

BAB II

TINJAUAN PUSTAKA

II.1 Kriptografi

II.1.1 Sejarah Kriptografi

Menurut Doni Ariyus (2008:13), dalam bukunya “*Pengantar Ilmu Kriptografi Teori, Analisis dan implementasi*” menjelaskan bahwa kriptografi memiliki sejarah yang panjang dan mengagumkan. Penulisan rahasia ini dapat dilacak kembali ke 3000 tahun SM saat digunakan oleh bangsa Mesir. Mereka menggunakan *hieroglyphcs* untuk menyembunyikan tulisan dari mereka yang tidak diharapkan. *Hieroglyphcs* diturunkan dari bahasa Yunani *hieroglyphica* yang berarti ukiran rahasia. *Hieroglyphcs* berevolusi menjadi hieratic, yaitu *stylized script* yang lebih mudah untuk digunakan. Sekitar 400 SM, kriptografi militer digunakan oleh bangsa Spartan dalam bentuk sepotong papyrus atau perkamen dibungkus dengan batang kayu. Sistem ini disebut Scytale.

Sekitar 50 SM, Julius Caesar, kaisar Roma, menggunakan *Cipher substitusi* untuk mengirim pesan ke Marcus Tullius Cicero. Pada *Cipher* ini, huruf-huruf alfabet disubstitusi dengan huruf-huruf yang lain pada alfabet yang sama. Karena hanya satu alfabet yang digunakan, *Cipher* ini merupakan substitusi monoalfabetik. *Cipher* semacam ini mencakup penggeseran alfabet dengan 3 huruf dan mensubstitusikan huruf tersebut. Substitusi ini kadang dikenal dengan C3 (untuk Caesar menggeser 3 tempat). Secara umum sistem *Cipher* Caesar dapat ditulis sebagai berikut :

$$Z_i = C_n(P_i) \dots \dots \dots (P_n) \dots \dots \dots (1)$$

Dimana Z_i adalah karakter-karakter *Ciphertext*, C_n adalah transformasi substitusi alfabetik, n adalah jumlah huruf yang digeser, dan P_i adalah karakter-karakter *plaintext*. Disk mempunyai peranan penting dalam kriptografi sekitar 500 th yang lalu. Di Italia sekitar tahun 1460, Leon Battista Alberti mengembangkan disk *Cipher* untuk enkripsi. Sistemnya terdiri dari dua disk konsentris. Setiap disk memiliki alfabet di sekelilingnya, dan dengan memutar satu disk berhubungan dengan yang lainnya, huruf pada satu alfabet dapat ditransformasi ke huruf pada alfabet yang lain.

Bangsa Arab menemukan cryptanalysis karena kemahirannya dalam bidang matematika, statistik, dan linguistik. Karena setiap orang muslim harus menambah pengetahuannya, mereka mempelajari peradaban terdahulu dan mendekodekan tulisan-tulisannya ke huruf-huruf Arab. Pada tahun 815, Caliph al-Mamun mendirikan *House of Wisdom* di Baghdad yang merupakan titik pusat dari usaha-usaha translasi. Pada abad ke-9, filsuf Arab al-Kindi menulis risalat (ditemukan kembali th 1987) yang diberi judul "*A Manuscript on DeCiphering Cryptographic Messages*". Pada 1790, Thomas Jefferson mengembangkan alat enkripsi dengan menggunakan tumpukan yang terdiri dari 26 disk yang dapat diputar secara individual. Pesan dirakit dengan memutar setiap disk ke huruf yang tepat dibawah batang berjajar yang menjalankan panjang tumpukan disk. Kemudian, batang berjajar diputar dengan sudut tertentu, A , dan huruf-huruf dibawah batang adalah pesan yang terenkripsi. Penerima akan menjajarkan

karakter-karakter *Cipher* dibawah batang berjajar, memutar batang kembali dengan sudut A dan membaca pesan *plaintext*.

Sistem disk digunakan secara luas selama perang sipil US. *Federal Signal Officer* mendapatkan hak paten pada sistem disk mirip dengan yang ditemukan oleh Leon Battista Alberti di Italia, dan dia menggunakannya untuk mengkode dan mendekodekan sinyal-sinyal bendera diantara unit-unit. Sistem Unix menggunakan *Cipher substitusi* yang disebut ROT 13 yang menggeser alfabet sebanyak 13 tempat. Penggeseran 13 tempat yang lain membawa alfabet kembali ke posisi semula, dengan demikian mendekodekan pesan. Mesin kriptografi mekanik yang disebut Hagelin Machine dibuat pada tahun 1920 oleh Boris Hagelin di Scockholm, Swedia. Di US, mesin Hagelin dikenal sebagai M-209. Pada tahun 20-an, Herbert O. Yardley bertugas pada organisasi rahasia US MI-8 yang dikenal sebagai "*Black Chamber*". MI-8 menjebol kode-kode sejumlah negara. Selama konferensi Angkatan Laut Washington tahun 1921-1922, US membatasi negosiasi dengan Jepang karena MI-8 telah memberikan rencana negosiasi Jepang yang telap disadap kepada sekretaris negara US. Departemen negara menutup MI-8 pada tahun 1929 sehingga Yardley merasa kecewa. Sebagai wujud kekecewaanya, Yardley menerbitkan buku *The American Black Chamber*, yang menggambarkan kepada dunia rahasia dari MI-8. Sebagai konsekuensinya, pihak Jepang menginstal kode-kode baru. Karena kepeloporannya dalam bidang ini, Yardley dikenal sebagai "Bapak Kriptografi Amerika".

