BAB II

TINJAUAN PUSTAKA

II.1 Pengenalan Kriptografi

II.1.1 Sejarah Kriptografi

Kriptografi adalah kata serapan dari bahasa asing, dalam hal ini bahasa inggris, yaitu *cryptography*. *Cryptography* atau *cryptology* berasal dari bahasa Yunani, yaitu κρυπτός, *kryptos*, "hidden, secret" atau “tersembunyi, rahasia”, dan γράφω, *graphō*, "I write" atau “aku menulis”, dan -λογία, *-logia* atau “ilmu”.

Kriptografi adalah ilmu atau seni untuk menyembunyikan informasi. Proses menyembunyikan informasi ini dilakukan dengan teknik penyandian, atau mengubah pesan atau informasi menjadi sandi-sandi yang tidak dimengerti oleh orang lain, selain pembuat dan penerimanya.

Pada kriptografi dikenal istilah-istilah seperti *plain text*, *cipher text,* enkripsi, dan dekripsi. *Plain text* adalah pesan asli yang ingin dikirimkan. *Cipher text* adalah pesan yang telah disandikan dengan metode enkripsi tertentu. Enkripsi adalah proses mengubah sebuah *plain text* menjadi *cipher text*, dan dekripsi adalah proses mengubah sebuah *cipher text* menjadi *plain text*. Transaksi data banyak dilakukan dalam kehidupan sehari-hari. Beberapa informasi yang dikirimkan tersebut adalah informasi-informasi yang bersifat rahasia dan pribadi. Karena itu datadata yang dikirimkan perlu dirahasiakan sehingga pihak lain yang mencoba mendapatkan informasi tersebut tanpa izin tidak akan dapat mengetahuinya (Suhendra, 2011)

Pada masa lalu, informasi yang dianggap rahasia adalah seperti pada masa peperangan. Informasi mengenai operasi militer, kekuatan militer, dan sebagainya adalah informasi yang tidak boleh sampai ketahuan oleh pihak musuh. Karena itu informasi-informasi ini perlu

disandikan ketika ingin dikirimkan ke tempat lain, sehingga jika informasi yang dikirim jatuh ke tangan pihak musuh, mereka tetap tidak mengerti. Sementara pada era modern ini, kita sering melakukan transaksi secara elektronik. Transaksitransaksi ini juga perlu dilindungi karena mengandung informasi yang bersifat pribadi, misalnya nomor kartu kredit.

Karena itu kriptografi menjadi sebuah cabang ilmu yang terus diteliti dan dikembangkan untuk melindungi kerahasiaan data. Sebelum era modern ini, manusia telah menggunakan teknik penyandian dalam pertukaran atau pengiriman informasi yang dianggap sangat rahasia. Pesan-pesan yang dikirim diubah menjadi sandi-sandi sehingga orang yang menyadap tidak dapat mengetahui isi pesan tanpa pengetahuan mengenai cara penyandian. *Scytale* yang dimiliki masyarakat Yunani kuno mungkin merupakan salah satu alat pertama yang digunakan untuk menyandikan sebuah pesan. Cara kerja s*cytale* ini adalah pesan yang ingin dikirim dituliskan di secarik kulit yang dililitkan di sebatang kayu berbentuk silinder sehingga jika dilepaskan, maka tulisan tersebut menjadi teracak. Orang yang dapat menyandikan kembali pesan itu ke pesan yang sebenarnya perlu memiliki kayu yang berukuran sama (atau kayu yang sama). *Scytale* ini termasuk dalam *transposition cipher* yaitu metode enkripsi yang melakukan perubahan posisi terhadap karakter-karakter di pesan asli. Misalnya pesan “saya disini” dapat disandikan menjadi “ini sidayas” dengan menggunakan *transposition cipher* ini. Dikenal juga *substitution cipher*, atau metode enkripsi yang melakukan substitusi terhadap karakter-karakter pada pesan asli. Salah satu *substitution cipher* yang terkenal adalah *Caesar Cipher* yang mengubah setiap huruf dari pesan asli dengan huruf ke sekian di depannya. Metode ini diberi nama menurut nama Julius Caesar yang dipercaya menggunakan metode ini untuk berkomunikasi dengan jendral-jendralnya selama peperangan.

Pada perang dunia II, banyak negara yang juga mengembangkan dan menggunakan mesin mesin tertentu untuk menyandikan pesan mereka. Contohnya seperti Mesin Lorenz dan Enigma yang digunakan oleh militer Jerman. Perkembangan teknologi dan ditemukannya komputer membuat penggunaan algoritma kriptografi yang lebih kompleks lagi menjadi terbuka kemungkinannya. Banyak penelitian mengenai algoritma kriptografi telah dilakukan sehingga muncul metode baru dalam sistem penyandian pesan. Metode-metode enkripsi yang selama ini dikenal adalah dengan menggunakan sebuah kunci atau kode khusus yang digunakan untuk menyandikan dan membaca pesan.

