BAB II

TINJAUAN PUSTAKA

Masalah keamanan data merupakan salah satu aspek yang sangat penting dari sebuah sistem informasi. Tetapi, masalah ini kurang mendapat perhatian dari para pemilik dan pengelola sistem informasi yang mementingkan performansi sistem, bila menggangu performansi dari sistem maka tingkat keamanan dikurangi atau bahkan ditiadakan. Dalam keamanan terdapat dua masalah yang penting, yaitu :

1. Kehilangan data, dapat disebabkan antara lain :
   a. Bencana, seperti kebakaran, banjir, gempa bumi, perang, kerusuhan.
   b. Kesalahan perangkat keras dan perangkat lunak, disebabkan oleh tidak berfungsinya pemroses, disk yang tidak terbaca, kesalahan telekomunikasi, kesalahan program (bugs).
   c. Kesalahan manusia, seperti kesalahan pemasukan data, memasang disk yang salah, eksekusi program yang salah, kehilangan disk.
2. Penyusup *(intruder*) terdiri dari :
   a. Penyusup pasif, yaitu membaca data yang tidak diotorisasi.
   b. Penyusup aktif, yaitu mengubah data yang tidak diotorisasi.

Informasi saat ini sudah menjadi sebuah komoditi yang sangat penting. Kemampuan untuk mengakses dan menyediakan informasi secara cepat dan akurat menjadi sangat esensial bagi sebuah organisasi, baik yang berupa

organisasi komersial (perusahaan), perguruan tinggi, lembaga pemerintahan, maupun individual.

Sangat pentingnya nilai sebuah informasi menyebabkan informasi yang diinginkan hanya boleh diakses oleh orang-orang tertentu. Jatuhnya informasi ke pihak lain dapat menimbulkan kerugian bagi pemilik informasi. Untuk itu keamanan dari sistem informasi yang digunakan harus terjamin dalam batas yang dapat diterima. Salah satu cara untuk mengamankan data dan informasi adalah dengan menggunakan metode kriptografi.

2.1 Pengertian Kriptografi

Kemajuan di bidang telekomunikasi dan komputer telah memungkinkan seseorang untuk melakukan transaksi bisnis secara *cashless*, selain itu ia juga dapat mengirimkan informasi kepada temannya secara *on-line*. Kegiatan-kegiatan tersebut tentu saja akan menimbulkan resiko bilamana informasi yang sensitif dan berharga tersebut diakses oleh orang-orang yang tidak berhak (*unauthorized persons*). Misalnya, informasi mengenai nomor kartu kredit anda, bila informasi ini jatuh kepada orang-orang yang jahat maka anda harus bersiap-siap terhadap melonjaknya tagihan kartu kredit anda. Sebelum tahun 1970-an, teknologi kriptografi digunakan terbatas hanya untuk tujuan militer dan diplomatik. Akan tetapi kemudian bidang bisnis dan perorangan mulai menyadari pentingnya melindungi informasi berharga.

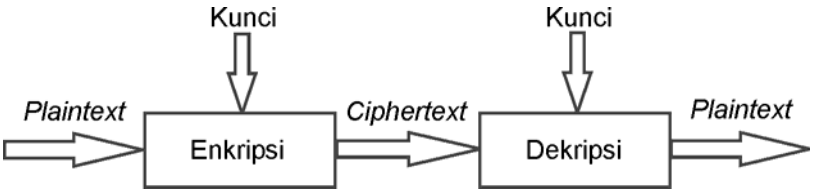
Kriptografi (*Cryptography*) berasal dari bahasa Yunani yaitu dari kata Crypto dan Graphia yang berarti penulisan rahasia. Kriptografi adalah suatu ilmu

yang mempelajari penulisan secara rahasia. Kriptografi merupakan bagian dari suatu cabang ilmu matematika yang disebut *Cryptology*. Kriptografi bertujuan menjaga kerahasiaan informasi yang terkandung dalam data sehingga informasi tersebut tidak dapat diketahui oleh pihak yang tidak sah.

Ketika suatu pesan ditransfer dari suatu tempat ke tempat lain, isi dari pesan tersebut kemungkinan dapat disadap oleh pihak lain. Untuk menjaga keamanan pesan, maka pesan tersebut dapat di-*scramble* atau diacak menjadi kode yang tidak dapat dimengerti oleh orang lain.