II.1.2 Taksonomi Primitif-primitif Kriptografi

Menurut Doni Ariyus (2008:13), Ada beberapa dasar tool kriptografi (primitif) yang digunakan untuk mendukung keamanan informasi. Contoh dari primitif termasuk skema enkripsi, fungsi hash, dan skema tanda tangan digital. Primitif-primitif ini harus dapat dievaluasi berdasarkan beberapa kriteria seperti:

1. Level keamanan. Hal ini biasanya sulit untuk dihitung. Sering diwakili dengan jumlah operasi yang dibutuhkan (menggunakan metode terbaik yang diketahui) untuk melawan tujuan yang diharapkan. Level keamanan biasanya didefinisikan *work factor*.
2. *Fungsionalitas*. Primitif-primitif dibutuhkan untuk memenuhi tujuan keamanan informasi yang bermacam-macam. Primitif mana yang paling efektif untuk tujuan yang diberikan akan ditentukan dengan properti dasar dari primitif.
3. Metode operasi. Primitif, saat diterapkan dengan bermacam cara dan dengan bermacam input, biasanya akan menunjukkan karakteristik yang berbeda, sehingga satu primitif dapat menyediakan fungsionalitas yang sangat berbeda pada mode operasi atau penggunaannya.
4. Unjuk kerja. Merupakan efisiensi sebuah primitif pada mode tertentu. (sebagai contoh algoritma enkripsi dapat dihitung dengan jumlah bit per detik yang dapat dienkrripsinya).
5. Kemudahan implementasi. Merupakan kesulitan dalam merealisasikan primitif pada prakteknya. Dapat meliputi kompleksitas pengimplementasian primitif dalam lingkungan *software* maupun *hardware*.

Kepentingan relatif dari bermacam kriteria ini sangat tergantung pada aplikasi dan sumber daya yang tersedia.

II.1.3 Enkripsi Kunci Rahasia

Menurut Doni Ariyus (2008:13), *Secret-key cryptography* kadang disebut sebagai *symmetric cryptography* merupakan bentuk kriptografi yang lebih tradisional, dimana sebuah kunci tunggal dapat digunakan untuk mengenkrip dan mendekrip pesan. *Secret-key cryptography* tidak hanya berkaitan dengan enkripsi tetapi juga berkaitan dengan otentikasi, disebut juga *message authentication Codes*.

Masalah utama yang dihadapi *secret-key cryptography* adalah membuat pengirim dan penerima menyetujui kunci rahasia tanpa ada orang lain yang mengetahuinya. Ini membutuhkan metode dimana dua pihak dapat berkomunikasi tanpa takut akan disadap. Kelebihan *secret-key cryptography* dari *public-key cryptography* adalah lebih cepat. Teknik yang paling umum dalam *secret-key cryptography* adalah *block Ciphers*, *stream Ciphers*, dan *message authentication Codes*.

Berdasarkan jenis kunci yang digunakannya, algoritma kriptografi dikelompokkan menjadi dua bagian, yaitu

1. *Symmetric Algorithm*

Symmetric algorithm atau disebut juga *secret key algorithm* adalah algoritma yang kunci enkripsinya dapat dihitung dari kunci dekripsi dan begitu pula sebaliknya, kunci dekripsi dapat dihitung dari kunci enkripsi. Pada sebagian

besar *symmetric algorithm* kunci enkripsi dan kunci dekripsi adalah sama. *Symmetric algorithm* memerlukan kesepakatan antara pengirim dan penerima pesan pada suatu kunci sebelum dapat berkomunikasi secara aman. Keamanan *symmetric algorithm* tergantung pada rahasia kunci. Pemecahan kunci berarti memungkinkan setiap orang dapat mengenkripsi dan mendekripsi pesan dengan mudah. *Symmetric algorithm* dapat dikelompokkan menjadi dua jenis, yaitu *stream Cipher* dan *block Cipher*. *Stream Cipher* beroperasi bit per bit (atau byte per byte) pada satu waktu. Sedangkan *block Cipher* beroperasi per kelompokkelompok bit yang disebut blok (*block*) pada satu waktu.

