B a b 7

Keamanan Sistem dan Proteksi

Tujuan Pelajaran

Setelah mempelajari bab ini, Anda diharapkan :

- Memahami mekanisme proteksi yang diperlukan dalam suatu system computer dari segala macam ancaman
- Menganalisa masalah sekuriti system computer

7.1. Pendahuluan

Keamanan komputer adalah suatu perlindungan yang diusahakan oleh suatu system informasi dalam rangka mencapai sasaran hasil yang bisa diterapkan atau cara untuk memelihara integritas, kerahasiaan dan tersedianya informasi. Tetapi pada saat ini sistem komputer yang terpasang makin mudah diakses. Sistem *time sharing* dan akses jarak jauh menyebabkan kelemahan komunikasi data menjadi pokok masalah keamanan. Kelemahan ini menjadi amat serius dnegan meningkatknya perkembangan jaringan komputer.

Kecenderungan lain saat ini adalahmemberi tanggung jawab pengelolaan aktivitas pribadi dan bisnis ke komputer. Komputer telah rutin dipakai untuk korespondensi yang sanga sensitif, seperti :

- sistem transfer dana elektronis (*electronic fund transfer system*) : melewatkan sejumlah uang sebagai aliran bit
- sistem kendali lalu lintas udara (*air traffic control system)* : melakukan banyak kerja yang sebelumnya ditangani pengendali manusia.
- Unit rawat intensif di rumah sakit sudah sangat terkomputerisasi.

Saat ini, implementasi pengamanan sangat penting untuk enjamin sistem tidak diinterupsi dan diganggu. Proteksi dan pengaman terhadap perangkat keras dan sistem operasi sama pentingnya.

Sistem operasi hanya satu porsi kecil dari seluruh perangkat lunak di suatu sistem. Tetapi karena peran sistem operasi mengendalikan pengaksesan ke sumber daya, dimana perangkat lunak lain pengaksesan sumber daya lewat sistem operasi, maka sistem operasi menempati posisi yang penting dalam pengamanan sistem. Pengamanan perangkat lunak cenderung memfokuskan pada pengamanan sistem operasi. Perlu diingat bahwa perangkat lunak aplikasi juga memberi resiko keamanan.

Keamanan sistem operasi merupakan bagian dari masalah sistem komputer secara total, tapi telah menjadi bagian yang meningkat kepentingannya. Pengamanan sistem operasi berarti kecil jika setipa orang dapat berjalan melenggang di ruang sistem komputer. Pengamanan secara fisik dengan membatasi pengaksesan fisik secara langsung dengan fasilitas sistem komputer harus dilakukan juga.

7.2. Keamanan

Pengamanan sistem komputer bertujuan untuk menjamin sumber daya tidak digunakan atau dimodifikasi oleh orang tak berhak. Pengamanan termasuk masalah teknis, manajerial, legalitas dan politis.

Terdapat empat macama kejahatan komputer, antara lain :

1. Pencurian waktu komputer. Ini meliputi waktu yang diperlukan memperbaiki sistem komputer setelah terkena virus.
2. Pencurian data
3. Manipulasi program komputer
4. Pencurian software maupun pengkopian software

Keamanan sistem terbagi menjadi tiga, yaitu :

1. Keamanan eksternal
   Keamanan eksternal berkaitan dengan pengamanan fasilitas komputer dari penyusup dan bencana seperti kebakaran atau kebanjiran.
2. Keamanan interface pemakai
   Keamanan interface pemakai berkaitan dengan identifikasi pemakai sebelum pemakai diijinkan mengakses program dan data yang disimpan.
3. Keamanan internal

Keamanan internal berkaitan dengan pengamanan beragam kendali yang dibangun pada perangkat keras dan sistem operasi yang menjamin operasi yang lhandal dan tak terkorupsi untuk menjaga integritas program dan data.

Masalah-masalah Keamanan

Pada keamanan, terdapat dua masalah penting, yaitu :
1. Kehilangan
2. Penyusup

Kehilangan data dapat disebabkan, antara lain :
a. Bencana
   - kebakaran
   - banjir
   - gempa bumi
   - perang
   - kerusuhan
   - gerogotan tikus pada pita rekaman data atau floopy disk
b. Kesalahan perangkat keras dan perangkat lunak
   - ketidak berfungsian pemroses
   - disk atau tape yang tidak terbaca
   - kesalahan program (bugs)
   - keandalan perangkat keras dapat dilakukan dengan pencegahan dan perawatan rutin
   - keandalan perangkat lunak dilakukan dengan testing dan debugging
c. Kesalahan manusia
   - kesalahan pemasukan data
   - memasang tape atau disk yang salah
   - eksekusi program yang salah
   - kehilangan disk atau tape

Kehilangan data dapat diatasi dengan mengelola beberapa *backup* dan *backup* ditempatkan dari data yang *online*.