Misalnya dalam *Caesar Cipher* digunakan 3 huruf setelah huruf yang digunakan untuk menyandikan pesan. Dan hal itu berarti untuk membaca pesan, kita cukup menggeser setiap huruf yang muncul dengan 3 huruf sebelumnya. Metode ini dikenal dengan *symmetric-key cryptography*, dimana untuk membaca dan menyandikan sebuah pesan digunakan sebuah *key* atau kunci yang sama.

Di tahun 1976, sebuah sistem kriptografi yang *asymmetric-key* dipublikasikan oleh Whitfield Diffie dan Martin Hellman yang dipengaruhi oleh metode distribusi kunci publik (*public-key*) yang dikembangkan oleh Ralph Merkle. Sistem pertukaran kunci ini dikenal dengan nama *Diffie- Hellman key exchage* (pada tahun 2002, Hellman menyarankan agar nama ini diubah menjadi *Diffie-Hellman-Merkle key exchage* atas jasa Merkle dalam penemuan metode *public-key*). Sistem ini juga disebut *public-key cryptography.* Dalam *public-key cryptography*, kedua pihak yang ingin mengirimkan dan menerima pesan akan saling bertukar kunci yang digunakan untuk mengenkripsi data, sedangkan kunci untuk melakukan dekripsi tetap dirahasiakan. Salah satu *public-key cryptography* adalah kriptografi RSA. (Suhendra 2011).

II.1.2 Pengertian Kriptografi

Suatu pesan yang tidak disandikan disebut sebagai *plaintext* ataupun dapat disebut juga sebagai *cleartext*. Proses yang dilakukan untuk mengubah *plaintext* ke dalam *ciphertext* disebut *encryption* atau *encipherment*. Sedangkan proses untuk mengubah *ciphertext* kembali ke *plaintext* disebut *decryption* atau *decipherment*.

Cryptographic adalah studi teknik matematik yang berkaitan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, autentikasi entitas, dan autentikasi asal data (Menezes *et al*. 1996). Sedang, *cryptanalysis* adalah suatu ilmu dan seni membuka (*breaking*) *ciphertext*. Orang yang melakukannya disebut kriptanalis (Stallings 2003).

Kriptografi dapat memenuhi kebutuhan umum suatu transaksi yaitu:

1. Kerahasiaan (*confidentiality*) dijamin dengan melakukan enkripsi (penyandian).
2. Keutuhan (*integrity*) atas data-data pembayaran dilakukan dengan fungsi *hash* satu arah.
3. Jaminan atas identitas dan keabsahan (*authenticity*) pihak-pihak yang melakukan transaksi dilakukan dengan menggunakan *password* atau sertifikat dijital. Sedangkan keautentikan data transaksi dapat dilakukan dengan tanda tangan dijital.
4. Transaksi dapat dijadikan barang bukti yang tidak bisa disangkal (*nonrepudiation*) dengan memanfaatkan tanda tangan digital dan sertifikat dijital.

Cryptographic system atau *cryptosystem* adalah suatu fasilitas untuk mengkonversikan *plaintext* ke *ciphertext* dan sebaliknya. Proses enkripsi dan dekripsi diatur oleh satu atau beberapa kunci kriptografi. Secara umum, kunci-kunci yang digunakan untuk proses pengenkripsian dan pendekripsisan tidak perlu identik, tergantung pada sistem yang digunakan.

Secara umum operasi *enkripsi* dan *dekripsi* dapat diterangkan secara matematis sebagai berikut :

EK(M)=C (Proses *Enkripsi*)

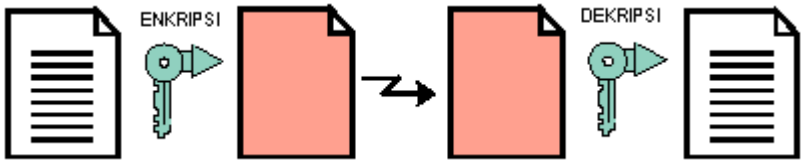
DK(C)=M (Proses *Dekripsi*)

Pada saat proses enkripsi kita menyandikan pesan M dengan suatu kunci K lalu dihasilkan pesan C. Sedangkan pada proses dekripsi, pesan C tersebut diuraikan dengan menggunakan kunci K sehingga dihasilkan pesan M yang sama seperti pesan sebelumnya.