2. *Asymmetric Algorithm*

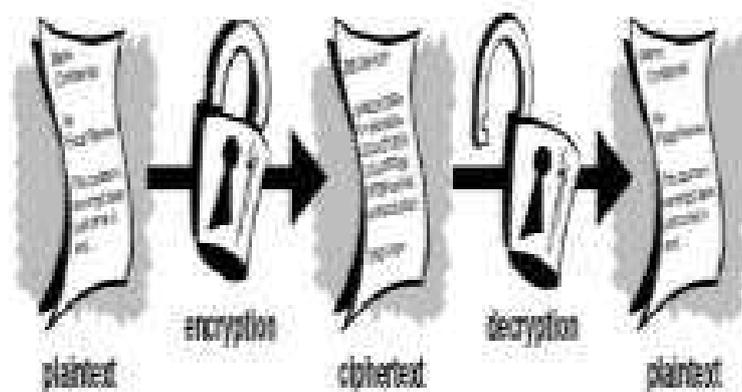
Asymmetric algorithm atau disebut juga *public key algorithm* didesain agar memudahkan dalam distribusi kunci yang digunakan untuk enkripsi dan dekripsi. Kunci dekripsi pada *public key algorithm* secara praktis tidak dapat dihitung dari kunci enkripsi. Algoritma ini disebut “*public key*” karena kunci dapat dibuat menjadi publik. Setiap orang dapat menggunakan kunci enkripsi untuk mengenkripsi pesan, tetapi hanya orang yang memiliki kunci dekripsi yang dapat mendekripsi pesan tersebut. Pada sistem ini kunci enkripsi sering disebut kunci publik (*public key*), dan kunci dekripsi disebut kunci rahasia (*private key*).

II.1.4 Prinsip Dasar Kriptografi

Ilmu kriptografi adalah ilmu yang mempelajari tentang penyembunyian huruf atau tulisan sehingga membuat tulisan tersebut tidak dapat dibaca oleh

orang yang tidak berkepentingan. Kriptografi sudah dipakai sejak jaman Julius Caesar dimana akan mengirimkan pesan kepada panglimanya tetapi tidak mempercayai kurir pembawa pesan tersebut.

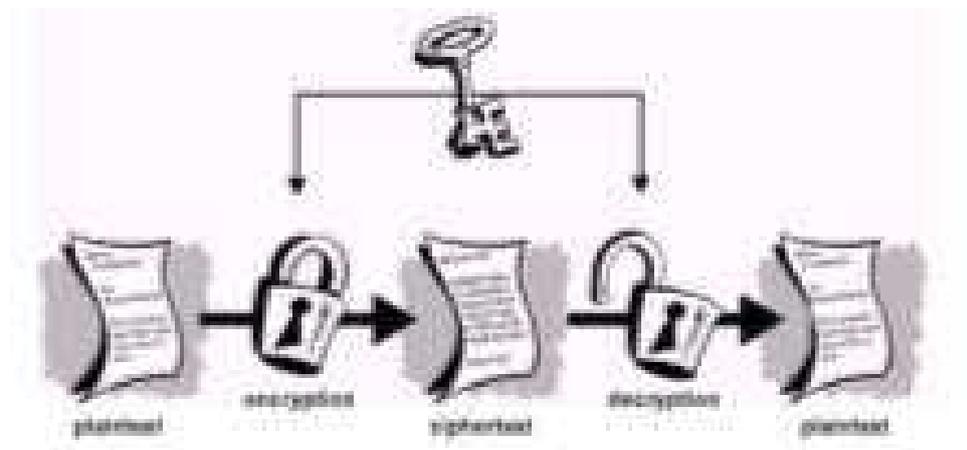
Kriptografi mempunyai 2 (dua) bagian yang penting, yaitu enkripsi dan dekripsi. Enkripsi adalah proses dari penyandian pesan asli menjadi pesan yang tidak dapat diartikan seperti aslinya. Dekripsi sendiri berarti merubah pesan yang sudah disandikan menjadi pesan aslinya. Pesan asli biasanya disebut *plaintext*, sedangkan pesan yang sudah disandikan disebut *ciphertext*. Pada Gambar 1 dapat dilihat bahwa masukan berupa *plaintext* akan masuk ke dalam blok enkripsi dan keluarannya akan berupa *ciphertext*, kemudian *ciphertext* akan masuk ke dalam blok dekripsi dan keluarannya akan kembali menjadi *plaintext* semula.



Gambar II.1 : Proses Enkripsi dan Dekripsi

(Sumber : M. Sholeh dan J.V. Hamokwarong, *Aplikasi Kriptografi Dengan Metode Vernam Cipher dan Metode Permutasi Biner*, 2011)

Ada 2 (dua) model algoritma enkripsi yang menggunakan kunci, yaitu kunci simetrik dan kunci asimetrik. Enkripsi kunci simetrik yang biasanya disebut enkripsi konvensional adalah enkripsi yang menggunakan kunci yang sama untuk enkripsi maupun dekripsi, dari Gambar diatas terlihat bahwa untuk mengenkripsi maupun mendekripsi pesan hanya menggunakan satu buah kunci (K) saja.