Dalam dunia kriptografi, pesan yang akan dirahasiakan disebut *plaintext*. Pesan yang sudah diacak disebut *ciphertext*. Proses untuk mengkonversi *plaintext* menjadi *ciphertext* disebut enkripsi. Proses untuk mengembalikan *plaintext* dari *ciphertext* disebut dekripsi. Algoritma kriptografi (*ciphers*) adalah fungsi-fungsi matematika yang digunakan untuk melakukan enkripsi dan dekripsi. Diperlukan kunci yaitu kode untuk melakukan enkripsi dan dekripsi. Secara umum, proses enkripsi dan dekripsi dapat dilihat seperti pada gambar 2.1.

Ada dua macam tipe kunci enkripsi yaitu: kunci simetrik dan kunci asimetrik. Algoritma simetrik menggunakan kunci yang sama untuk melakukan enkripsi dan dekripsi. Algoritma asimetrik menggunakan dua kunci, kunci publik untuk enkripsi dan kunci pribadi untuk melakukan deskripsi.



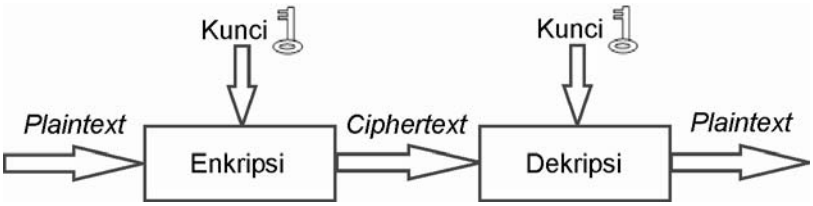
Gambar 2.1 Proses enkripsi dan dekripsi

2.2 Kelompok Enkripsi

Kelompok enkripsi terdiri dari 3 kelompok yaitu :

1. Enkripsi Simetrik (*Symmetric encryption*)

   Menggunakan kunci (*key*) yang sama untuk proses enkripsi dan dekripsi, masalah yang muncul adalah kesulitan dalam manajemen *key*. Misalnya dilakukan pengirim data yang telah dienkripsi ke seorang penerima melalui e-mail, orang yang diberi data ini hanya dapat merubah data ke dalam bentuk aslinya dengan menggunakan *key* yang sama. Jadi pengirim harus memberitahukan ke penerima data *key* yang digunakan, disini masalahnya bagaimana mengirim *key* ke penerima tersebut. Bentuk blok diagram metode simetrik diperlihatkan pada gambar 2.2.



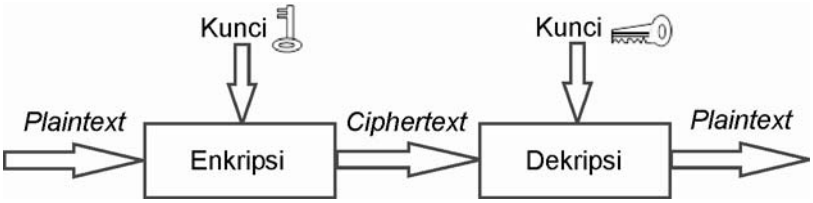
Gambar 2.2 Enkripsi simetrik

Algoritma enkripsi yang menggunakan metode ini adalah Data Encryption Standard (DES), AES, IDEA, RC-2, RC-4, RC-5.

2. Enkripsi Asimetrik (*Asymmetric encryption*)

   Menggunakan kunci (*key*) yang berbeda untuk proses enkripsi dan dekripsi. Pada sistem asimetrik dikenal adanya *private key* dan *public key*. Jika melakukan pengiriman data yang dienkripsi ke penerima, pengirim melakukan enkripsi dengan menggunakan *public key* dari penerima dan data hanya dapat dirubah kembali dalam bentuk *plaintext* dengan menggunakan *private key*