Penyusup, terdiri dari :
1. Penyusup pasif, yaitu yang memabca data yang tak diotorisasi
2. Penyusup aktif, yaitu mengubah data yang tak diotorisasi

Kategori penyusupan
1. Lirikan mata pemakai *non-teknis*. Pada sistem *time-sharing* kerja pemakai dapat diamati orang sekelilingnya. Bila dengan lirikan mata itu dapat mengetahui apa yang diketik pengisian *password*, maka pemakaian non teknis dapat mengakses fasilitas yang bukan haknya
2. Penyadapan oleh orang dalam
3. Usaha *hacker* dalam mencari uang
4. Spionase militer atau bisnis

Ancaman-ancaman Keamanan

Sasaran pengamanan adalah menghindari, mencegah dan mengatasi ancaman terhadap sistem. Kebutuhan keamanan sistem komputer dikategorikan ke dalam tiga aspek, yaitu :
1. Kerahasiaan (*secrecy*, diantaranya adalah *privacy*)
   Kerahasiaan adalah keterjaminan bahwa informasi di sistem komputer hanya dapat diakses oleh pihak-pihak yang diotorisasi dan modifikasi tetap menjaga konsistensi dan keutuhan data di sistem.

2. Integritas (*integrity)*
   Integritas adalah keterjaminan bahwa sumber daya sistem komputer hanya dapat dimodifikasi oelh pihak-pihak yang diotorisasi

3. Ketersediaan (*availabilitiy)*
   Ketersediaan adalah keterjaminan bahwa sumber daya sistem komputer tersedia bagi pihak-pihak yang diotorisasi saat diperlukan.

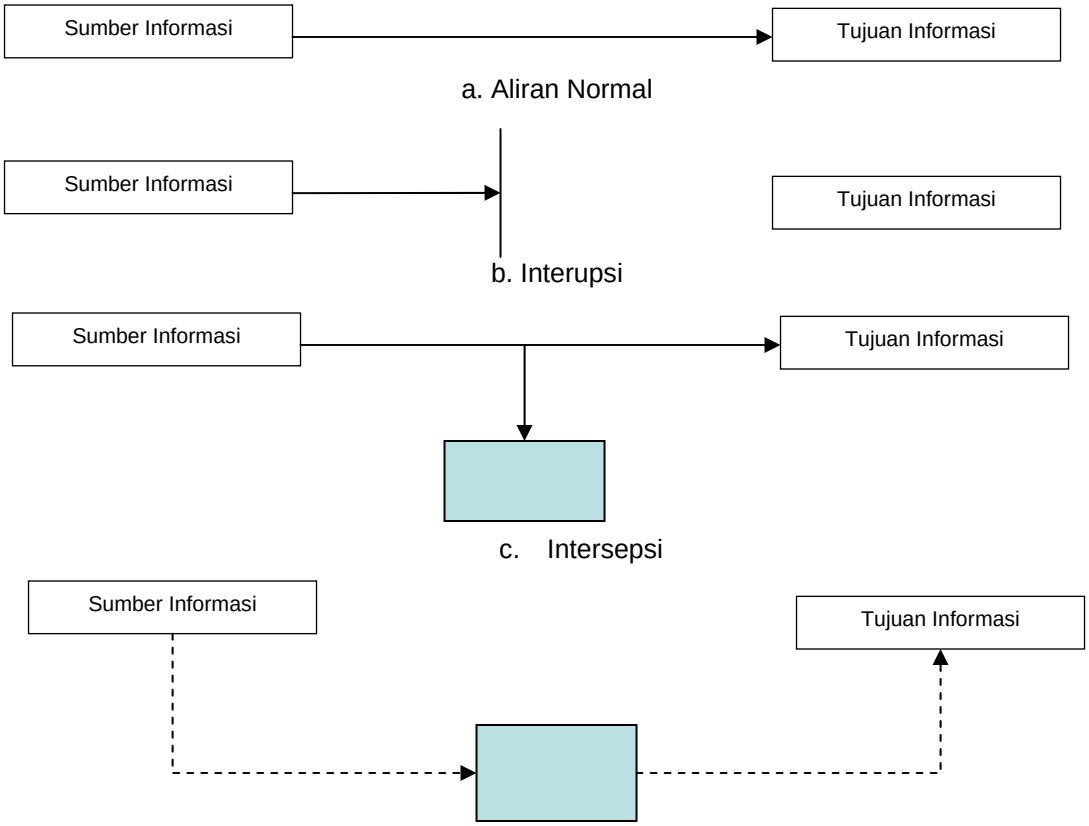
Tipe-tipe ancaman terhadap keamanan sistem komputer dapat dimodelkan dengan memandang fungsi sistem komputer sebagai penyedia informasi. Berdasarkan fungsi ini, ancaman terhadap sistem komputer dikategorikan menajdi empat ancaman, yaitu :