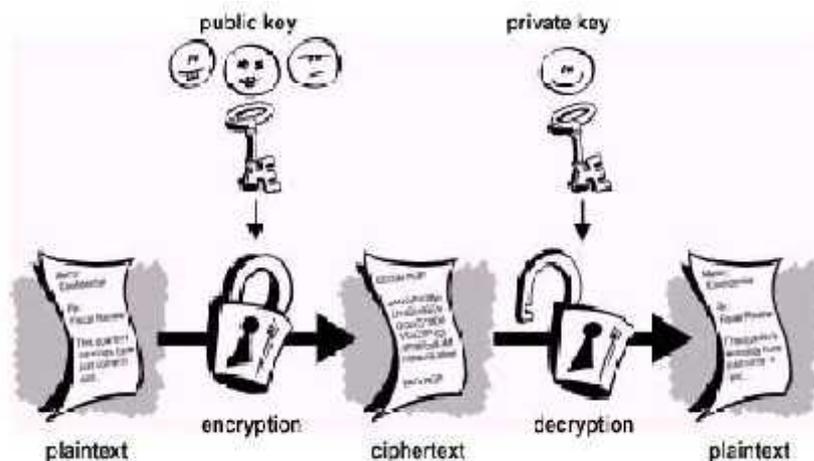


Gambar II.2 : Enkripsi-dekripsi Kunci Simetrik

Sumber : M. Sholeh dan J.V. Hamokwarong, 2011

Penggunaan metode ini membutuhkan persetujuan antara pengirim dan penerima tentang kunci sebelum mereka saling mengirim pesan. Keamanan dari kunci simetrik tergantung pada kerahasiaan kunci, apabila seorang penyusup dapat menemukan kunci maka dengan mudah dapat membaca pesan yang sudah dienkripsi. Enkripsi kunci simetrik dapat dibagi kedalam 2 (dua) kelompok yaitu metode *stream cipher* dan metode *block cipher*. Enkripsi kunci asimetrik (biasa disebut enkripsi kunci publik) dibuat sedemikian rupa sehingga kunci yang dipakai untuk enkripsi berbeda dengan kunci yang dipakai untuk dekripsi.

Enkripsi kunci *public* disebut demikian karena kunci untuk enkripsi boleh disebarluaskan kepada umum sedangkan kunci untuk mendekripsi hanya disimpan oleh orang yang bersangkutan. Contohnya seperti pada gambar diatas bila seseorang ingin mengirim pesan kepada orang lain maka orang tersebut menggunakan kunci *public* orang tersebut untuk mengenkripsi pesan yang kita kirim kepadanya lalu orang tersebut akan mendekripsi pesan tersebut dengan kunci privat miliknya.



Gambar II.3 : Enkripsi Kunci Asimetrik

(Sumber : M. Sholeh dan J.V. Hamokwarong, *Aplikasi Kriptografi Dengan Metode Vernam Cipher dan Metode Permutasi Biner*, 2011)

II.1.5 Tujuan Kriptografi

Dalam teknologi informasi telah dan sedang dikembangkan cara-cara untuk menangkal berbagai serangan, seperti penyadap dan perubahan data yang sedang dikirimkan. Salah satu cara yang ditempuh untuk mengatasi masalah ini adalah dengan menggunakan kriptografi yang menggunakan transformasi data sehingga data yang dihasilkan tidak dapat dimengerti oleh pihak yang tidak

berhak mengakses. Transformasi ini memberikan solusi pada dua macam masalah keamanan data, yaitu masalah privasi (*privacy*) dan keotentikan (*authentication*).

Privacy mengandung arti bahwa data yang dikirimkan hanya dapat dimengerti oleh penerima yang sah atau berhak. Sedangkan keotentikan mencegah pihak ketiga untuk mengirimkan data yang salah atau mengubah data yang dikirimkan.

Ada empat tujuan mendasar dari ilmu kriptografi yang merupakan aspek keamanan informasi yaitu :

1. Kerahasiaan

Merupakan layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka atau mengupas informasi yang telah disandi.

2. Integritas data

Berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain kedalam data yang sebenarnya.

3. Autentikasi

Berhubungan dengan identifikasi / pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain.

4. Non-repudiasi

Usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman atau terciptanya suatu informasi oleh yang mengirimkan / membuat.

II.1.6 Pola-pola Penyaringan Data

Proteksi data dan informasi dalam komunikasi komputer menjadi penting karena nilai informasi itu sendiri dan meningkatnya penggunaan komputer di berbagai sektor. Melihat kenyataan semakin banyak data yang diproses dengan komputer dan dikirim melalui perangkat komunikasi elektronik maka ancaman terhadap pengamanan data akan semakin meningkat. Beberapa pola ancaman terhadap komunikasi data dalam komputer dapat diterangkan sebagai berikut :

a. Interruption

Interception terjadi bila data yang dikirimkan dari A tidak sampai pada orang yang berhak B. *Interruption* merupakan pola penyerangan terhadap sifat *availability* (ketersediaan data)

b. Interception

Serangan ini terjadi bila pihak ketiga C berhasil membaca data yang dikirimkan. *interception* merupakan pola penyerangan terhadap sifat *confidentiality* (kerahasiaan data)

c. Modification

Pada serangan ini pihak ketiga C berhasil merubah pesan yang dikirimkan. *Modification* merupakan pola penyerangan terhadap sifat *integrity* (keaslian data)

d. Fabrication

Pada serangan ini, penyerang berhasil mengirimkan data ke tujuan dengan memanfaatkan identitas orang lain. *Fabrication* merupakan pola penyerangan terhadap sifat *authenticity*

Ancaman-ancaman tersebut di atas menjadi masalah terutama dengan semakin meningkatnya komunikasi data yang bersifat rahasia seperti: pemindahan dana secara elektronik kepada dunia perbankan atau pengiriman dokumen rahasia pada instansi pemerintah. Untuk mengantisipasi ancaman-ancaman tersebut perlu dilakukan usaha untuk melindungi data yang dikirim melalui saluran komunikasi salah satunya adalah dengan teknik enkripsi. Dan untuk masalah kekuatan pengamanannya tergantung pada algoritma metode enkripsi tersebut dan juga kunci yang digunakan di dalamnya.