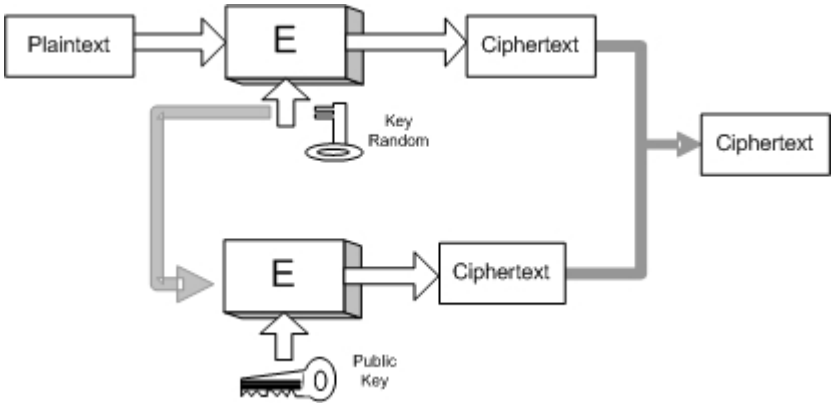
yang dimiliki penerima. Bentuk blok diagram metode asimetrik diperlihatkan pada gambar 2.3. Algoritma enkripsi yang menggunakan metode ini adalah RSA dan Dife-Hillman.



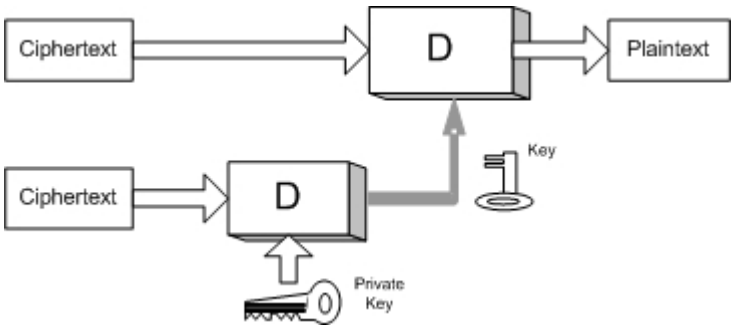
Gambar 2.3 Enkripsi asimetrik

3. Enkripsi Hibrid (*Hybrid encryption*)

Metode ini menggabungkan konsep simetrik dan asimetrik, dengan menggunakan *key random* data dienkripsi dengan metode simetrik. Tahap berikutnya *key random* yang dipakai untuk proses enkripsi data juga dienkripsi, tetapi menggunakan metode asimetrik dengan menggunakan *public key* dari penerima. *Ciphertext* yang dihasilkan masing-masing metode digabungkan dan dikirimkan ke penerima. Penerima akan membuka *Ciphertext* untuk *key word* dengan menggunakan *private key* yang dimilikinya. Bentuk blok diagram metode ini diperlihatkan pada gambar 2.4.



(a)



(b)

Gambar 2.4 (a) Proses enkripsi *hybrid* (b) Proses dekripsi *hybrid*

2.3 Mekanisme Kriptografi

Keamanan telah menjadi aspek yang sangat penting dari suatu sistem informasi. Sebuah informasi umumnya hanya ditujukan bagi pihak tertentu. Oleh karena itu sangat penting untuk mencegah informasi jatuh kepada pihak-pihak lain yang tidak berkepentingan. Untuk melaksanakan tujuan tersebutlah dirancang suatu sistem keamanan yang berfungsi melindungi sistem informasi. Salah satu upaya pengamanan sistem informasi yang dapat dilakukan adalah kriptografi. Kriptografi sesungguhnya merupakan studi terhadap teknik matematis yang terkait dengan aspek keamanan suatu sistem informasi, antara lain seperti kerahasiaan, integritas data, otentikasi, dan ketiadaan penyangkalan.

Keempat aspek tersebut merupakan tujuan fundamental dari suatu sistem kriptografi:

1. Kerahasiaan (*confidentiality*)

   Kerahasiaan adalah layanan yang digunakan untuk menjaga informasi dari setiap pihak yang tidak berwenang untuk mengaksesnya.

Dengan demikian informasi hanya akan dapat diakses oleh pihak-pihak yang berhak saja.

2. Integritas data (*data integrity)*

   Integritas data merupakan layanan yang bertujuan untuk mencegah terjadinya pengubahan informasi oleh pihak-pihak yang tidak berwenang. Untuk meyakinkan integritas data ini harus dipastikan agar sistem informasi mampu mendeteksi terjadinya manipulasi data. Manipulasi data yang dimaksud di sini meliputi penyisipan, penghapusan, maupun penggantian data.

