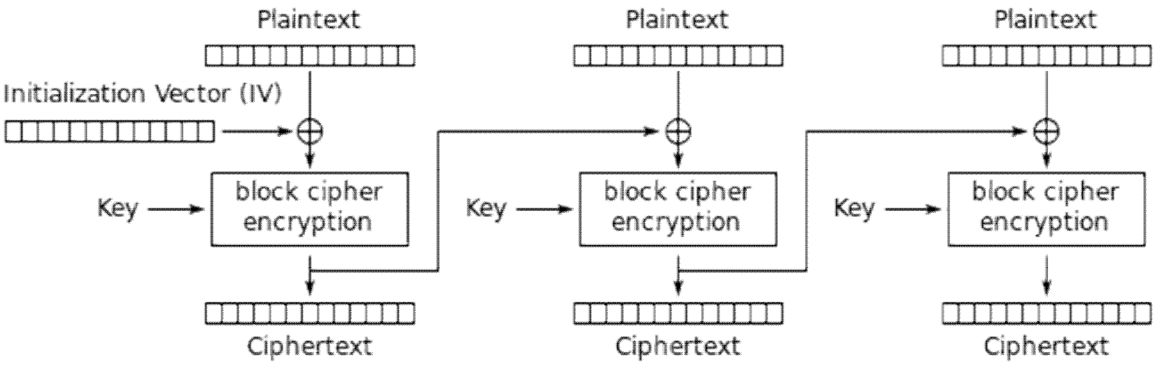
Untuk menghasilkan blok *cipher* pertama, IV (*initialization vector*) digunakan untuk menggantikan blok *cipherteks* sebelumnya. Sebaliknya pada dekripsi, blok *plainteks* pertama diperoleh dengan cara meng-XOR-kan IV dengan hasil dekripsi terhadap blok *cipherteks* pertama. Secara matematis, enkripsi dengan mode CBC dinyatakan sebagai berikut :

$$C_i = E_K(P_i \oplus C_{i-1}),$$
$$C_0 = IV$$

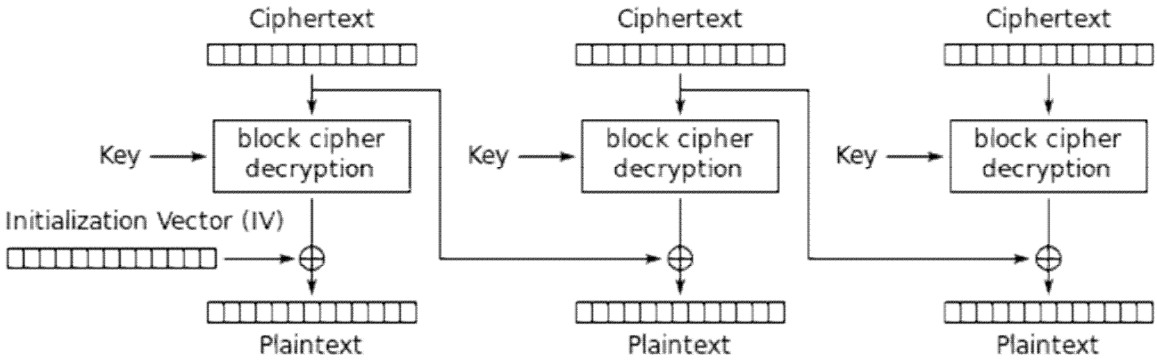
dan dekripsi sebagai

$$P_i = D_K(C_i) \oplus C_{i-1},$$
$$C_0 = IV.$$

Mode Operasi *Cipher Block Chaining* (CBC) enkripsi dan dekripsi dapat digambarkan sebagai berikut :



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

Cara kerja mode operasi *Cipher Block Chaining* :

1. Proses Enkripsi
   - Bagi plainteks menjadi blok yang telah ditentukan ukurannya, pada perangkat lunak ini tiap blok berukuran 64 bit.
   - Tiap blok yang telah dibagi kemudian di-XOR-kan dengan IV.
   - Kemudian hasil yang telah didapatkan di-XOR-kan kembali dengan kunci.
   - Hasil operasi XOR tersebut digeser 1 bit ke kiri.
   - Hasil tersebut menjadi IV untuk blok berikutnya.
   - Proses diulang sampai blok terakhir.