Namun pada pengertian modern kriptografi adalah ilmu yang bersandarkan pada teknik matematika unntuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas. Jadi pengertian kriptografi modern adalah tidak saja berurusan hanya dengan penyembunyian pesan namun pada sekumpulan teknik yang menyediakan keamanan informasi (Rifki Sadikin, 2012).

Berikut ini adalah rangkuman mekanisme yang berkembang pada kriptografi modern :

1. Fungsi Hash : fungsi yang melakukan pemetaan pesan dengan panjang sembarang ke sebuah teks khusus yang disebut *message digest* dengan panjang tetap.
2. Penyandian dengan kunci simetrik (*Symmetric key emcipherment*): penyandian yang kunci enkripsi dan dekripsinya bernilai sama. Kunci pada penyandian simetrik diasumsikan bersifat rahasia hanya pihak yang melakukan enkripsi dan dekripsi yang mengetahui nilainya.
3. Penyandian dengan kunci asimetrik (*Asymmetric key encipherment*) : penyandian dengan kunci enkripsi dan dekripsi bernlai berbeda. Kunci enkripsi yang juga disebut dengan kunci publik (*public key*) bersifat terbuka. Sedangkan , kunci dekripsi yang juga disebut kunci privat (*private key*) bersifat tertutup/rahasia. (Rifki Sadikin, 2012)



Gambar II.1 Konsep Dasar dari Enkripsi dan Dekripsi
(Sumber : Dony Ariyus, 2006)

II.1.3 Enkripsi Kunci Publik

Pada enkripsi kunci publik, satu kunci digunakan untuk enkripsi dan satu kunci digunakan untuk dekripsi. Kunci enkripsi disediakan untuk umum, sedangkan kunci dekripsi harus dijaga kerahasiannya. Pengirim dan penerima pesan harus mempunyai satu dari pasangan kunci yang cocok. Algoritma ini mempunyai karakteristik yang penting yaitu secara perhitungan tidak layak menentukan kunci dekripsi dari kunci enkripsi. (Suhendra, 2011).

Berikut ini adalah hal-hal yang dibutuhkan supaya keamanan enkripsi kunci publik tetap terjaga.

1. Salah satu kunci harus dijaga kerahasiannya.
2. Pesan tidak dapat di-*dechipher* jika tidak ada informasi ekstra.
3. Kunci yang lain tidak bisa ditentukan walaupun dilengkapi dengan pengetahuan mengenai algoritma, salah satu kunci, dan contoh-contoh dari *ciphertext.* (Suhendra, 2011)

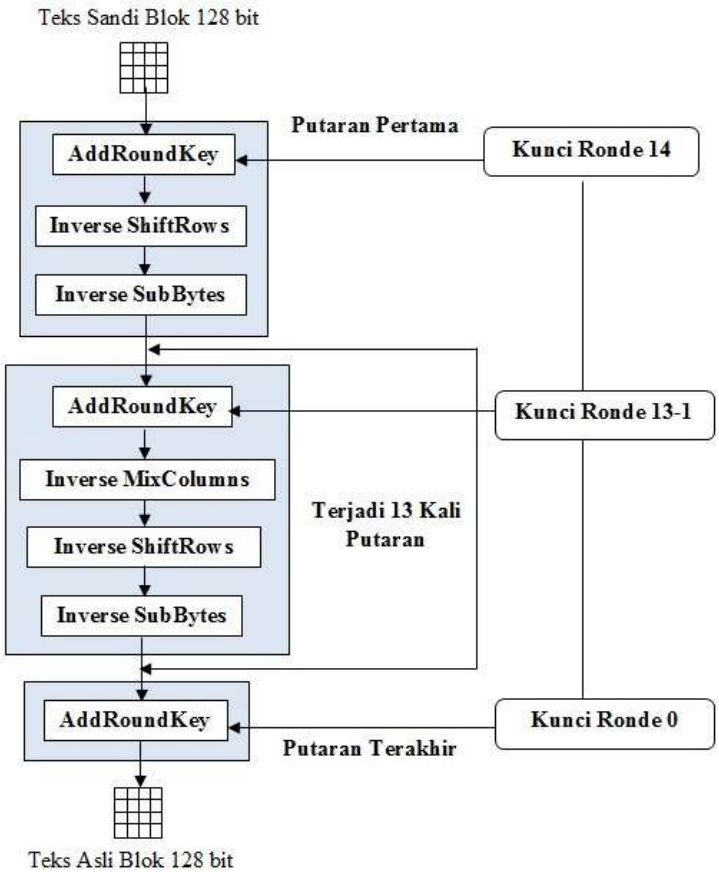
II.2 Algoritma *AES (Advanded Encryption Standard)*

Advanded Encryption Standard (AES) dipublikasikan oleh NIST (*National Institude of Standard and Technology*) pada tahun 2001. AES merupakan blok kode simetris untuk menggantikan DES (*Data Encryption Standard*).