II.2 One Time Pad

Algoritma *One Time Pad* (OTP) merupakan algoritma berjenis *symetric key* yang artinya bahwa kunci yang digunakan untuk melakukan enkripsi dan dekripsi merupakan kunci yang sama. Dalam proses enkripsi, algoritma ini menggunakan cara stream cipher yang berasal dari hasil *XOR* antara bit plaintext dan bit key. Pada metode ini plain text diubah kedalam kode ASCII dan kemudian dikenakan operasi *XOR* terhadap kunci yang sudah diubah ke dalam kode ASCII (M. Sholeh, 2011).

Proses dekripsi merupakan proses yang dilakukan untuk mengembalikan *file* dari bentuk simbol-simbol kembali ke bentuk semula. Dekripsi Vernam

Chiper dapat dilakukan dengan menggunakan rumus dibawah ini (Endro Ariyanto, 2008):

$$P(i) = C(i) \text{ XOR } K(i)$$

Dalam hal ini :

C = Chiper Teks

P = Plain Teks

K = Kunci

Enkripsi data merupakan bagian awal dari proses pengamanan data. Dalam proses enkripsi ini data yang asli akan dilakukan proses pengacakan dengan algoritma yang sudah ditentukan, adapun proses enkripsi pada Vernam Chiper dapat dilakukan dengan menggunakan rumus dibawah ini (Endro Ariyanto, 2008) :

$$C(i) = P(i) \text{ XOR } K(i)$$

Dalam hal ini :

C = Chiper Teks

P = Plain Teks

K = Kunci

I = index karakter

II.3 Sejarah Java

Bahasa pemrograman Java terlahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak. Proyek ini dimotori oleh

Patrick Naughton, Mike Sheridan, dan James Gosling, beserta sembilan pemrogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah maskot Duke yang dibuat oleh Joe Palrang. Pertemuan proyek berlangsung di sebuah gedung perkantoran Sand Hill Road di Menlo Park. Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program Java Oak pertama, yang ditujukan sebagai pengendali sebuah peralatan dengan teknologi layar sentuh (touch screen), seperti pada PDA sekarang ini. Teknologi baru ini dinamai "*7" (Star Seven). Setelah era Star Seven selesai, sebuah anak perusahaan Tv kabel tertarik ditambah beberapa orang dari proyek The Green Project. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, Palo Alto. Perusahaan baru ini bertambah maju: jumlah karyawan meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada rentang waktu ini juga ditetapkan pemakaian Internet sebagai medium yang menjembatani kerja dan ide di antara mereka. Pada awal tahun 1990an, Internet masih merupakan rintisan, yang dipakai hanya di kalangan akademisi dan militer. Mereka menjadikan perambah (browser) Mosaic sebagai landasan awal untuk membuat perambah Java pertama yang dinamai Web Runner, terinspirasi dari film 1980-an, Blade Runner.

Pada perkembangan rilis pertama, Web Runner berganti nama menjadi Hot Java. Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar San Jose Mercury News pada tanggal 23 Mei 1995. Sayangnya terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah

ruangan hotel Sheraton Palace. Tiga dari pimpinan utama proyek, Eric Schmidt dan George Paolini dari Sun Microsystems bersama Marc Andreessen, membentuk Netscape

Nama Oak, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "Bapak Java", James Gosling. Nama Oak ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemrograman Java tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah Java)

(Sumber : Pengenalan Java Programming, Dany Setiawan, 2008)

II.3.1 Teknologi Java

Sebagai sebuah bahasa pemrograman, Java dapat membuat seluruh bentuk aplikasi, desktop, web dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain. Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai platform sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat open source.

1. Platform Java

Java sebagai platform pengembangan software, secara garis besar dibedakan ke dalam 3 arahan :

a. J2SE

Java 2 Standard Edition mencakup *core* dari bahasa pemrograman Java, memuat library-library inti yang dibutuhkan seperti IO, Networking dan JDBC.

b. J2EE

Java 2 Enterprise Edition adalah pengembangan Java untuk solusi enterprise, mulai dari aplikasi berbasis Web dengan Servlet dan JSP, aplikasi terdistribusi dengan EJB, sebagaimana aplikasi integrasi enterprise seperti *Web Service*.

c. J2ME

Java 2 Micro Edition adalah pengembangan Java untuk mobile device, seperti handphone, pocket PC dan PDA. Pengembangan ke arah mobile device ini menuntut Java untuk beradaptasi dengan mesin yang terbatas dalam memory dan processor

2. *Development Environment*

Sebagai sebuah peralatan pembangun, teknologi Java menyediakan banyak *tools compiler, interpreter, penyusun dokumentasi, paket kelas* dan sebagainya.

3. *Aplikasi*

Aplikasi dengan teknologi Java secara umum adalah aplikasi serba guna yang dapat dijalankan pada seluruh mesin yang memiliki Java Runtime Environment (JRE).