   Contoh Enkripsi :
   Plainteks (P) : U
   Kunci (K) : K
   IV/C0 : 00000000

   Maka
   P : 01010101
   K : 01001011
   C0 : 00000000
   U = 01010101

   C1 diperoleh sebagai berikut :
   P1 xor C0 = 01010101 xor 00000000 = 01010101

   Kemudian hasil yang diperoleh di-XOR dengan kunci 01010101 xor 01001011 = 00011110 Geser ke kiri 1 bit.

   0001 menjadi 0010 dan 1110 menjadi 1101. Maka C1 = 00101101 atau dalam hexa = 2D

2. Proses Dekripsi
   - Proses dekripsi dilakukan dari blok paling akhir.
   - Hasil pembagian blok kemudian digeser 1 bit ke kanan.
   - Kemudian hasil pergeseran tersebut di-XOR-kan dengan kunci.
   - Kemudian hasil tersebut di-XOR-kan kembali dengan blok cipherteks sebelumnya.
   - Proses diulang sampai blok paling awal, blok paling awal di-XOR-kan dengan IV.

   Contoh Dekripsi :
   C1 = 00101101
   Geser 1 bit ke kanan 0010 menjadi 0001 dan 1101 menjadi 1110

   Kemudian di-XOR-kan dengan kunci 00011110 xor 01001011 = 01010101

   Hasil yang diperoleh kemudian di-XOR kembali dengan C0
   01010101 xor 00000000 = 01010101 maka 01010101 = U

Contoh Program

Misalkan Anda ingin membuat modul *blok cipher* Anda sendiri yaitu *Crypt :: CBC-compliant*. *Block cipher* Anda harus mampu mengembalikan ukuran blok itu menggunakan *Crypt :: CBC*. Misalkan blok *cipher* Anda bernama *MyBlockCipher*, dan menggunakan ukuran blok 64 *bit* (8 *byte*) dan ukuran kunci 128 *bit* (16 byte), dan selanjutnya anggaplah bahwa Anda ingin menggunakan pemrograman XS. Edit file .xs Anda dan tambahkan baris berikut:

```
int
keysize(...)
    CODE:
        RETVAL = 16;
    OUTPUT:
        RETVAL

int
blocksize(...)
    CODE:
        RETVAL = 8;
    OUTPUT:
        RETVAL
```

File .xs Anda sekarang seharusnya berisi sebagai berikut :

```
1  #include "EXTERN.h"
2  #include "perl.h"
3  #include "XSUB.h"
4  #include "ppport.h"
5
6  /* some code here */
7
8  MODULE = Crypt::MyBlockCipher     PACKAGE = Crypt::MyBlockCipher
9
10  int
11  keysize(...)
12      CODE:
13          RETVAL = 16;
14      OUTPUT:
15          RETVAL
16
17  int
18  blocksize(...)
19      CODE:
20          RETVAL = 8;
21      OUTPUT:
22          RETVAL
23
24  /* some more code here */
```

"RETVAL" singkatan dari "*return value*" dan ini adalah data yang diperlukan oleh *Crypt :: CBC*. Dalam file .xs di atas, fungsi "*keysize()*" mengembalikan nilai 16 *byte* (kunci 128-*bit*), sedangkan "*blocksize()*" mengembalikan 8 *byte* (panjang blok 64-*bit*).

Contoh lain menunjukkan bagaimana untuk mengenkripsi pesan sederhana menggunakan *script Crypt :: CBC*.

```
1  #!/usr/local/bin/perl
2  use diagnostics;
3  use strict;
4  use warnings;
5  use Crypt::CBC;
6
7  my $key = "hello, there!";
8  my $IV = pack "H16", "0102030405060708";
9
10  my $cipher = Crypt::CBC->new({'key' => $key,
11                                'cipher' => 'Khazad',
12                                'iv' => $IV,
13                                'regenerate_key' => 1,
14                                'padding' => 'standard',
15                                'prepend_iv' => 0
16                               });
17
18  my $plaintext1 = pack "H32", "0123456789abcdeffedcba9876543210";
19  print "plaintext1  : ", unpack("H*", $plaintext1), "\n";
20
21  my $ciphertext1 = $cipher->encrypt($plaintext1);
22  print "ciphertext1 : ", unpack("H*", $ciphertext1), "\n";
23
24  my $plaintext2 = $cipher->decrypt($ciphertext1);
25  print "plaintext2  : ", unpack("H*", $plaintext2), "\n";
```