3. Otentikasi (*authentication*)

   Otentikasi merupakan layanan yang terkait dengan identifikasi terhadap pihak-pihak yang ingin mengakses sistem informasi (*entity authentication*) maupun keaslian data dari sistem informasi itu sendiri (*data origin authentication*).

4. Ketiadaan penyangkalan (*non-repudiation*)

   Ketiadaan penyangkalan adalah layanan yang berfungsi untuk mencegah terjadinya penyangkalan terhadap suatu aksi yang dilakukan oleh pelaku sistem informasi.

Suatu sistem kriptografi (kriptosistem) bekerja dengan cara menyandikan suatu pesan menjadi suatu kode rahasia yang dimengerti oleh pelaku sistem informasi saja. Pada dasarnya mekanisme kerja semacam ini telah dikenal sejak jaman dahulu. Dalam era teknologi informasi sekarang ini, mekanisme yang sama masih digunakan tetapi tentunya implementasi sistemnya berbeda.

Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu himpunan berisi elemen plaintext dan himpunan berisi

elemen ciphertext. Enkripsi dan dekripsi merupakan fungsi tranformasi antara dua himpunan tersebut. Bila himpunan plaintext dinotasikan dengan P dan himpunan ciphertext dinotasikan dengan C, lalu fungsi enkripsi dinotasikan dengan E dan fungsi dekripsi dengan D maka proses enkripsi-dekripsi dapat dinyatakan dalam notasi matematis sebagai berikut:

$E(P) = C$ dan

$D(C) = P$

Karena proses enkripsi-dekripsi bertujuan memperoleh kembali data asal, maka:

$D(E(P)) = P$

Relasi antara himpunan plaintext dengan himpunan ciphertext harus merupakan fungsi korespondensi satu-satu (*one to one relation*). Hal ini merupakan keharusan untuk mencegah terjadinya ambigu dalam dekripsi yaitu satu elemen ciphertext menyatakan lebih dari satu elemen plaintext.

Pada metode kriptografi simetris atau konvensional digunakan satu buah kunci. Bila kunci dinotasikan dengan ‘K’ maka proses enkripsi-dekripsi metode kriptografi simetris dapat dinotasikan dengan:

$E_K(P) = C$ dan

$D_K(C) = P$

Dan keseluruhan sistem dinyatakan sebagai :

$D_K(E_K(P)) = P$

Pada metode kriptografi asimetris digunakan kunci umum untuk enkripsi dan kunci pribadi untuk dekripsi. Bila kunci umum dinotasikan dengan ‘PK’ dan

kunci pribadi dinotasikan dengan ‘SK’ maka proses enkripsi-dekripsi metode kriptografi asimetris dapat dinotasikan dengan:

E_{PK} (P) = C dan

D_{SK} (C) =P

Dan keseluruhan sistem dinyatakan sebagai

$D_{SK}(E_{PK}(P)) = P$

Kriptografi sederhana seperti ini menggunakan algoritma penyandian yang disebut *cipher*. Keamanannya bergantung pada kerahasiaan algoritma penyandian tersebut, karena itu algoritmanya harus dirahasiakan. Pada kelompok dengan jumlah besar dan anggota yang senantiasa berubah, penggunaannya akan menimbulkan masalah. Setiap ada anggota yang meninggalkan kelompok, algoritma harus diganti karena anggota ini dapat saja membocorkan algoritma.

Kriptografi modern selain memanfaatkan algoritma juga menggunakan kunci (*key*) untuk memecahkan masalah tersebut. Proses enkripsi dan dekripsi dilakukan dengan menggunakan kunci ini. Setiap anggota memiliki kuncinya masing-masing yang digunakan untuk proses enkripsi dan dekripsi yang akan dilakukannya.

2.4 Keamanan Sistem Kriptografi

Keamanan suatu sistem kriptografi merupakan masalah yang paling fundamental. Dengan menggunakan sistem standar terbuka, maka keamanan suatu sistem kriptografi akan lebih mudah dan lebih cepat dianalisa. Mengingat kenyataan inilah maka sekarang tidak digunakan lagi algoritma rahasia yang tidak

diketahui tingkat keamanannya. Terdapat tiga faktor metode algoritma enkripsi yang perlu dipertimbangkan yaitu:

1. Tingkat kesulitan untuk meng-*crack*/menembus pesan yang dienkripsi menggunakan suatu metode enkripsi.
2. Performa dari metode enkripsi misalnya kecepatan dalam melakukan enkripsi dan dekripsi.
3. Keamanan dari kunci.

