

Analisis Pembuatan Virus Multiaction Dan Antivirus Menggunakan Metode CRC64

Dimas Nugrah W. U

Fakultas Ilmu Komputer Universitas AKI

Abstract

Computer users today are still familiar with the problems caused by computer viruses. This problem arises because the emergence of new variants are always adapted to the existing anti-virus applications. In addition the system security holes that can be utilized by the virus are always found in the virus makers. Based on these problems, the writer built a viral application showing the evidence that the existence of security holes in computer operating systems can be utilized by the virus. The writer also built an antivirus application that can handle viruses which are able to either adapt or close the gap that can not be solved by other antivirus applications, where the gap can still be utilized by the virus in a computer attack victim.

The virus which is built using waterfall process model is a viral application that has several capabilities including the ability to hide, to multiply, to find, to check the target of attacks, and to manipulate the system. The application is called the virus ISO that is built using language programming Delphi 7, and antivirus applications are built using the CRC64 method as a way for naming a file and also for the introduction of a file identified as potential viruses. Antivirus application is called AISO that is built using language programming Visual Basic 6. Virus applications are built to find out the gap potentially used by the viruses that can damage the security of computer systems, whereas the antivirus application is built to cover the gap of the computer system as well as the gaps that have not been addressed by other anti-virus, so the virus can not enter and damage the computer system.

Key words: Virus, Antivirus, ISO, AISO, file, waterfall, CRC64

Pendahuluan

Virus komputer merupakan program komputer yang dapat menggandakan atau menyalin dirinya sendiri dan menyebar dengan cara menyisipkan salinan dirinya ke

dalam program atau dokumen lain [1].

Virus komputer dapat dianalogikan dengan virus biologis yang menyebar dengan cara menyisipkan dirinya sendiri ke sel makhluk hidup. Virus komputer sifatnya

dapat merusak misalnya dengan merusak data pada dokumen, membuat pengguna komputer merasa terganggu dengan keberadaannya dalam sebuah sistem komputer, maupun tidak menimbulkan efek merusak sama sekali [6].

Dengan kemampuan yang dimiliki sebuah virus tersebut, terkadang membuat seorang pengguna komputer awam menjadi panik. Bahkan disebabkan oleh ulah sebuah virus tersebut, seorang pengguna komputer dengan mudahnya melakukan format ulang terhadap sistem operasi yang digunakannya dengan harapan virus tersebut hilang. Namun hal ini juga menyebabkan beberapa data maupun program yang sudah terinstal juga ikut hilang.

Berdasarkan observasi yang dilakukan penulis mengenai sebuah aplikasi virus dan antivirus, kinerja dari kedua aplikasi tersebut memiliki sifat dan perilaku yang sama diantara berbagai macam variannya. Dengan berkembangnya teknologi informasi, maka pembuatan aplikasi virus maupun aplikasi antivirus semakin mudah, sehingga diharapkan dengan mengetahui sifat dan perilaku keduanya, aplikasi virus dan antivirus yang akan dibangun dapat semakin memberikan

penjelasan yang baik mengenai cara kerja kedua aplikasi tersebut.

1.1 Rumusan Masalah

- 1 Proses pembuatan sebuah virus yang dapat menyebar dari satu komputer ke komputer lain dengan memanfaatkan *mobile device* seperti Flash Disk dan disket.
- 2 Pemanfaatan kemampuan potensial dari Sistem Operasi *Microsoft Windows XP* sebagai suatu alat yang dapat mengaktifkan virus bekerja secara otomatis.
- 3 Penyebaran virus memanfaatkan kemampuan dari virus itu sendiri maupun kesalahan atau ketidaktahuan yang dilakukan oleh pengguna komputer dengan mengaktifkan virus tersebut.
- 4 Proses pembuatan sebuah virus yang mampu mempertahankan diri dengan bersembunyi dan dapat menyerang fungsi kerja dari sistem maupun data pada komputer korban.
- 5 Proses pembuatan antivirus sederhana dengan memanfaatkan sumber-sumber potensial dari sistem komputer itu sendiri, dan juga mampu mengenali sebuah virus berdasarkan identitasnya

yang telah tercatat sebelumnya dalam basis data antivirus tersebut.