- Interupsi
  Sumber daya sistem komputer dihancurkan atau menjadi tak tersedia atau tak berguna. Interupsi merupakan ancaman terhadap ketersediaan.
  Contoh :
  - Penghancuran bagian perangkat keras, seperti harddisk
  - Pemotongan kabel komunikasi

- Intersepsi
  Pihak tak diotorisasi dapat mengakses sumber daya. Intersepsi merupakan ancaman terhadap keterahasiaan. Pihak tak diotorisasi dapat berupa orang atau progaram komputer.
  Contoh :
  - Penyadapan untuk mengambil data rahasia.
  - Mengkopi file tanpa diotorisasi

- Modifikasi
  Pihak tak diotorisasi tidak hanya mengakses tapi juga merusak sumber daya. Modifikasi merupakan ancaman terhadap integritas.
  Contoh :
  - Mengubah nilai-nilai file data
  - Mengubah program sehingga bertindak secara beda
  - Memodifikasi pesan-pesan yang ditransmisikan pada jaringan

- Fabrikasi
  Pihak tak diotorisasi menyisipkan atau memasukkan objek-objek palsu ke sistem. Fabrikasi merupakan ancaman terhadap integritas.
  Contoh :
  - Memasukkan pesan-pesan palsu ke jaringan
  - Penambahan record ke file.



Gambar 7.1. Skema ancaman terhadap Sistem Komputer

Petunjuk Pengaman Sistem

Saltzer dan Schrooder (1975) memberik petunjuk mengenai prinsip-prinsip pengamanan sistem komputer, yaitu :

1. Rancangan sistem seharusnya publik
   Keamanan sistem seharusnya tidak tergantung pada kerahasiaan rancangan mekanisme pengamanan. Mengasumsikan penyusup tidak akan mengetahui cara kerja sistem pengamanan hanya menipu atau memperdaya perancang sehingga tidak membuat mekanisme proteksi yang bagus.

2. Dapat diterima
   Skema yang dipilih harus dapat diterima secara psikologis. Mekanisme proteksi seharusnya tidak menganggu kerja pemakai dan memenuhi kebutuhan otorisasi pengaksesan. Jika mekanisme tidak mudah digunakan maka tidak akan digunakan atau digunakan secara tidak benar.

3. Pemeriksaan otoritas saat itu
   Sistem tidak seharusnya memeriksa ijin dan menyatakan penagaksesan diijinkan, serta kemudian menetapkan terus informasi ini untuk penggunaan selanjutnya. Banyak sistem memeriksa ijin ketika file dibuka dan setelah itu (operasi-operasi lain)

tidak diperiksa. Pemakai yang membuka file dan lupa menutup file akan terus dapat walaupun pemilik file telah mengubah atribut proteksi file.

4. Kewenangan serendah mungkin
   Progarm atau pemakai sistem seharusnya beroperasi dengan kumpulan wewenang serendah mungkin yang diperlukan untuk menyelesaikan tugasnya. *Default* sistem yang digunakan harus tak ada akses sama sekali.

5. Mekanisme yang ekonomis
   Mekanisme proteksi seharusnya sekecil, sesederhana mungkin dan seragam sehingga memudahkan verifikasi. Progeksi seharusnya dibangun di lapisan terbawah. Proteksi merupakan bagian integral rancangan sistem, bukan mekanisme yang ditambahakan pada rancangan yang telah ada.

Autentikasi Pemakai

Kebanyakan proteksi didasarkan asumsi sistem mengentahui identitas pemakai. Masalah identifikasi pemakai ketika *login* disebut otentifikasi pemakai (*user authentication)*. Kebanyakan metode atutentikasi didasarkan pada tiga cara, yaitu :

1. Suatu yang diketahui pemakai, misalnya :
   - passsword
   - kombinasi kunci
   - nama kecil ibu mertua, dsb
2. Sesuatu yang dimiliki pemakai, misalnya :
   - badge
   - kartu identitas
   - kunci, dsb
3. Sesuatu mengenai (merupakan ciri) pemakai, misalnya :
   - sidik jari
   - sidik suara
   - foto
   - tanda tangan, dsb

Password

Pemakai memilih satu kata kode, mengingatnya dan mengetikan saat akan mengakses sistem komputer. Saat diketikkan, komputer tidak menampilkan di layar. Teknik ini mempunyai kelemahan yang sanagt banyak dan mudah ditembus. Pemakai cenderung memilih *password* yang mudah diingat. Seseorang yang kenal dengan pemakai dapat mencoba login dengan sesuatu yang diketahuinya mengenai pemakai. Percobaan Morris dan Thompson menyatakan proteksi *password* dapat ditembus dengan mudah. Percobaan yang dilakukan adalah :

- terdapat file berisi nama depan, nama kecil, nama jalan, nama kota dari kamus ukuran sedang disertai dnegan pengejaan dibalik, nomor plat mobil yang valid dan *string-string* pendek karakter acak.
- Isian di file dicocokkan dengan file *password*

Hasil percobaan menunjukkan lebih dari 86% cocok dengan *password* digunakan pemakai di file *password*.