4. *Deployment Environment*

(Sumber : *Pengenalan Java Programming, Dany Setiawan, 2008*)

II.3.2 Aplikasi Eclipse

Eclipse merupakan komunitas *Open source* yang bertujuan menghasilkan *platform* pemrograman terbuka. Eclipse terdiri dari *framework* yang dapat dikembangkan lebih lanjut, peralatan bantu untuk membuat dan manage *software* sejak awal hingga diluncurkan. *Platform* Eclipse didukung oleh ekosistem besar yang terdiri dari vendor teknologi, *start-up inovatif*, universitas, riset institusi serta individu.

Banyak orang mengenal Eclipse sebagai IDE (*Integrated Development Environment*) untuk bahasa Java, tapi Eclipse lebih dari sekedar IDE untuk Java. Komunitas Eclipse memiliki lebih dari 60 proyek *Open source*. Proyek-proyek ini secara konsep terbagi menjadi 7 kategori :

1. Enterprise Development
2. Embedded and Device Development
3. Rich Client Platform
4. Rich Internet Applications
5. Application Frameworks
6. Application Lifecycle Management (ALM)
7. Service Oriented Architecture (SOA)

Secara umum Eclipse digunakan untuk membangun *software inovatif* berstandar industri, dan alat bantu beserta *frameworknya* membantu pekerjaan menjadi lebih mudah.

II.3.3 Lisensi

Eclipse menggunakan EPL (*Eclipse Public License*), yaitu lisensi yang memungkinkan organisasi untuk menjadikan Eclipse sebagai produk komersialnya, dan pada saat yang sama meminta orang yang melakukan perubahan untuk mengkontribusikan hasilnya kembali kepada komunitas.

II.3.4 Instalasi

1. Anda membutuhkan Java 5 JRE untuk menjalankan Eclipse.
2. Download [Eclipse IDE for Java Developers](#) untuk menggunakan kode pada situs Belajar Java ini.
3. Gunakan *utility* pada sistem operasi anda untuk membuka *kompresi file* tersebut ke dalam *hard disk* anda.
4. **Catatan untuk Windows:** Apabila Anda menggunakan *utilitas* kompresi file yang berasal dari Windows XP atau Windows Vista itu sendiri, kadang kala utilitas tersebut tidak berhasil membuka file dengan nama yang panjang. Jika Anda mengalami masalah dekompresi Eclipse pada Windows, letakkan hasil *dekompresi* pada *root directory* (misalnya C:\eclipse) atau gunakan *software dekompresi* lain yang gratis seperti [7-Zip](#)

II.3.5 Menjalankan Eclipse

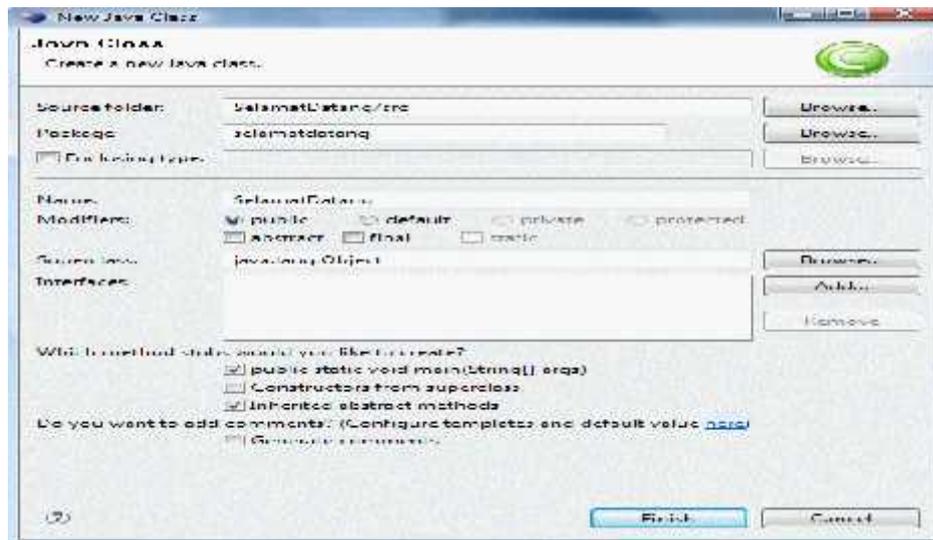
1. Cari file bernama *eclipse.exe* (pada Windows) atau *eclipse* (pada Ubuntu) kemudian *double-click*

2. Pada saat Eclipse pertama kali dijalankan, Eclipse akan menanyakan *workspace*, yaitu folder tempat proyek dan data diletakkan. Anda bisa menempatkan di mana saja asalkan jangan di dalam folder Eclipse itu sendiri.
3. *Click Browse* dan pilih folder yang ada inginkan. Tik "*Use this as default and do not ask again*"
4. Halaman pembuka akan muncul. Klik "*Workspace*", tombol paling kanan berbentuk anak panah untuk masuk ke dalam *workspace* Anda.