1.2 Batasan Masalah

Batasan masalah dalam penulisan tentang analisis dan implemementasi pembuatan virus *multiaction* dan antivirus menggunakan metode CRC64, akan dibatasi pada :

- 1) Proses pembuatan virus dan antivirus, yang dibatasi hingga proses pengujian.
- 2) Implementasi virus dan antivirus hanya dapat dilakukan pada semua komputer yang menggunakan Sistem Operasi *Microsoft Windows XP*.
- 3) Virus tidak dapat bekerja atau aktif pada saat *firewall* komputer yang akan diinfeksi dalam keadaan aktif.
- 4) Penyebaran virus memanfaatkan media *mobile device* seperti *flash disk*, *disket*, dan CD/DVD alih-alih menyebar pada jaringan baik lokal maupun Internet.

1. Tinjauan Pustaka

Siklus Hidup Pengembangan Aplikasi

Dalam penyusunan dan pembuatan virus dan antivirus ini digunakan model sekuensial linier, model sekuensial linier mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang

sistematik dan sekuensial yang diawali pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan. Model sekuensial linier sering disebut juga dengan "siklus kehidupan klasik" atau "model air terjun" seperti layaknya air terjun.

1. Tahap Analisis

Analisis merupakan proses pengumpulan kebutuhan pada perangkat lunak (*software*). Untuk memahami sifat program yang dibangun, perekraya perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk kerja, dan antarmuka (*interface*) yang diperlukan. Kebutuhan baik untuk sistem maupun untuk perangkat lunak didokumentasikan dan dilihat kembali oleh pengguna (*user*). Tujuan dari tahap analisis ini adalah menjabarkan kebutuhan pengguna, serta meletakkan dasar-dasar untuk proses perancangan perangkat lunak.

2. Tahap Desain

Desain perangkat lunak sebenarnya adalah proses multilangkah yang berfokus pada empat atribut sebuah program yang berbeda yaitu, struktur data, arsitektur perangkat lunak, representasi antarmuka (*interface*), dan alur (*algoritma*) prosedural.

Proses desain menerjemahkan syarat atau kebutuhan ke dalam sebuah representasi perangkat lunak yang dapat diperkirakan demi kualitas perangkat lunak sebelum dimulai pemunculan kode. Sebagaimana persyaratan, desain didokumentasikan dan menjadi bagian dari konfigurasi perangkat lunak.

3. Tahap Kode

Kode atau kode merupakan penerjemahan dari proses pada tahap desain ke dalam bentuk bahasa pemrograman yang dapat dibaca oleh mesin. Penulisan kode diharapkan menghasilkan suatu fungsionalitas yang mengacu pada tahap perancangan.

4. Pengujian Perangkat Lunak

Pengujian adalah proses mengeksekusi program atau sistem secara keseluruhan untuk menemukan kesalahan-kesalahan. Tahap ini akan terus berulang sampai dengan sistem yang dikembangkan sesuai dengan *requirements*. Tahap tes berfokus pada logika internal perangkat lunak, yang memastikan bahwa semua pernyataan telah diuji, dan pada eksternal fungsional yaitu mengarahkan pengujian untuk menemukan kesalahan-

kesalahan dan memastikan bahwa *input* yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

Cara Kerja Aplikasi Virus dan Antivirus

Deskripsi aplikasi virus dan antivirus merupakan gambaran dari aplikasi- aplikasi tersebut. Berikut ini merupakan deskripsi dari aplikasi virus dan antivirus.

1) Cara Kerja Virus

Setiap virus komputer yang aktif pada dasarnya memiliki sifat-sifat dasar, pada tugas akhir ini virus yang dibuat akan memanfaatkan kemampuan-kemampuan dasar tersebut. Beberapa *routine* yang merupakan kemampuan dasar sebuah virus, yaitu: *search routine*, *copy routine*, *stealth routine*, *checking routine*, dan *manipulation routine*.

2) Cara Kerja Antivirus

Setiap antivirus komputer yang aktif pada dasarnya memiliki sifat-sifat dasar, pada tugas akhir ini antivirus yang dibuat akan memanfaatkan kemampuan-kemampuan dasar tersebut. Cara kerja program-program penghapus virus ini kebalikan dari program virus itu sendiri, yaitu mendeteksi apakah suatu *file* terkena virus atau tidak, jika terkena, maka antivirus akan

menindaklanjuti *file* tersebut yaitu dengan mengkarantina, menghapus, atau memperbaiki. Seperti program lainnya, antivirus mempunyai struktur tersendiri. Kode antivirus tersebut wajib dimiliki oleh suatu program antivirus. Komponen dasar yang harus dimiliki oleh sebuah program antivirus, yaitu: *application code*, *method code*, *virus list code*, *manipulation file code*, *kill process code*, dan *recovery code*.