Upaya untuk lebih mengamankan proteksi password, antara lain :

1. Salting
   Menambahkan string lpendek ke string *password* yang diberikan pemakai sehingga mencapai panjang *password* tertentu.
2. One-Time Password

Pemakai harus mngganti *password* secara teratur. Upaya ini untuk membatasi peluang *password* telah diketehaui atau dicoba-coba pemakai lain. Bentuk ekstrim pendekatan ini adalah *one time password*, yaitu pemakai mendapat satu buku berisi dafatar *password.* Setiap kali pemakai login, pemakai menggunakan *password* berikutnya yang terdapat di daftar *password*, pemakai direpotkan keharusan menjaga agar buku passwrod-nya jangan sampai dicuri.

3. Satu Daftar Panjang Pertanyaan dan Jawaban
   Variasi terhadap *password* adalah mengharuskan pemakai memberii satu daftar pertanyaan panjang dan jawabannya. Pertanyaan-pertanyaan dan jawabannya dipilih pemakai sehingga pemakai mudah mengingatnya dan tak perlu menuliskan di kertas. Pada saat login, komputer memilih salah satu dari pertanayaan-pertanyaan secara acak, menanyakan ke pemakai dan memeriksa jawaban yang diberikan.
4. Tantangan-Tanggapan (Chalenge-Response)
   Pemakai diberi kebebasan memilih suatu algoritma, misalnya x3., Ketika pemakai login, komputer menuliskan di layar anga 3. Dalam kasus ini pemakai mengetik angka 27. Algoritma dapat berbeda di pagi, sore dan hari berbeda,dari terminal berbeda dan seterusnya.

Identifikasi Fisik

Pendekatan lain adalah memeriksa yang dimiliki pemakai, misalnya :

1. Kartu Berpita Magnetik
   Kartu pengenal dengan selarik pita magnetik. Kartu ini disisipkan ke suatu perangkat pembaca kartu magnetik jika akan mengakses komputer. Teknik ini biasanya dikombinasikan dengan *password* sehingga pemakai dapat pogin sistem komputer bila memenuhi dua syarat berikut :
   - memenuhi kartu
   - mengetahui password yang spesifik dari kartu itu
2. Sidik Fisik
   Pendekatan lain adalah mengukur ciri fisik yang sulit ditiru, seperti :
   - sidik jari dan sidik suara
   - analisis panjang jari
   - pengenalan visual dnegan menggunakan kamera .
3. Analisis Tanda Tangan
   Disediakan papan dan pen khusus dimana pemakai menulis tanda tangan. Pada teknik ini, bukan membandingkan bentuk tanda tangan tapi gerakan (arah) dan tekanan pena saat menulis. Seseorang dapat meniru bentuk tanda tangan, sulit meniru persis cara (gerakan dinamis dan irama tekanan) saat pmebuatan tanda tangan.
4. Analsis Suatu yang Dipunyai Pemakai
   Pendekatan lain adalah meniru perilaku kucing dan anjing dalam menandai batas wilayah, yaitu *urine*. Disediakan alat *urinalysis*. Bila pemakai ingin login, maka pemakai harus membawa sample urine-nya. Sampel urine dimasukkan ke tabung dan segera dilakukan analisis dan menentukan apakah termasuk salah satu pemakai sistem. *Urinalysis* harus dapat dilakukan sesaat.
5. Analisis Darah
   Disediakan satu jarum dimana pemakai dapat mencobloskan jari sampai menetes darahnya. Darah itu kemudian dianalisis dengan spektografi . Dari analisis dapat ditentukan mengenai pemilik darah. Pendekatan ini relatif aman, tapi tidak diterima secara psikologis.

Pembatasan

Pembatasan-pembatasan dapat dilakukan sehingga memperkecil peluang penembuasan oleh pemakai yang tak diotorisasi. Misalnya :

- Pembatasan login

- Pembatasan dengan *call back*
- Pembatasan jumlah usaha login

Pembatasan Login
Login hanya dibolehkan :
- pada terminal tertentu
- hanya pada waktu dan hari tertentu

Pembatasan dengan *Call Back*

Login dapat dilakukan oleh siapa pun. Bila telah sukses login, system segera memutuskan koneksi dan memanggil nomor telepon yang telah disepakati. Penyusup tidak dapat menghubungi lewat sembarang saluran telepon, tapi hanya pada saluran telepon tertentu.