II.3.6 Membuat Program Java

1. Klik "**File -> New -> Java Project**"
2. Isi nama proyek (misalnya SelamatDatang), kemudian klik "**Finish**"
3. Setelah Eclipse membuat proyek untuk Anda, di bagian kiri *workspace* Anda akan melihat struktur direktori proyek Anda yang dimulai dengan nama proyek, folder *src*, dan folder *JRE System Library*
4. Klik kanan pada folder *src*, kemudian "**New -> Package**"
5. Isi nama *package* (misalnya selamatdatang), kemudian klik "**Finish**"
6. Klik kanan lagi pada folder selamatdatang, kemudian "**New -> Class**"
7. Isi nama *class* (misalnya SelamatDatang)
8. Karena *class* ini adalah *class* utama yang akan langsung dijalankan oleh JRE (*Java Runtime Environment*), click "*public static void main(String[] args)*" pada bagian "Which method stubs would you like to create?"

Berikut ini adalah form untuk membuat *project* baru dalam aplikasi eclipse yang dapat dilihat pada gambar 2.1 dibawah ini



Gambar II.4 : New Project

Sumber : Widiyanto Pratama, Tutorial Android Programming, 2013

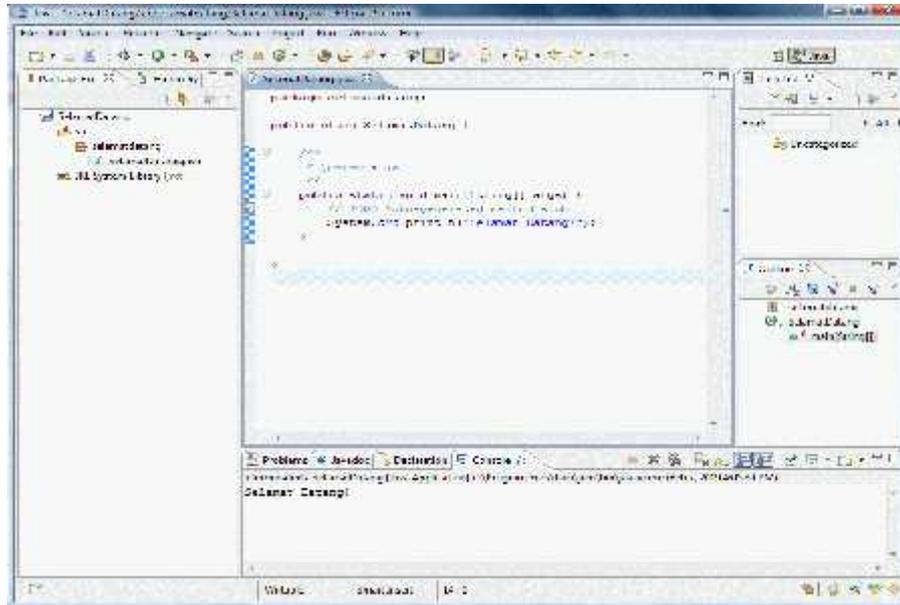
9. Klik "**Finish**"
10. Eclipse akan membuat program kosong yang berisi *package* dan class sesuai dengan nama yang Anda masukkan pada tahap sebelumnya
11. Sekarang ketik program berikut di bawah "`// TODO`"


```
System.out.println("Selamat Datang!");
```
12. Kemudian simpan hasilnya

II.3.7 Menjalankan program Java

- a. Untuk menjalankan program Anda, klik "Run -> Run"
- b. Di bagian bawah pada tab yang berjudul "Console" hasil program Anda ditampilkan
- c. Program ini akan menampilkan tulisan Selamat Datang! seperti pada gambar berikut ini

Berikut ini adalah form untuk menjalankan *project* baru dalam aplikasi eclipse yang dapat dilihat pada gambar dibawah ini



Gambar II.5 : Editor Eclipse
Sumber : Widiyanto Pratama, 2013

II.4 Sistem Operasi Android

Android adalah sistem operasi mobile yang berjalan pada kernel **Linux**, yang dirilis pada 21 Oktober 2008. Awalnya, sistem operasi ini dikembangkan oleh Android, Inc, yang kemudian dibeli oleh *Google*, dan yang terakhir, sistem operasi ini dibeli oleh *Open Handset Alliance*, sebuah consortium dari 47 perusahaan *hardware*, *software*, dan *telecom* yang didirikan untuk membuat *Open standard* bagi perangkat lunak mobile. Sistem operasi ini bersifat *free* dan *Open source*. Perangkat mobile yang mendukung sistem operasi ini di antaranya adalah HTC Dream dan HTC Magic, ponsel keluaran vendor asal Taiwan, HTC.

Awalnya, *Google* membeli Android, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah

Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia*. *Google* sendiri ternyata mempunyai alasan cukup kuat untuk melirik pangsa ini, karena perkembangan teknologi telepon seluler dewasa ini sudah bukan merupakan evolusi lagi, melainkan sebuah revolusi. Babak baru dalam dunia telekomunikasi *nirkabel* ini terus bergulir dengan cepat. Jika sekarang seseorang mempunyai PC di rumah, dan notebook untuk ke kantor atau kuliah, serta berkomunikasi melalui telepon seluler. Maka pergerakan yang kemudian terjadi sekarang adalah, orang mulai berpikir bagaimana menyatukan semuanya dalam satu genggam.