AES mendukung berbagai variasi ukuran blok dan kunci yang akan digunakan. Namun AES mempunyai ukuran blok dan kunci yang tetap sebesar 128, 192, 256 bit. Pemilihan ukuran blok data dan kunci akan menentukan jumlah proses yang harus dilalui untuk proses enkripsi dan dekripsi. Berikut adalah perbandingan jumlah proses yang harus dilalui untuk masing-masing masukan.

Blok-blok data masukan dan kunci dioperasikan dalam bentuk *array*. Setiap anggota *array* sebelum menghasilkan keluaran *ciphertext* dinamakan dengan *state*. Setiap *state* akan mengalami proses yang secara garis besar terdiri dari empat tahap yaitu, *AddRoundKey*, *SubBytes*, *ShiftRows*, dan *MixColumns*. Kecuali tahap *MixColumns*, ketiga tahap lainnya akan diulang pada setiap proses sedangkan tahap *MixColumns* tidak akan dilakukan pada tahap terakhir. Proses dekripsi adalah kebalikkan dari dekripsi (Joko Tri Susilo Widodo, 2014)

Algoritma dekripsi AES dapat diilustrasikan sperti gambar dibawah. Secara ringkas algoritma deksripsi merupakan kebalikan algoritma enkripsi AES. Algoritma dekripsi AES menggunakan transformasi invers semua transformasi dasar yang digunakan pada algoritma enkripsi AES. Setiap transformasi dasar AES memiliki transformasi invers, yaitu : InvSubBytes, InvShiftRows dan InvMixColumns. AddRoundKey merupakan transformasi yang bersifat self-invers dengan syarat menggunakan kunci yang sama (Rifki Sadikin, 2012).



Gambar II.2 Flowchart Algoritma AES
(Sumber : Rifki Sadikin, 2012)

II.3 Android

Android adalah sekumpulan perangkat lunak yang ditujukan bagi perangkat bergerak mencakup sistem operasi, *middleware*, dan aplikasi kunci. *Android Standart Development Kid* (SDK) menyediakan perlengkapan dan *Application Programming Interface* (API) yang diperlukan untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

Android dikembangkan oleh *Google* bersama *Open Handset Alliance* (OHA) yaitu aliansi perngkat seluler terbuka yang terdiri dari 47 perusahaan *Hardware, Software* dan

perusaan telekomunikasi ditujukan untuk mengembangkan standar terbuka bagi perangkat selular.

Android adalah sistem operasi untuk telepon seluler yang berbasiskan Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri sehingga dapat digunakan oleh bermacam peranti penggerak. Awalnya Google Inc. membeli Android Inc. pendatang baru yang membuat *software* (perangkat lunak) untuk telepon genggam. Kemudian untuk mengembangkan Android di bentuklah *Open Handset Alliance* yang merupakan gabungan dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, TMobile, dan NVidia.

Pada saat perilisan perdana Android pada tanggal 5 november 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android dibawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Terdapat dua jenis distributor sistem operasi Android. Pertama yang dapat dukungan penuh dari Google atau *Google Mail Service* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung dari Google atau dikenal sebagai *Open Handset Distribution* (DHD) (Firdan Ardiansyah, 2011).

II.3.1 Sejarah dan Perkembangan Android

Pada mulanya terdapat berbagai macam sistem operasi pada perangkat selular, diantaranya sistem operasi *Symbian*, *Microsoft Windows Mobile*, *Mobile Linux*, *iPhone*, dan

sistem operasi lainnya. Namun diantara sistem operasi yang ada belum mendukung standar dan penerbitan API yang dapat dimanfaatkan secara keseluruhan dan dengan biaya yang murah. Kemudian *Google* ikut berkecimpung di dalamnya dengan *platform* Android, yang menjanjikan keterbukaan, keterjangkauan, *open source dan framework* berkualitas.