Visual Basic 6.0

Mengenal MS-Visual Basic 6.0

MS-Visual Basic 6.0 adalah bahasa pemrograman yang bekerja dalam lingkup MS-Windows. MS-Visual Basic 6.0 dapat memanfaatkan kemampuan MS-Windows secara optimal. Kemampuannya dapat dipakai untuk merancang program aplikasi yang berpanampilan seperti program aplikasi lainnya berbasis MS-Windows. (Menjadi mahir tanpa guru Microsoft Visual Basic Versi 6.0. 2001). Kelebihan dari program ini adalah kemudahan dalam praktek dan struktur sintaknya yang mudah dimengerti karena bahasa program yang digunakan seperti bahasa Inggris sehari-hari. Disamping itu, jika terjadi kesalahan akan ditunjukkan jenis kesalahannya sekaligus letak kesalahannya.

Langkah Umum Membuat Program Aplikasi

Langkah-langkah umum untuk membuat program aplikasi adalah sebagai berikut :

1. Gambar komponen dan tata letak ke dalam jendela **Form** menggunakan piranti dalam jendela **Toolbox**.
2. Tentukan properti pada tiap komponen menggunakan jendela **Properties**.
3. Tulis kode program untuk event (suatu kejadian yang dirasakan oleh objek, misal tunjuk, klik, atau kejadian lainnya) yang bersangkutan pada tiap komponen.

Beberapa Komponen MS-Visual Basic 6.0 Project

Project adalah sekumpulan modul. Jadi project adalah program aplikasi itu sendiri. Project disimpan dalam akhiran **.VBP**. File ini menyimpan seluruh komponen program, termasuk pilihan proyek, pilihan **Environment**, pilihan file **.EXE**, dan segala sesuatu yang berhubungan dengan project. Pada jendela project berisi semua file yang dibutuhkan untuk menjalankan program aplikasi yang dibuat. Pada jendela project terdapat tiga icon, yaitu icon **View Code**, **View Object** dan **Toggle Folders**. Icon **View Code**

dipakai untuk menampilkan jendela editor kode program. Icon **View Object** dipakai untuk menampilkan bentuk formulir (*Form*) dan icon **Toggle Folders** berguna untuk menampilkan folder (tempat penyimpanan file).

Form

Form adalah suatu objek yang dipakai sebagai tempat bekerja program aplikasi. Form berbentuk jendela dan dapat dibayangkan sebagai kertas atau meja kerjaya yang dapat dilukisi atau diletakkan ke dalamnya objek-objek lain. Biasanya pada saat mendesain form, terdapat garis titik-titik yang disebut **grid**. **Grid** sangat membantu dalam pengaturan tata letak objek yang dimasukkan dalam form, karena gerakan penunjuk mouse akan sesuai (tepat) pada titik grid.

Toolbox

Toolbox adalah kotak alat-alat yang berisi icon-icon untuk memasukkan objek-objek tertentu ke dalam jendela form. MS-Visual Basic 6.0 menyediakan beberapa kategori aplikasi,. Misalnya **VB Enterprise Edition, Standart EXE, ActiveX EXE, ActiveX DLL, Data Project** dan lain-lain. Toolbox dapat dimodifikasi dengan

menambah komponen icon-icon dengan cara mengklik kanan pada toolbox, lalu pilih **Componen** atau **Add Tab**.

Properties

Properties digunakan untuk menentukan setting suatu objek. Suatu objek biasanya mempunyai beberapa properti yang dapat diatur langsung dari jendela properties atau langsung melalui kode program. Setting properti akan menentukan cara kerja dari objek yang bersangkutan saat program aplikasi dijalankan, misalnya menentukan warna objek, bingkai objek, pengambilan data dan lain-lain.

Kode Program

Kode program adalah serangkaian tulisan perintah yang akan dilaksanakan jika suatu objek dijalankan. Kode program ini akan mengontrol dan menentukan jalannya suatu objek.

Event

Event adalah peristiwa atau kejadian yang diterima oleh suatu objek, misalnya, klik, seret, tunjuk dan lain-lain. Event yang diterima objek akan memicu MS-Visual Basic 6.0 menjalankan kode program yang

ada didalamnya. Sebagai contoh dapat dilihat pada pernyataan,

```
Private Sub cmdubah_Click()
```

Baris diatas menunjukkan penggunaan event **Click** pada objek **cmdubah**, yang berarti jika kita mengklik **cmdubah** maka baris-baris perintah dibawahnya akan dilaksanakan sampai menjumpai baris perintah **End Sub**. Perhatikan juga pernyataan berikut,

```
Private Sub Form_Load()
```

Baris diatas menunjukkan penggunaan event **Load** pada objek form, yang berarti jika pada saat form diload (dijalankan) program dibawahnya akan dilaksanakan sampai menjumpai baris perintah **End Sub**.