Pembatasan Jumlah Usaha Login

Login dibatasi sampai tiga kali dan segera dikunci dan diberitahu ke administrator. Semua login direkam dan sistem operasi melaporkan informasi-informasi berikut :
- waktu, yaitu waktu pemakai login
- terminal, yaitu terminal pemakai login

Mekanisme Proteksi Sistem Komputer

Masalah proteksi adalah mengenai cara mencegah proses-proses mengakses objek-objek yang tidak diotorisasi. Mekanisme ini juga harus memungkinkan membatasi proses-proses ke suatu sub-set operasi legal yang diperlukan. Misalnya, proses A dapat membaca file F, tapi tidak menulisinya. Agar dapat menyediakan mekanisme proteksi berbeda dikembangkan berdasarkan konsep *domain*. *Domain* adalah himpunan pasangan (objek, hak). Tiap pasangan menspesifikasikan objek dari suatu subset operasi yang dapat dilakukan terhadapnya. Hak dalam konteks ini berarti ijin melakukan suatu proses. Proses berjalan pada suatu *domain* proteksi, yaitu proses merupakan anggota suatu *domain* atau berberapa *domain*. lTerdapata kumpulan objek yang dapat diakses proses. Untuk tiap objek, proses mempunyai suatu kumpulan hak terhadap objek itu. Proses-prosess dapat ljuga lberalih daru satu *domai* ke *domain* lain selama eksekusi. Aturan peralihan *domain* ini bergantung pada sistem. *Domain* ditetapkan dengan mendaftarkan pemakiai-pemakai yang termasuk *domiaii itu.* Proses-proses yang dijalankan lpemakai adalah proses-proses pada *domain* itu dana mempunyai hak akses terhadap objek seperti ditentukan *domain*-nya.

Cara Penyimpanan Informasi Anggota Domain

Secara konseptual adalah berupa matriks besar, dimana :
- baris menunjukkan *domain*
- kolom menunjukkan objek

Tiap elemen matriks mendaftarakan hak-hak yang dimiliki *domain* terhadap objek. Dengan matriks ini, sistem dapat mengetahui hak pengaksesan terhadap objek. Gambar di bawah ini menunjukkan matriks pengaksesan objek.

	File1	File2	Printer1	Plotter1	Modem1

Domain 1	Read	Read	Write		
Domain 2	Read			Write	Write
Domain 3		Read Write Execute	Write	Write	Write

Gambar 7.2. Matriks Pengaksesan Objek

Untuk sistem-sistem yang mengijinkan peralihan *domain* dimodelkan dnegan menanggap *domain* sebagai objek, yaitu :

- Jika terdapat operasi ENTER berarti mempunyai hak berpindah *domain*.

	File1	File2	Printer1	Ploter1	Modem1	Domain1	Domain2	Domain3
Domain 1	Read	Read Write	Write				Enter	
Domain 2	Read			Write	Write	Enter		
Domain 3		Read Write Exercise						

Gambar 7.3. Matriks Pengaksesan Objek dengan Operasi Peralihan Domain

Gambar di atas menunjukkan matriks pengaksesan objek dengan operasi pengalihan *domain*. Proses-proses pada *domain* 1 dapat berpindah ke *domain* 2 dan proses *domain* 2 dapat berpindah ke *domain* 1.

A C L (Access Control List)

Matriks pengaksesan objek akan berbentuk matriks jarang (sparse matrix). Matriks jarang memboroskan ruang penyimpanan dan lambat karena memerlukan ruang besar. Dua alternatif unutk memperbaikinya adalah :

- menyimpan matriks sebagai baris
- menyimpan matriks sebagai kolom

Penyimpanan dilakukan hanya untuk isian yang tak kosong.

Teknik Penyimpanan Per Kolom

Teknik yang digunakan adalah mengasosiasikan tiap objek dengan senarai terurut, berisi semua *domain* yang boleh mengakses dan operasi-operasi yang dibolehkan. Teknik ini menghasilkan senarai yang disebut ACL (Access Control List).
Contoh :

File1	(Gadis,*,rwx),(Soni,*,rw-)
File2	(Gadis, system,rwx)
Printer1	(Gadis,*,-w-),(Putri,karyawan,-w-)
Plotter1	(Dewa,*,-w-)
Modem1	(Odic,*,-w-)

Gambar 7.4. ACL (Access List Control)

Tiap ACL yang disebutkan di kurung menyatakan komponen uid, gid dan hak akses. Dengan ACL, dimungkinkan mencegah uid, gid spesifik mengakses objek sementara mengijinkan yang lain. Pemilik objek dapat mengubah ACL kapan pun. Cara ini untuk mempermudah pencegahan/pelanggaran pengaksesan yang sebelumnya diperbolehkan.