Sebuah sistem kriptografi dirancang untuk menjaga *plaintext* dari kemungkinan dibaca oleh pihak-pihak yang tidak berwenang, yang secara umum dinamakan sebagai penyerang (*attacker*). Penyerang diasumsikan memiliki akses tak terbatas terhadap jalur tak aman yang digunakan untuk transaksi *ciphertext*. Oleh karena itu, penyerang dianggap memiliki akses langsung terhadap *ciphertext*.

Tipe penyerang paling umum terhadap suatu sistem kriptografi adalah serangan kriptanalisis (*cryptanalysis attack).* Kriptanalisis merupakan ilmu yang mempelajari tentang upaya-upaya untuk memperoleh *plaintext* dari *ciphertext* tanpa informasi tentang kunci yang digunakan.

2.4.1 Macam-macam Basic *Cryptanalytic Attacks*

Tujuan *cryptanalytic attack* adalah untuk mengetahui beberapa *plaintext* yang sesuai dengan *ciphertext* yang ada dan berusaha menentukan kunci yang memetakan satu dengan yang lainnya. *Plaintext* ini dapat diketahui karena ia merupakan standar atau karena pendugaan. Jika suatu teks diduga berada di dalam suatu pesan, posisinya mungkin tidak diketahui, tetapi suatu pesan lazimnya cukup pendek sehingga memungkinkan *cryptanalyst* menduga *plaintext*

yang diketahui dalam setiap posisi yang mungkin dan melakukan penyerangan pada setiap kasus secara paralel.

Suatu algoritma enkripsi yang kuat tidak hanya mampu bertahan terhadap serangan *plaintext* yang dikenal tetapi juga mampu bertahan terhadap *adaptive chosen plaintext*. Dalam penyerangan ini, *cryptanalyst* berkesempatan memilih *plaintext* yang digunakan dan dapat melakukannya secara berulang kali.

Cryptanalytic attacks adalah usaha-usaha yang dilakukan seseorang untuk memperoleh informasi ataupun data yang telah dienkripsi. Secara ringkas terdapat tujuh macam *basic cryptanalytic attacks* berdasarkan tingkat kesulitannya bagi penyerang, dimulai dari yang paling sulit adalah:

1. *Ciphertext-only attack*. Dalam penyerangan ini, seorang *cryptanalyst* memiliki *ciphertext* dari sejumlah pesan yang seluruhnya telah dienkripsi menggunakan algoritma yang sama.
2. *Known-plaintext attack*. Dalam tipe penyerangan ini, *cryptanalyst* memiliki akses tidak hanya ke *ciphertext* sejumlah pesan, namun ia juga memiliki *plaintext* pesan-pesan tersebut.
3. *Choosen-plaintext attack*. Pada penyerangan ini, *cryptanalyst* tidak hanya memiliki akses atas *ciphertext* dan *plaintext* untuk beberapa pesan, tetapi ia juga dapat memilih *plaintext* yang di enkripsi.
4. *Adaptive-choosen-plaintext attack*. Penyerangan tipe ini merupakan suatu kasus khusus *choosen-plaintext attack*. *Cryptanalyst* tidak hanya dapat memilih *plaintext* yang dienkripsi, ia pun memiliki kemampuan untuk memodifikasi pilihan berdasarkan hasil enkripsi sebelumnya. Dalam *chosen-*

plaintext attack, *cryptanalyst* mungkin hanya dapat memiliki *plaintext* dalam suatu blok besar untuk dienkripsi, dalam *adaptive-choosen-plaintext attack* ini ia dapat memilih blok *plaintext* yang lebih kecil dan kemudian memilih yang lain berdasarkan hasil yang pertama, proses ini dapat dilakukannya terus menerus hingga ia dapat memperoleh seluruh informasi.

5. *Choosen-ciphertext attack*. Pada tipe ini, *cryptanalyst* dapat memilih *ciphertext* yang berbeda untuk didekripsi dan memiliki akses atas *plaintext* yang didekripsi.
6. *Choosen-key attack*. *Cryptanalyst* pada tipe penyerangan ini memiliki pengetahuan tentang hubungan antara kunci-kunci yang berbeda.
7. *Rubber-hose cryptanalysis*. Pada tipe penyerangan ini, *cryptanalyst* mengancam, memeras, atau bahkan memaksa seseorang hingga mereka memberikan kuncinya.