Metoda (Method)

Metode adalah set perintah seperti halnya fungsi dan procedure, tetapi sudah tersedia dalam suatu objek. Seperti halnya dengan properti (yang juga terdapat dalam suatu objek), suatu metoda dapat dipanggil dengan menyebut nama objek diikuti dengan tanda titik dan nama metodenya metoda

biasanya akan mengerjakan suatu tugas khusus pada suatu objek tertentu, sedangkan properti biasanya memberi definisi nilai atau setting pada objek. Antara objek satu dengan yang lainnya mempunyai metoda yang sama atau tidak sama, baik dalam segi jumlah metoda atau macam metoda.

Module

Module dapat disejajarkan dengan form, tetapi tidak mengandung objek dan bentuk standar. Module dapat berisi beberapa kode program atau prosedur yang dapat digunakan dalam program aplikasi. Sebenarnya suatu form juga adalah sebuah module, tetapi mempunyai objek berupa form. Selain itu ada juga **Class module**, yaitu suatu module berupa objek class, yaitu suatu objek yang mempunyai metoda dan properti yang sudah terdefinisi.

Tipe Data, Variabel, dan Operator

Tipe Data

MS-Visual Basic 6.0 menyediakan beberapa tipe data seperti pada table berikut ini,

Tabel 1. Tipe Data

Tipe	Ukuran Byte	Tipe	Ukuran Byte
Integer	2 byte	Byte	1 byte
Long	4 byte	Boolean	2 byte
Single	4 byte	Date	8 byte
Double	8 byte	Object	4 byte
Currency	8 byte	Variant	16 byte + 1 byte ekarakter
String	1 byte per karakter		

Pemilihan tipe data yang digunakan secara tepat akan sangat berguna dalam hal menghemat memori, kecepatan proses, ketelitian perhitungan dan lain-lain.

Variabel

Variabel adalah suatu tempat dalam memori komputer yang diberi nama dan dialokasikan untuk menampung data. Sesuai dengan data yang ditampung, variabel harus mempunyai tipe data yang sesuai dengan isinya.

Cara pertama adalah dengan menggunakan perintah **DIM** untuk mendeklarasikan nama variable beserta tipe datanya pada awal procedure. Pernyataan deklarasi diatas disebut deklarasi eksplisit dan menyebabkan MS-Visual Basic 6.0 mengalokasikan memori sesuai dengan nama dan tipe yang diinginkan, sehingga

terkontrol variabel dan tipe variabel yang digunakan.

Cara kedua disebut dengan deklarasi, deklarasi implisit ini membuat MS-Visual Basic 6.0 mengalokasikan nama variabel, berikut tipenya sekaligus mengisi nilai variabelnya.

Cara yang ketiga yaitu mendeklarasikan variabel yang dikenali semua objek, yaitu dengan menggunakan perintah *public*.

Operator

Untuk melakukan beberapa manipulasi dan pengolahan data. MS-Visual Basic 6.0 menyediakan bermacam-macam operator, seperti

1. Operator penugasan

Operator ini disimbolkan dengan tanda (=) dan berfungsi untuk memasukkan

suatu data kedalam suatu variabel, penulisannya adalah:

$$\langle \text{Variabel} \rangle = \langle \text{Ekspresi} \rangle$$

2. Operator Aritmatika

Digunakan untuk melakukan operasi matematika. Operator aritmatika mempunyai hirarki paling tinggi dibandingkan dengan operator lainnya, yaitu operator pembanding

dan logika. Jadi bila dalam satu baris terdapat operator aritmatika, operator perbandingan dan operator logika, maka yang didahulukan adalah operator aritmatika dilanjutkan dengan operator pembanding dan operator logika. Operator aritmatika dengan hierarki mulai dari paling tinggi ke paling rendah ditunjukkan pada tabel berikut ini,

Tabel 2. Operasi Aritmatika

Operator	Operasi
^	Pemangkatan
-	Tanda negatif
*, /	Perkalian dan Pembagian
\	Pembagian Integer
Mod	Modulus (Sisa Pembagian)
+, -	Penambahan dan Pengurangan
+, &	Penggabungan String

3. Operator Pembanding

Operator pembanding digunakan untuk membandingkan suatu data dengan data lain dan menghasilkan nilai logika (boolean) benar atau salah. Dua data yang

dibandingkan harusnya mempunyai tipe data yang sama. Adapun operator pembanding yang disediakan MS-Visual Basic 6.0 adalah seperti pada tabel berikut ini.