Kapabilitas

Cara lain adalah memecah matriks per baris. Diasosiasikan tiap proses satu daftar objek yang boleh diakses, bila terdapta tanda operasi yang diijinkan padanya atau domainnya. Senarai ini disebut senarai kapabilitas (*capabilities list)*.
Contoh :

	Tipe	Hak	Objek
0	File	Rwx	Pointer ke file2
1	Printer	-w-	Pointer ke printer1
2	Plotter	-w-	Pointer ke plotter1
3	Modem	-w-	Pointer ke modem1

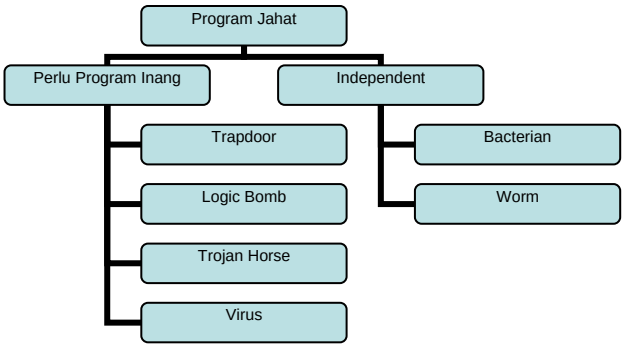
Gambar 7.5. Senarai Kapabilitas untuk Domain 3

Program-program Jahat

Ancaman-ancaman canggih terhadap sistem komputer adalah programyang mengeksploitasi kelemahan sistem komputer. Kita berurusasn dengan program aplikasi begitu juga program utilitas, seperti editor dan kompilator.

Bowles memberikan taksonomi ancaman perangkat lunak atau klasifikasi program jahat (*malicious program*). Gambar 7.6 men menunjukkan taksonomi yang diberikan oleh Bowles.
Ancaman-ancaman itu dapat dibagi menjadi dua kategori, yaitu :

1. Program-program yang memerlukan program inang (*host program)*
   Fragmen progam tidak dapat mandiri secara independen dari suatu program aplikasi, program utilitas atau program sistem.
2. Program-program yang tidak memerlukan program inang
   Progarm sendiri yang dapat dijadualkan oleh sistem operasi.



Gambar 7.6. Taksononomi program-program jahat

Pembagian atau taksonomi Bowles menghasilkan tipe-tipe program jahat sebagai berikut:

1. *Bacteria*
2. *Logic Bomb*
3. *Trapdoor*
4. *Trojan Horse*
5. *Virus*
6. *Worm*

Bacteria

Bacteria adalah program yang mengkonsumsi sumber daya sistem dengan mereplikasi dirinya sendiri. *Bacteria* tidak secara eksplisit merusak file. Tujuan program ini hanya satu,yaitu mereplikasi dirinya. Program *bacteria* yang sederhanan bisa hanya mengeksekusi dua kopian dirinya secara simultan pada sistem *multiproramming* atau menciptakan dua file baru, masing-masing adalah kopian file program *bacteria*. Kedua kopian ini kemudian mengkopi dua kali dan seterusnya.

Bacteria bereproduksi secara eksponensial, dengna cepat mengambil alih seluruh kapsitas pemroses, memori atau ruang disk, mengakibatkan penolakan pengaksesan pemakai ke sumber daya.

Logic Bomb

Logic Bomb adalah logik yang ditempelkan pada program komputer agar memeriksa suatu kumpulan kondisi di sistem Ketika kondisi-kondisi yang dimaksud ditemui, logik mengeksekusi suatu fungsi yang menghasilkan aksi-aksi tak diotorisasi.

Logic Bomb menempel pada suatu program resmi yang di-set “meledak” ketika kondisi-kondisi tertentu dipenuhi. Contoh kondisi-kondisi untuk memicu *logic bomb* adalah ada atau tidak adanya file-file tertentu, hari tertentu dari minggu atau tanggal atau pemakai dan pola *bit* yang sama di semua kopiannya. Teknik ini terbatas untuk deteksi *virus-virus* yang telah dikenal. Tipe lain *anti virus* generasi pertama adalah mengelola rekaman panjang (ukuran) program dan memeriksa perubahan panjang program.

Virus

Virus komputer adalah buatan manusia yang bertujuan merugikan orang lain. Ancaman yang paling serius dari sebuah virus komputer adalah sifatnya yang merusak. Tidak semua program virus dibuat untuk merusak. Ada yang mungkin membuat virus dengan tujuan melindungi hasil karya sendiri.

Perbedaa yang sangat mendasar antara virus dengan worm meskipun sama-sama mempunyai sifat merusak dan kekuatannya untuk memperbanyak diri, worm tidak mampu menempelkan dirinya pada program lain. Worm hanya memperbanyak diri dengan memakan ruang kosong dalam memori komputer. Kegiatan memperbanyak diri ini dilakuakan terus sampai memori komputer menjadi penuh dan sistem menjadi macet.