2.5 Sekilas Tentang Java

Java mula-mula dibangun oleh Sun pada tahun 1991, Java adalah suatu bahasa pemrograman yang digunakan untuk membangun suatu aplikasi seperti program utiliti, *game*, *plug-ins*, dan lain-lain untuk perangkat keras dan lunak pada platform yang berbeda. Para pengguna perangkat dengan dukungan Java dapat menginstal berbagai aplikasi baru untuk membuat perangkat yang lebih pribadi dan sesuai dengan kebutuhan.

Sun mempromosikan Java dengan konsep "*write once, run anywhere*", dimana program mampu dijalankan di sembarang platform dan sistem operasi.

Sun menerapkan konsep Java menjadi dua bagian, yaitu aplikasi Java dan Java Interpreter, yang biasa disebut sebagai *Java Virtual Machine* (JVM). Suatu aplikasi Java tidak bisa berdiri sendiri karena memerlukan suatu interpreter yang dapat menterjemahkan kode dan mengeksekusi program. Hal ini merupakan rahasia di balik kemampuan Java untuk bekerja pada berbagai platform. Suatu pengembang bisa menulis suatu aplikasi tanpa harus memikirkan komputer dan sistem operasi yang berbeda selama perangkat keras dan lunak yang digunakan telah terinstal JVM di dalamnya. Meskipun menggunakan konsep "*write once, run anywhere*", perkembangan teknologi telekomunikasi dan perangkat bergerak (*mobile devices*) telah memaksa Sun untuk mengadaptasikan Java pada perangkat bergerak yang memiliki tampilan layar yang kecil dan koneksi yang lambat. Hal ini menjadi pemicu dalam membangun *PersonalJava*, J2ME™ (*Java 2 Platform, Micro Edition*), dan *K Virtual Machine* (KVM).

2.5.1 Personal Java

Personal Java merupakan pemrograman Java pertama yang ditargetkan untuk aplikasi pada perangkat yang memiliki sumber daya terbatas, seperti PDA, Communicator/Smartphone. Personal Java memperkenalkan fitur yang dapat mengurangi penggunaan memori dan menyesuaikan aplikasi ke layar dan antarmuka grafis yang berbeda. Personal Java ditransfer ke dalam platform J2ME pada tahun 1999. Versi barunya disebut CDC/*Personal Profile*.

2.5.2 CLDC/MIDP

Sun menyusun Java menjadi tiga platform atau edisi pada tahun 1999. J2ME™ (*Java 2 Platform, Micro Edition*) merupakan platform yang ditargetkan

untuk perangkat kecil dengan kapasitas memori dan prosesor yang rendah, seperti ponsel, communicator, dan PDA. Dua platform Java yang lain adalah *Java 2 Platform, Standard Edition* (J2SE™), dan *Java 2 Platform, Enterprise Edition* (J2EE™).

J2ME ditujukan untuk berbagai perangkat kecil. Untuk menangani perangkat yang berbeda, diperkenalkan dua konsep yaitu konfigurasi dan profil. Konfigurasi mendefinisikan platform minimal untuk perangkat dengan kapasitas memori dan prosesor yang serupa. Profil ditargetkan untuk kategori perangkat yang spesifik di antara perangkat dengan konsep konfigurasi sebagai contoh ponsel.

Konsep konfigurasi dalam J2ME dibagi menjadi dua, yaitu:

1. CDC, *Connected Device Configuration*. Konfigurasi ini diarahkan untuk PDA.
2. CLDC, *Connected Limited Device Configuration*. Konfigurasi ini diarahkan untuk *mobile phones* seperti ponsel, pager, dan sebagainya.

Sejauh ini, dua profil telah dibentuk untuk J2ME yaitu MIDP, *Mobile Information Device Profile*, dan *Personal Profile*.

MIDP dihubungkan kepada CLDC dan menyediakan kepada pengembang suatu informasi penting dan bimbingan ketika menulis program untuk ponsel. Untuk menjalankan aplikasi dibutuhkan KVM yang merupakan Java Interpreter untuk J2ME CLDC/MIDP yang dapat berjalan pada prosesor dengan clock frekuensi rendah dan hanya membutuhkan memori sebesar 40-80 kilobyte.