Tabel 3. Operator Pembanding

Operator	Keterangan
=	Sama dengan

<>	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil atau Sama dengan
>=	Lebih besar atau Sama dengan
Like	Mempunyai ciri yang sesuai
Is	Sama Referensi objek

4. Operator Logika

Digunakan untuk mengekspresikan satu atau lebih data logika menghasilkan

data logika baru. Tabel logika dengan hierarki dari atas ke bawah adalah sebagai berikut,

Tabel 4. Operator Logika

Operator	Keterangan
Not	Tidak
And	Dan
Or	Atau
Xor	Exclusive Or
Eqv	Ekivalen
Imp	Implikasi

5. Pengulangan Do...Loop

Untuk mengulang suatu blok pernyataan selama (while) suatu kondisi (syarat) bernilai true atau sampai (until) suatu kondisi menjadi true. Penulisannya adalah sebagai berikut:

Do[{While / Until} <kondisi>]

[satu atau lebih pernyataan program]

[Exit Do]

[satu atau lebih pernyataan program]

Loop

Bentuk penulisan lain adalah sebagai berikut,

Do

```
[satu atau lebih pernyataan program]
[Exit Do]
[satu atau lebih pernyataan program]
    Loop
[While / Until]<kondisi>]
```

6. Pengulangan For...Next

Untuk melakukan pengulangan (iterasi) suatu blok program beberapa kali yang ditentukan dari nilai awal dan nilai akhir. Penulisannya adalah sebagai berikut,

```
For <counter> = <awal> to <akhir>
[step <step>]
[satu atau lebih pernyataan program]
[Exit For]
[satu atau lebih pernyataan program]
    Next <counter>
```

Counter adalah variabel tipe data numerik dan tidak boleh berupa element dari suatu array. Counter ini pertama kali akan diisi dengan data numerik <awal> dan setelah mengerjakan badan Loop, maka akan otomatis bertambah sebanyak tipe data numerik <step>, proses pengulangan akan dilakukan selama counter lebih kecil atau sama dengan <akhir>.

7. Percabangan bersyarat If...Then...Else

Digunakan untuk menjalankan satu blok perintah, bergantung pada nilai yang diuji, penulisannya adalah sebagai berikut,

```
If <kondisi> Then
[satu atau lebih pernyataan program]
[Elseif <kondisi lain> Then
[satu atau lebih pernyataan program]
[Else
[satu atau lebih pernyataan program]
    End If
```

<Kondisi> adalah suatu ekspresi logika. Perintah If yang pertama akan menguji <kondisi> disebelah kanannya, jika benar maka blok perintah yang berada dibawahnya akan dikerjakan sampai menemui perintah ElseIf atau Else lalu langsung melompat ke pernyataan End If, jika kondisi pertama salah dan dijumpai perintah ElseIf, maka kondisi disebelah kanan ElseIf akan diuji, jika benar blok perintah dibawahnya akan dilaksanakan. Jika salah maka program akan mengerjakan blok perintah dibawah Else atau ke End If.

Operasi Database

Database dapat diartikan sebagai kumpulan data yang terdiri dari satu atau lebih tabel yang terintegrasi satu dengan yang lainnya dimana setiap pemakai diberi wewenang untuk dapat mengakses

(mengubah, menghapus, menganalisis, menambah, memperbaiki, menghapus) data dalam tabel-tabel tersebut. (Membangun Database dengan VISUAL BASIC 6.0 dan Perintah SQL, 2002)

2. Pembahasan

Aplikasi virus merupakan sebuah aplikasi yang tidak membutuhkan interaksi secara langsung dengan pengguna komputer. Pada aplikasi virus yang akan dibangun adalah sebuah aplikasi yang melakukan mekanisme kerja secara otomatis antara aplikasi dengan sistem yang menjadi korban. Pada mekanisme interaksi virus dengan sistem merupakan kerja sifat dan perilaku dasar dari virus, diantaranya yaitu *search routine*, *copy routine*, *stealth routine*, *checking routine*, dan *manipulation routine*.