Kuda troya merupakan teknik integrasi yang sering dilakukan oleh virus dengan membuat suatu program yang bermanfaat, tetapi dalam program ini diselipkan suatu program yang amat berbahaya karena dapat menghancurkan atau menghapus data yang ada dalam disket atau harddisk. Program ini berupa virus komputer. Oleh karena itu apabila komputer diaktifkan, secara tidak disengaja juga mengaktifkan program virus.

Trapdoor merupakan suatu titik masuk ke dalam suatu sistem dengan cara mem-bypass sistem keamanan. Dengan demikian, seseorang dapat melakukan sesuatu dengan sistem tersebut. Para hacker pada umumnya menggunakan cara itu untuk bisa masuk ke sistem komputer yang ingin dirusaknya.

Ada empat cara membagi jenis virus komputer , yaitu :

1. Berdasarkan cara penularannya atau penyebarannya
2. Berdasarkan keganasannya
3. Berdasarkan maksud dan tujuan pembuatan virus
4. Berdasarkan sistem operasinya

Cara virus masuk ke sistem

Berdasarkan berbagai evaluasi mengenai virus diketahui bahwa virus-virus canggih dibentuk dengan empat komponen utama, yaitu :

a. inisialisasi ke memori
b. menyalinkan dirinya ke disk
c. beraksi

Tahap inisialisasi merupakan tahap awal kegiatan virus. Beberapa cara yang digunakan virus untuk melakukan tahap ini, antara lain :

a. memodifikasi ke dalam bentuk file .exe atau .com
b. memodifikasi atau mengganti boot record
c. memodifikasi atau mengganti partisi record
d. memodifikasi atau mengganti program kerja peralatan komputer
e. memodifikasi file-file overlay

Worm

Worm adalah programyang dapat mereplikasi dirinya dan mengirim kpian dari komputer ke komputer lewat hubungan jaringan. Begitu tiba, *worm* diaktifkan untuk mereplikasi dan propagasi kembali. Selain hanya propagasi, *worm* biasanya melakukan fungsi yang tak diinginkan.

Network worm menggunakan hubungan jaringan untuk menyebar dari sistem ke sistem lain. Sekali aktif di suatu sistem, *network worm* dapat berlaku seperti virus atau bacteria atau menempelkan program *trojan horse* atau melakukan sejumlah aksi menjengkelkan atau menghancurkan.

Untuk mereplikasi dirinya, *network worm* menggunakan suatu layanan jaringan, seperti :

- fasilitas surat elektronik *(electronic mail facility)* yaitu *worm* mengirimkan kopian dirinya ke sistem-sistem lain
- kemampuan eksekusi jarak jauh (*remote execution capability)* yaitu *worm* mengeksekusi kopian dirinya di sistem lain.
- Kemampuan *login* jarak jauh (*remote login capability)* yaitu *worm log* pada sistem jauh sebagai pemakai dan kemudian menggunakan perintah untuk mengkopi dirinya dari satu sistem ke sistem lain.

Kopian program *worm* yang lbaru kemudian dijalankan di sistem jauh dan melakukan fungsi-fungsi lain yang dilakukan di sistem itu, *worm* terus menyebar dengan cara yang sama.

Network worm mempunyai ciri-ciri yang sama dengan virus komputer, yaitu mempunyai fase-fase :

- *Dormant phase*
- *Propagation phase*
- *Trigerring phase*
- *Execution phase*

Network worm juga berusaha menentukan apakah sistem sebelumnya telah diinfeksi sebelum mengirim kopian dirinya ke sistem itu.

Virus dan Antivirus

Siklus Hidup Virus

Virus adalah sama dengan program komputer lain. Perbedaan dengan program lain adalah virus dapat mencantolkan dirinya ke program lain dan mengeksekusi kodenya secara rahasia setiap kali program inang berjalan. Masalah yang ditimbulkan virus adalah virus sering merusak sistem komputer seperti menghapus file, partisi disk atau mengacaukan program.

Virus mengalami siklus hidup dalam empat tahap (fase), yaitu :

1. fase tidur (*dormant phase)*
2. fase propagasi (*propagation phase)*
3. fase pemicuan (*trigerring phase)*
4. fase eksekusi (*execution phase)*

Fase Tidur (*Dormant Phase)*

Virus dalam keadaan menganggur. Virus akan tiba-tiba aktif oleh suatu kejadian seperti tibanya tanggal tertentu, kehadiran program atau file tertentu, atau kapasitas disk yang melewati batas. Tidak semua virus mempunyai tahap ini.