Personal Profile dihubungkan ke CDC, yang ditargetkan untuk PDA, kombinasi ini menggantikan Personal Java dalam J2ME.

Lingkungan J2ME CLDC/MIDP secara terus-menerus diperbaharui. Spesifikasi baru ini dikerjakan/dilakukan dalam *Java Community Process* yang diatur oleh SUN. Spesifikasi baru diciptakan berdasarkan *Java Specification Request* (JSR). Selama tahun 2003 suatu inisiatif dimulai untuk memastikan pembaruan yang terjadi dalam J2ME. Inisiatif ini disebut "*Joint Technology in the Wireless Industry* (JTWI)" yang direferensikan sebagai JSR-185. Inisiatif ini dimaksudkan untuk menggambarkan suatu *roadmap* dari ponsel MIDP/CLDC dan untuk mengurangi kerancuan di dalam spesifikasi yang diterapkan. Rilis pertama dari roadmap tersebut didasarkan kepada MIDP 2.0 yang dirilis pada akhir tahun 2002.

Sampai saat ini banyak JSR yang telah disetujui dan dikembangkan pada ponsel, di antaranya yaitu JSR-120 (*Wireless Messaging 1.1*), JSR-135 (*Mobile Media*), JSR-118 (MIDP 2.0), dan JSR-82 (*Bluetooth*). Konfigurasi CLDC/MIDP terbaru yaitu CLDC 1.1 dan MIDP 2.0 yang diterapkan pada ponsel saat ini telah mendukung hampir semua JSR yang ada.

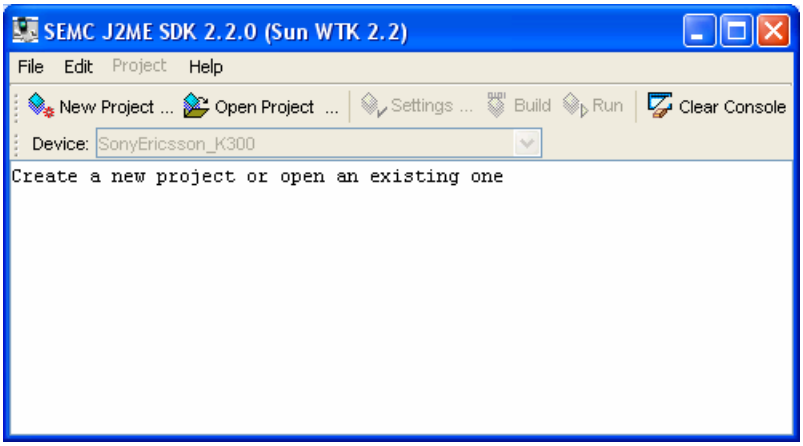
2.6 Sekilas Tentang Sony Ericsson Java MIDP SDK

Sony Ericsson Java MIDP SDK merupakan suatu emulator program aplikasi untuk ponsel Sony Ericsson. Dengan Sony Ericsson Java MIDP SDK, kita dapat menjalankan dan menguji program yang telah dibuat pada emulator ponsel Sony Ericsson yang tersedia sebelum diintegrasikan langsung pada ponsel.

Sony Ericsson Java MIDP SDK dapat berjalan dengan memenuhi spesifikasi minimum seperti berikut:

1. Microsoft Windows 2000/XP.
2. Java™ 2 SDK, Standard Edition (JDK) 1.4.1 atau yang lebih tinggi (rekomendasi JDK 1.4.2).
3. DirectX 8.1 atau yang terbaru.
4. SPOT xde (R) Player.
5. Windows Script 5.6.
6. 110 MB hard disk.
7. 256 MB sistem RAM
8. Prosesor dengan clock frekuensi 500 MHz.

Pada dasarnya untuk membuat aplikasi pada ponsel dengan J2ME menggunakan profil MIDP 2.0 diperlukan tiga buah tool. Tool-tool tersebut adalah J2SE SDK versi 1.4.1 atau yang lebih tinggi, J2ME Wireless Toolkit 2.2 atau Sony Ericsson Java MIDP SDK Release 2.2, dan sebuah Text Editor seperti UltraEdit, Borland JBuilder, Kawa, EditPlus, jEdit, dan lain-lain.



Gambar 2.5 Sony Ericsson Java MIDP SDK Release 2.2