Pada tahun 2005, *Google* mengakuisisi perusahaan Android Inc. untuk mamulai perkembangan *platform* Android. Dimana terlibat pengembangan ini adalah Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Pada pertengahan 2007 sekelompok pemimpin industri bersama-sama membentuk analisis aliansi perangkat selular terbuka, *Open Handset Alliance* (OHA). Bagian dari tujuan aliansi ini adalah berinovasi dengan cepat dan menaggapi kebutuhan konsumen dengan lebih baik, dengan produk awalnya adalah *platform* Android. Dimana Android dirancang untuk melayani kebutuhan *operator* telekomunikasi, *manufaktur handset*, dan pengembangan aplikasi, OHA berkomitmen untuk membuat *Android open source* dengan lisensi Apache Versi 2.0.

Android pertama kali diluncurkan pada 5 November 2007, dan smartphone pertama yang menggunakan sistem operasi Android dikeluarkan oleh *T-Mobile* dengan sebutan G1 pada bulan September 2008. Hingga saat ini Android telah merilis beberapa versi Android untuk menyempurnakan versi sebelumnya. Selain berdsarkan penomoran, pada setiap versi Android terdapat kode nama berdasarkan nama-nama kue. Hingga saat ini sudah terdapat beberapa versi yang telah diluncurkan, diantaranya: versi 1.1 dirlis pada 9 maret 2009, versi 1.5 dirilis pada 30 April 2009 diberi nama *Cupcake*, versi 1.6 dirilis pada 15 September 2009 diberi nama *Donut*, versi 2.0 dirilis pada 26 Oktober 2009 diberi nama *Éclair*, versi 2.2 dirilis pada 20 Mei 2010 diberi nama *Froyo* (*Frozen Yoghurt*), versi 2.3 dirilis pada 6

Desember 2010 diberi nama *Gingerbread,* versi 3.0 dirilis pada Mei 2011 diberi nama *Honeycomb,* versi 4.0 dirilis pada 19 Oktober 2011 diberi nama ICS (*Ice Cream Sandwich*).

II.3.2 Kelebihan Android

Sudah banyak platform untuk perangkat selular saat ini, termasuk didalamnya *Symbian, iPhone, Windows Mobile, BlackBerry, Java Mobile Edition, Linux Mobile* (LiM0), dan banyak lagi. Namun ada beberapa hal yang menjadi kelebihan Android. Walaupun beberapa fitur - fitur yang ada telah muncul sebelumnya pada *platform* lain, Android adalah yang pertama menggabungkan hal seperti berikut:

1. Keterbukaan, Bebas pengembangan tanpa dikenakan biaya terhadap sistem karena berbasiskan Linux dan open source. Pembuat perangkat menyukai hal ini karena dapat membangun platform yang sesuai yang diinginkan tanpa harus membayar *royality*. Sementara pengembang software menyukai karena Android dapat digunakan diperangkat manapun dan tanpa terikat oleh vendor manapun.
2. Arsitektur komponen dasar Android terinspirasi dari teknologi internet Mashup. Bagian dalam sebuah aplikasi dapat digunakan oleh aplikasi lainnya, bahkan dapat diganti dengan komponen lain yang sesuai dengan aplikasi yang dikembangkan.
3. Banyak dukungan *service,* kemudahan dalam menggunakan berbagai macam layanan pada aplikasi seperti penggunaan layanan pencarian lokasi, *database SQL,* browser dan penggunaan peta. Semua itu sudah tertanam pada Android sehingga memudahkan dalam pengembangan aplikasi.
4. Siklus hidup aplikasi diatur secara otomatis, setiap program terjaga antara satu sama lain oleh berbagai lapisan keamanan, sehingga kerja sistem menjadi lebih stabil. Pengguna tak perlu kawatir dalam menggunakan aplikasi pada perangkat yang memorinya terbatas.

5. Dukungan grafis dan suarat terbaik, dengan adanya dukungan 2D grafis dan animasi yang diilhami oleh *Flash* menyatu dalam 3D menggunakan *OpenGL* memungkinkan membuat aplikasi maupun game yang berbeda.

Portabilitas aplikasi, aplikasi dapat digunakan pada perangkat yang ada saat ini maupun yang akan datang. Semua program ditulis dengan menggunakan bahas pemrograman Java dan dieksekusi oleh mesin virtual Dalvik, sehingga kode program *portabel* antara ARM, X86, dan arsitektur lainnya. Sama halnya dengan dukungan masukan seperti penggunaan *Keyboard,* layar sentuh, *trackball* dan resolusi layar semua dapat disesuaikan dengan program.

II.4 Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform* (*platformindependent*) (Ardzi Firman Ihtiyar Rohanianto, 2014). Berikut ini adalah sifat dari Eclipse:

1. *Multi-platform*: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. *Multi-language*: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
3. *Multi-role*: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in* (Ardzi Firman Ihtiyar Rohanianto, 2014).