Kelima perilaku tersebut akan bekerja pada sistem komputer korban yang akan diinfeksi. Berdasarkan kelengkapan fungsi dasar pada sebuah virus, maka virus yang akan dibangun dinamakan "ISO". Sedangkan pada aplikasi antivirus merupakan sebuah aplikasi yang berfungsi sebagai penghenti kerja dari sebuah aplikasi virus, dengan kata lain aplikasi antivirus merupakan lawan dari aplikasi virus. Pada

aplikasi antivirus yang dibangun juga akan bekerja sifat dan perilaku yang mirip dengan apa yang dilakukan oleh sebuah virus. Mekanisme kerja antivirus membutuhkan peran dari sebuah pengguna komputer sebagai sebuah objek yang menjalankan aplikasi tersebut yaitu *update database* dan proses pendeteksian virus. Untuk interaksi antara aplikasi dengan sebuah objek yang akan diperiksa berlaku mekanisme *manipulation file code*, *kill process code*, dan *recovery code*. Berdasarkan kelengkapan fungsi dasar yang dimiliki pada aplikasi antivirus yang hendak dibangun, maka aplikasi antivirus dinamakan "AISO".

Aplikasi virus dan antivirus yang akan dibangun serta dijadikan sebagai bahan uji dengan tujuan untuk memberikan pengetahuan para pengguna komputer mengenai sifat dan perilaku umum sebuah virus, memberikan kewaspadaan terhadap perangkat keras yang tersambung pada komputernya, serta memberikan pengetahuan mengenai mekanisme pembuatan program virus dan antivirus.

Tabel 5. Spesifikasi Kebutuhan Perangkat Lunak Virus

SRS ID	Deskripsi
SRS – VIR – F01	Mengaktifkan diri sendiri
SRS – VIR – F02	Melakukan penggandaan diri
SRS – VIR – F03	Melakukan penyembunyian diri baik dari pengguna komputer maupun dari sistem operasi
SRS – VIR – F04	Melakukan manipulasi pada kinerja sistem
SRS – VIR – F05	Melakukan pencarian dan pemeriksaan terhadap <i>file</i> maupun lokasi yang akan dijadikan target penyerangan
SRS – VIR – F06	Melakukan <i>blocking</i> terhadap antivirus yang dikenali, baik yang sudah terpasang maupun yang baru akan dipasang

Tabel 6. Spesifikasi Kebutuhan Perangkat Lunak Antivirus

SRS ID	Desk
SRS – ATV – F01	Melakukan pencarian terhadap <i>file</i> maupun lokasi induk virus
SRS – ATV – F02	Melakukan pemeriksaan terhadap suatu <i>file</i> maupun lokasi
SRS – ATV – F03	Menghentikan proses kerja dari virus
SRS – ATV – F04	Menangkap atau menghancurkan aplikasi induk virus
SRS – ATV – F05	Melakukan manipulasi perbaikan pada kinerja sistem
SRS – ATV – F06	Melakukan pengecekan versi <i>database</i> virus

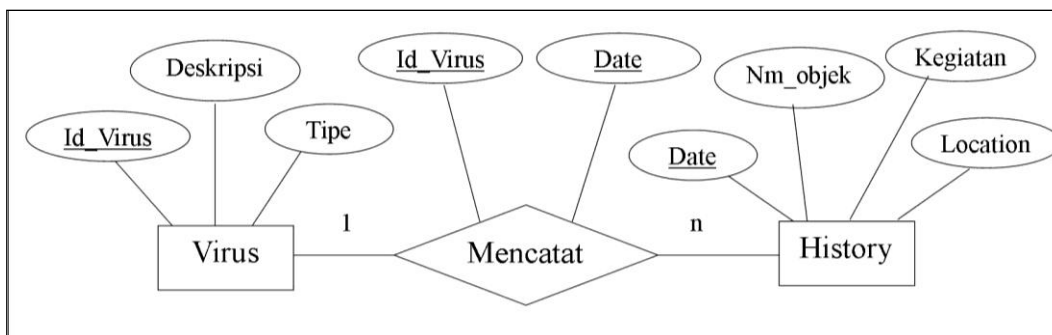
Entity Relationship Diagram (ERD)

Pada aplikasi virus tidak memiliki ER Diagram. Hal ini dikarenakan pada aplikasi virus dalam proses pembuatannya

tidak memiliki kebutuhan data. Sedangkan untuk aplikasi antivirus memiliki kebutuhan data, dan data yang dibutuhkan oleh aplikasi antivirus ini memiliki dua

entitas data, yang memiliki relasi *one to many* atau satu ke banyak. Pada ER diagram yang terdapat pada gambar 3.1. relasi yaitu MENCATAT.