Sebenarnya hal tersebut telah mulai dipenuhi dengan munculnya PDA/*smartphone*, di mana seseorang dapat merangkum semua kebutuhan komputasinya dalam satu genggam. Dan perkembangan inilah yang membuat *Google* berambisi untuk menguasai pangsa ini. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Adapun Fitur-fitur Android adalah sebagai berikut :

- a. *Framework Aplikasi* yang mendukung penggantian komponen dan *reusable*.
- b. Mesin *virtual Dalvik* dioptimalkan untuk perangkat *mobile*.
- c. *Integrated browser* berdasarkan *engine Open source WebKit*.
- d. Grafis yang dioptimalkan dan didukung oleh perpustakaan grafis 2D, grafis 3D berdasarkan spesifikasi *OpenGL ES 1,0* (Opsional akselerasi hardware).
- e. *SQLite* untuk penyimpanan data.

- f. Media Support yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- g. GSM Telephony (tergantung hardware).
- h. Bluetooth, EDGE, 3G, dan WiFi (tergantung hardware).
- i. Kamera, GPS, kompas, dan accelerometer (hardware tergantung).

II.4.1 Versi Android

1. Android versi awal (2007 – 2008)

Pada September 2007 *Google* mengajukan hak paten aplikasi telepon seluler. *Google* mengenalkan *Nexus One*, salah satu jenis telepon pintar GSM yang menggunakan Android pada sistem operasinya. Telepon seluler ini diproduksi oleh *HTC Corporation* dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, *Atheros Communications*, diproduksi oleh *Asustek Computer Inc*, *Garmin Ltd*, *Softbank*, *Sony Ericsson*, *Toshiba Corp*, dan *Vodafone Group Plc*. Seiring pembentukan *Open Handset Alliance*, OHA mengumumkan produk perdana mereka, Android, perangkat bergerak (*mobile*) yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru. *Smartphone* yang memakai sistem operasi Android adalah *HTC Dream*, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android.

b. Android versi 1.1

Pada 9 Maret 2009, *Google* merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan email.

c. Android versi 1.5 (Cupcake)

Pada pertengahan Mei 2009, *Google* kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke *Youtube* dan gambar ke *Picasa* langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

d. Android versi 1.6 (Donut)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang diintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*; kemampuan dial kontak; teknologi text to change speech (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

e. Android versi 2.0 / 2.1 (Éclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1. Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (killer apps - aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik. Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

f. Android versi 2.2 (Froyo : Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan *Adobe Flash* 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 *JavaScript engine* yang dipakai *Google Chrome* yang mempercepat kemampuan rendering pada browser, pemasangan aplikasi

dalam SD Card, kemampuan *WiFi Hotspot portabel*, dan kemampuan auto update dalam aplikasi Android Market.

g. Android versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (*Gingerbread*) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (*User Interface*) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (*reverb, equalization, headphone virtualization, dan bass boost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

h. Android versi 3.0 / 3.1 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet. *Honeycomb* juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. *Tablet* pertama yang dibuat dengan menjalankan *Honeycomb* adalah *Motorola Xoom*. Perangkat tablet dengan platform Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama *Eee Pad Transformer* produksi dari Asus. Rencana masuk pasar Indonesia pada Mei 2011.

i. Android versi 4.0 (Ice Cream)

Android versi 4.0 akan dirilis akhir tahun 2011. Setelah ketahu versi Android ini perlu diketahui bahwa nama lain dari versi-versi tersebut diambil oleh *Google* dari nama makanan penutup.

II.4.2 Kelebihan Android

Adapun kelebihan Android adalah sebagai berikut ini :

1. Lengkap (*Complete Platform*) : Android dikatakan lengkap karena Android menyediakan tools untuk membangun software yang sangat lengkap dibanding dengan platform lain. Para pengembang dapat melakukan pendekatan yang komprehensif ketika mereka mengembangkan suatu aplikasi pada platform Android.
2. Terbuka (*Open Source Platform*) : Platform Android diciptakan dibawah lisensi *Open source*, dimana para pengembang bebas untuk mengembangkan aplikasi pada platform ini. Android menggunakan Linux kernel 2.6.
3. Bebas (*Free Platform*) : Android adalah *platform mobile* yang tidak memiliki batasan dalam mengembangkan aplikasinya. Tidak ada lisensi dalam mengembangkan aplikasi Android. Android dapat didistribusikan dan diperdagangkan dalam bentuk apapun.
4. Dapat menulis file di SD card.
5. Dapat mengolah database dengan SQLite.
6. *Application framework* berbasis komponen yang memudahkan reuse.
7. *Dalvik virtual machine* dioptimisasi untuk menjalankan program Java di mobile devices
8. *Integrated browser* berbasis WebKit engine
9. *Optimized graphics* tersedia 2D graphics library; dan *OpenGL ES 1.0* untuk 3D graphics
10. Media support untuk MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF

11. Support pada GSM Telephony fasilitas telepon.
12. Mendukung menggunakan jaringan Bluetooth, EDGE, 3G, dan WiFi
13. Mendukung penggunaan *device* Kamera, GPS, *compass* dan *accelerometer*
14. Mendukung Multitouch

Sumber : Widiyanto Pratama, Tutorial Android Programming, 2013