Script berikutnya menggambarkan penggunaan *Serpent*, 128-*bit block cipher* yang menggunakan kunci 128-*bit* :

```
1  #!/usr/local/bin/perl
2  use diagnostics;
3  use strict;
4  use warnings;
5  use Getopt::Long;
6  use Crypt::CBC;
7
8  my $SRC_RAND = "/dev/urandom";
9  my $IV_FILE = "iv.rand";
10  my $BLOCKSIZE = 16;
11  my $BLOCK_CIPHER = "Serpent";
12  my $IV;
13  my ($encrypt, $decrypt) = ();
14  GetOptions("encrypt" => \$encrypt, "decrypt" => \$decrypt);
15
16  sub get_input
17  {
18      my ($message) = @_;
19      local $| = 1;
20      local *TTY;
```

```
21      open TTY,"/dev/tty";
22      my ($tkey1, $tkey2);
23      system "stty -echo </dev/tty";
24      do {
25          print STDERR "Enter $message: ";
26          chomp($tkey1 = <TTY>);
27          print STDERR "\nRe-type $message: ";
28          chomp($tkey2 = <TTY>);
29          print STDERR "\n";
30          print STDERR "\nThe two $message", "s don't match. ",
31              "Please try again.\n\n" unless $tkey1 eq $tkey2;
32      } until $tkey1 eq $tkey2;
33
34      system "stty echo </dev/tty";
35      close TTY;
36      return $tkey1;
37  }
38
39  my $key = &get_input("password");
40
41  chomp $ARGV[0];
42  open INFILE, $ARGV[0];
43
44  my $cipher;
45
46  if ($encrypt) {
47      open RANDSRC, $SRC_RAND;
48      read(RANDSRC, $IV, $BLOCKSIZE);
49      close RANDSRC;
50      open SRC_IV, "> $IV_FILE";
51      print SRC_IV $IV;
52      close SRC_IV;
53
54      $cipher = Crypt::CBC->new({'key' => $key,
55                                 'cipher' => $BLOCK_CIPHER,
56                                 'iv' => $IV,
57                                 'regenerate_key' => 1,
58                                 'padding' => 'standard',
59                                 'prepend_iv' => 0
60                                });
61
62      $cipher->start('encrypt');
63  }
64
65  if ($decrypt) {
66      open RANDSRC, $IV_FILE;
67      read(RANDSRC, $IV, $BLOCKSIZE);
68      close RANDSRC;
69
70      $cipher = Crypt::CBC->new({'key' => $key,
71                                 'cipher' => $BLOCK_CIPHER,
72                                 'iv' => $IV,
73                                 'regenerate key' => 1,
```

```
74                                  'padding' => 'standard'
75                                });
76
77      $cipher->start('decrypt');
78  }
79
80  while (read(INFILE, my $buffer, 1048576)) {
81      print $cipher->crypt($buffer);
82  }
83
84  close INFILE;
85  print $cipher->finish;
```

PENUTUP

Mode operasi CBC dapat menyembunyikan pola dari *plaintext*. Mengapa demikian? Karena sebelum dienkripsi, *plaintext* di-XOR dengan IV atau *ciphertext* sebelumnya, sehingga *plaintext* yang sama belum tentu menghasilkan *ciphertext* yang sama, kecuali jika memiliki IV/*ciphertext* sebelumnya yang sama.

Kelemahan dari CBC adalah untuk mendekripsi *ciphertext,* dipengaruhi oleh *ciphertext* sebelumnya. Jika ada kesalahan pada *ciphertext* sebelumnya, maka *ciphertext* selanjutnya pun akan salah.

Proses kriptografi (enkripsi dan dekripsi) pesan/informasi menggunakan metode *Cipher Block Chaining* dapat mengacak pesan menjadi pesan yang tidak terbaca dan dapat didekripsi kembali menjadi pesan asli.

DAFTAR PUSTAKA

Setiawan Agung, Pengantar Sistem Komputer Edisi Revisi, Informatika, Bandung, 2005.