ER Diagram, terdapat dua entitas yaitu VIRUS dan HISTORY, serta satu



Gambar 3.1 Entity Relationship Diagram (ERD)

Tabel 7. Keruntutan kebutuhan dan rancangan fungsi ISO

No	SRS	Deskripsi SRS	Nomor Fungsi	Nama Fungsi
1	SRS – VIR – F01	Digunakan untuk melakukan penggandaan diri	2	<i>Copy Routine</i>
2	SRS – VIR – F02	Digunakan untuk melakukan penyembunyian diri baik dari pengguna komputer maupun dari sistem operasi	1	<i>Stealth Routine</i>
3	SRS – VIR – F03	Digunakan untuk melakukan pencarian dan pemeriksaan terhadap <i>file</i> maupun lokasi yang akan dijadikan target penyerangan	3	<i>Searching Routine</i>
4	SRS – VIR – F04	Digunakan untuk mengecek keberadaan dan mengaktifkan diri sendiri	4	<i>Checking Routine</i>
5	SRS – VIR – F05	Digunakan untuk melakukan manipulasi pada kinerja sistem	5	<i>Manipulation Routine</i>
6	SRS – VIR – F06	Digunakan untuk melakukan <i>blocking</i> terhadap antivirus yang dikenali, baik yang sudah terpasang maupun yang baru akan dipasang		

7	SRS – VIR – F07	Digunakan untuk melakukan <i>flooding</i> pada sistem dengan mengaktifkan beberapa program secara otomatis terus-menerus pada waktu tertentu		
8	SRS – VIR – F08	Digunakan untuk melakukan <i>blocking</i> terhadap beberapa kinerja sistem		

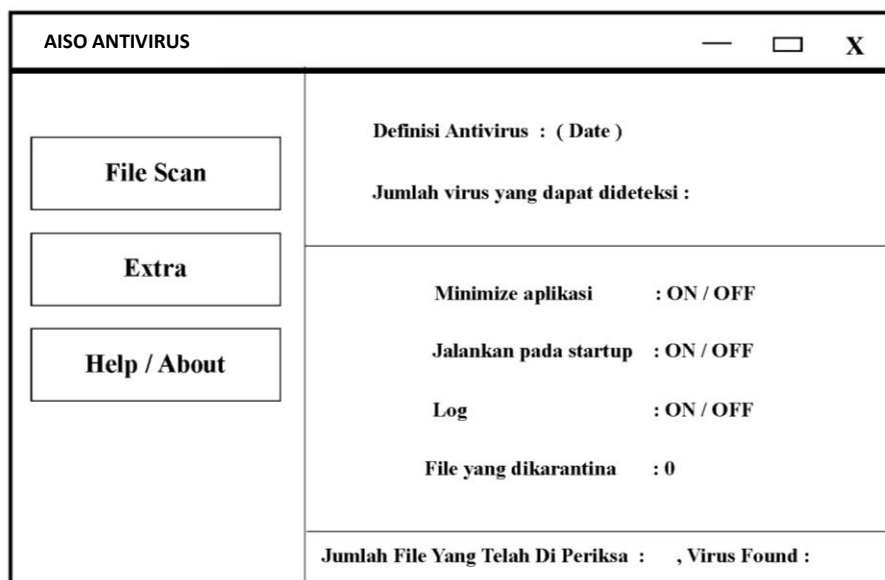
Tabel 8. Kerunutan kebutuhan dan rancangan fungsi AISO

No	SRS ID	Deskripsi	Nomor Fungsi	Nama Fungsi
1	SRS – ATV – F01	Digunakan untuk melakukan pengecekan versi database virus	1	Cek Database Virus
2	SRS – ATV – F02	Pencatatan <i>history scanning</i> virus pada aplikasi ini		
2	SRS – ATV – F03	Digunakan untuk melakukan <i>update database</i> antivirus	2	Update Database Virus
2	SRS – ATV – F04	Pencatatan <i>history upadet</i> antivirus.		
3	SRS – ATV – F05	Digunakan untuk melakukan pencarian terhadap <i>file</i> maupun lokasi induk virus	3	Deteksi File
4	SRS – ATV – F06	Digunakan untuk melakukan pemeriksaan terhadap suatu <i>file</i> maupun lokasi		
5	SRS – ATV – F07	Digunakan untuk menghentikan proses kerja dari virus	4	Kill Process Code
6	SRS – ATV – F08	Digunakan untuk menangkap atau menghancurkan aplikasi induk virus	5	Manipulation File Code
7	SRS – ATV – F09	Digunakan untuk melakukan manipulasi perbaikan pada kinerja sistem	6	Recovery Code

Rancangan Antarmuka

Antarmuka perangkat lunak yang digunakan adalah antarmuka berbasis GUI, dengan pemrograman visual dan menggunakan Sistem Operasi *Microsoft*

Windows XP. Pada aplikasi virus tidak memiliki rancangan antarmuka. Sedangkan untuk rancangan antarmuka aplikasi antivirus untuk AISO dapat dilihat pada gambar 3.2, Menu Utama AISO.



Gambar 3.2 Menu utama AISO

3. Implementasi dan Pengujian

Implementasi Rancangan Data

Implementasi rancangan data merupakan transformasi rancangan data

yang dihasilkan dari proses perancangan data menjadi suatu *database*. *Database* yang dibangun memiliki satu tabel yang melekat di dalamnya yaitu tabel VIRUS.

Tabel 9. Daftar Tabel Aplikasi AISO

Nama Tabel	Field	Deskripsi isi
VIRUS	<u>Id_Virus</u>	ID dari virus
	Deskripsi	Deskripsi virus
	Tipe	Tipe file yang terdeteksi
HISTORY	Date	Tanggal dan waktu pencatatan <i>history</i>
	Kegiatan	Kegiatan yang dilakukan dan dicatat di <i>history</i>
	Nm_objek	Nama objek yang tercatat di <i>history</i>
	Location	Lokasi induk virus yang terdeteksi

Pengujian Perangkat Lunak

Pengujian adalah proses mengeksekusi program atau sistem secara keseluruhan

untuk menemukan kesalahan-kesalahan. Tahap ini akan terus berulang sampai dengan sistem yang dikembangkan sesuai

dengan *requirements*. Pada pengujian aplikasi ini digunakan teknik *black-box*. *Black-box* adalah pengujian yang berfokus pada persyaratan fungsional perangkat lunak.

Kesimpulan

Kesimpulan yang dapat diambil adalah memberikan pengetahuan para pengguna komputer mengenai sifat dan perilaku umum sebuah virus. Membuat para pengguna komputer agar waspada terhadap perangkat keras yang tersambung pada komputernya. Aplikasi virus dapat memberikan pengetahuan mengenai mekanisme pembuatan program virus dan antivirus. Aplikasi antivirus dan virus dapat memberikan pengetahuan pengguna komputer mengenai pengaturan dan pemanfaatan akses sistem register pada Sistem Operasi *Microsoft Windows XP* yang potensial untuk digunakan para pembuatan virus dalam mendukung virus yang diciptakan agar dapat bekerja dengan baik. Antivirus ini bermanfaat untuk memberikan keamanan yang lebih bagi sistem komputer yang digunakan. Virus berguna untuk mencari celah pada sistem komputer yang berpotensi untuk dimanfaatkan oleh virus. Antivirus berguna untuk menutup celah pada sistem

yang belum tertangani oleh sistem komputer itu sendiri maupun antivirus lain.

Daftar Pustaka

- Aat Shadewa, 2006. *Rahasia Membuat Antivirus Menggunakan Visual Basic*. Yogyakarta : DSI Publishing.
- Hartini. 2006. *Analisis Dengan Diagram Aliran Data (DFD)*. Materi Kuliah, Ilmu Komputer, Universitas Sriwijaya.
- Leo Hendrawan. 2004. *Virus Komputer : Sejarah Dan Perkembangannya*.
- Marko Helenius. 2002. *A System to Support the Analysis of Antivirus Products' Virus Detection Capabilities*.
- Oviliani Yenty Yuliana. 2001. *Implementasi Referential Integrity Constraint Pada Microsoft Access Dalam Upaya Memelihara Konsistensi Data*. Jurnal Informatika, Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra.
- Pressman, Roger S. 1997. *Software Engineering : A Practitioner's Approach*. New York : McGraw – Hill.
- Virus Tutorial. 2006. *Mengenal Virus Lewat Checksum Error dengan metode CRC64*. [http://virologi.info/virologist/modul es/news/article.php?storyid=6](http://virologi.info/virologist/modul_es/news/article.php?storyid=6) : 20 April 2006.
- Wikipedia bahasa Indonesia, ensiklopedia bebas. 2007. *Perangkat Lunak Antivirus*. <http://id.wikipedia.org/wiki/Perangka>

- t_lunak_antivirus : 23 Oktober 2007.
- Wikipedia bahasa Indonesia, ensiklopedia bebas. 2007. *Virus Komputer*. http://www.id.wikipedia.org/wiki/Virus_komputer : 21 Oktober 2007.
- Wikipedia bahasa Indonesia, ensiklopedia bebas. 2009. *Teknologi Informasi*. http://id.wikipedia.org/wiki/Teknologi_informasi : 15 Juni 2009
- Pressman, Roger S. 1997. *Software Engineering: a Practitioner's Approach*, Fifth Edition (The McGraw-Hill Companies, New York)
- Wapedia-Wiki : Struktur Data, 2009. http://wapedia.mobi/id/struktur_data : 24 Nopember 2009