Bosworth Seymor, Kebay M. E : *Computer Security Handbook 4ed*, John Wiley & Sons 2002

Check Point Software Technologies: Principles of Network Security, Check Point Software Technologies 2003

Kaye Doug, Loosely Coupled : Missing Pieces of Web Services, RDS Press 2003

Skillsoft Press: Cryptography Protocols and Algorithms, Skillsoft press 2003

Menga Justin, Timm Carl: CCSP: *Secure Intrusion Detection and SAFE Implementation Study Guide, Sybex* 2004.

Howard Michael: *Designing Secure Web-Bases Applications for Microsoft Windows* 2000, *Microsoft Press* 2000

ITU-T: *Compendium of Approved ITU-T Security Definitions, edition* 2003 February, ITU 2003

Peuhkuri Markus: *Lecture Material: Securing the Network and Information*, 2004

Nguyen Hung Q., Johnson Bob, Hackett Michael: *Testing Applications on the Web: Test Planning for Mobile and Internet-Based System 2nd Edition*, John Wiley & Sons 2003

Russell Ryan et al., Stealing the Network: How to Own the Box, Syngress Publishing 2003

Koconis David, Murray Jim, Purvis Jos, Wassom Darrin: Securing Linux: A Survival Guide for Linux Security, SANS Institute 2003

Erickson Jon: Hacking: The Art of Exploitation, No Starch Press 2003

Mirza Ahmad David R. Et al.: Hack Proofing Your Network, 2nd Edition, Syngress Publishing 2002

Wang Wallace: Steal This Computer Book 3: What They Won't Tell You About the Internet No Starch Press 2003

Preethan V. V.:Internet Security and Firewalls Premier Press 2002

Brenton Chris, Hunt Cameron: Mastering Network Security 2nd Edition, Sybex 2003

Litlejohn Shinder, Debra : Scene of The Cybercrime - Computer Forensic handbook, Syngress Publishing 2003

Crayton Christopher A.: The Security+ Exam Guide: TestTakers Guide Series. Charles River Media 2003

Schmied Will et al.:MCSE/MCSA Implementing & Administering Security in a Windows 2000 Network Study Guide, Syngress Publising 2003

Kevin Mitnick: The Art of Deception, John Wiley & Sons 2003

Shimonski Robert J. Et al.: The Best Damn Firewall Book Period, Syngress Publishing 2003

Andres Steven, Kenyon Brian: Security Sage's Guide to Hardening the *Network Infrastructure, Syngress Publishing* 2004

CSI/FBI: *Computer Crime and Security Survey* 2004

Andress Mandy, Cox Phil, Tittel Ed (ed): *CIW Security Professional Certification Bible*, John Wiley & Sons 2001

Applied Cryptography: Protocols, Algorithms, and Source Code in C, *Second Edition*. Bruce Schneier, John Wiley & Sons, 1997, ISBN 0-471-12845-7

Crypt::CBC Module, Lincoln D. Stein, http://www.cpan.org/ authors/id/L/LD/ LDS/Crypt-CBC-1.00.readme. (diakses tanggal 15 Oktober 2015, 18.00)

http://www.drdobbs.com/web-development/encryption-using-cryptcbc/184416083 (diakses tanggal 15 Oktober 2015, 17.10)

Netcraft: Site Outages for The SCO Group, http://news.netcraft.com/archives/2004/05/27/site_outages_for_the_sco_group.html (diakses tanggal 15 Oktober 2015, 20.00)

www.google.com\[02-Keamanan_Jaringan_Komputer.pdf] (diakses tanggal 20 Oktober 2015, 20.00)

WWW.Keamanan Jaringan Komputer\ Tugas individu\renderwork - Cara Hacker Mendapatkan Password Anda.mht] (diakses tanggal 20 Oktober 2015, 20.30)

https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation (diakses tanggal 5 November 2015, 19.15)

http://www.metode-algoritma.com/2016

/01/kriptografi-cipher-block-chaining-cbc.html (diakses tanggal 25 Maret 2016, 20.45)

https://cryptobounce.wordpress.com/tag/kriptografi-simetrik/ (diakses tanggal 25 Maret 2016, 18.38)