Fase Propagasi (*Propagation Phase)*

Virus menempatkan kopian dirinya ke program lain atau daerah sistem tertentu di disk. Program yang terinfeksi virus akan mempunyai kloning virus. Kloning virus itu dapat kembali memasuki fase propagasi.

Fase Pemicuan (*Trigerring Phase)*

Virus diaktifkan untuk melakukan fungsi tertentu. Seperti pada fase tidur, fase pemicuan dapat disebabkan beragam kejadiansistem termasuk penghitungan jumlah kopian dirinya.

Fase Eksekusi (*Excecution Phase)*

Virus menjalankan fungsinya. Fungsinya mungkin sepele seperti sekedar menampilkanpesan di layar atau merusak seperti merusak program dan file-file data dan sebagainya.

Kebanyakan virus melakukan kerjanya untuk suatu sistem operasi tertentu, lebih spesifik lagi pada platform perangkat keras tertentu. Virus-virus dirancang memanfaatkan rincian-rincian dan kelemahan-kelemahan sistem.

Infeksi Virus

Sekali virus telah memasuki sistem dengan menginfeksi satu program, virus berada dalam posisi menginfeksi beberarapa atau semua file exe lain di sistem itu saat program yang terinfeksi dieksekusi. Infeksi virus dapat sepenuhnya dihindari dengna mencegah virus masuk sistem. Pencegahan ini sangat luar biasa sulit karena virus dapat menjadibagian program di luar sistem.

Kebanyakan virus mengawali infeksinya pengkopian disk yang telah terinfeksi virus. Banyak disk berisi game atau utilitas di rumah dikopikan ke mesin kantor. Disk berisi virus pun dapat terdapat di disk yang dikirim produsen aplikasi. Hanya sejumlah kecil infeksi virus yang dimulai dari hubungan jaringa.

Tipe-tipe Virus

Saat ini perkembangan virus masih berlanjut, terjadi perlombaan antara penulis virus dan pembuat antivirus. Begitu satu tipe dikembangkan antivirus-nya, tipe virus yang lain muncul.

Klasifikasi tipe virus adalah sebagai berikut :

- *Parasitic virus*
- *Memory resident virus*
- *Boot sector virus*
- *Stealth virus*
- *Polymorphic virus*

Parasitic Virus

Merupakan virus tradisional dan bentuk virus yang paling sering. Tipe ini mencantolkan dirinya ke file exe. Virus mereplikasi ketika program yang terinfeksi dieksekusi denganmencari file-file exe lain untuk diinfeksi

Memory Resident Virus

Virus memuatkan diri ke memori utama sebagai bagian program yang menetap. Virus menginfeksi setiap program yang dieksekusi

Boot Sector Virus

Virus menginfeksi master *boot record* atau *boot record* dan menyebar saat sistem di *boot* dari *disk* yang berisi virus

Stealth Virus

Virus yang bentuknya telah dirancang agar dapat menyembunyikan diri dari deteksi perangkat lunak antivirus.

Polymorphic Virus

Virus bermutasi setiap kali melakukan infeksi. Deteksi dengan “penandaan” virus tersebut tidak dimungkinkan.

Penulis virus dapat melengkapi dnenga alat-alat bantu penciptaan viru baru (*virus creation tollkit)* yaitu rutin-rutin untuk menciptakan virus-virus baru. Dengan alat bantu ini penciptaan virus baru dapat dilakukan dengan sangat cepat. Virus-virus yang diciptakan dengan alta bantu biasanay kurang canggih dibanding virus-virus yang dirancang dari awal.

Antivirus

Solusi ideal terhadap ancaman virus adalah pencegahan. Jangan ijinkan virus masuk ke sistem. Sasaran ini , tak mungkin dilaksanakan sepenuhnya. Pencegahan dapat mereduksi sejumlah serangan virus. Setelah pencegahan terhadap masuknya virus, maka pendekatan berikutnya yang dapta dilakukan adalah :

- Deteksi
- Identifikasi
- Penghilangan

Deteksi

Begitu infeksi terlah terjadi, tentukan apakah infeksi memang telah terjadi dan cari lokasi virus.

Identifikasi

Begitu virus terdeteksi, maka identifikasi virus yang menginfeksi program.

Penghilangan

Begitu virus dapat diidentifikasi, maka hilangkan semua jejak virus dari program yang terinfeksi dan program dikembalikan ke semula (sebelum terinfeksi).

Jika deteksi sukses dilakukan, tapi identifikasi atau penghilangan tidak dapat dilakukan, maka alternatif yang dilakukan adalah hapus program yang terinfeksi dan kopi kembali backup program yang masih bersih.

Sebagaimana virus berkembang dari yang sederhana menjadi semakin canggih, begitu juga paket perangkat lunak antivirus. Saat ini program antivirus semakin kompleks dan canggih. Perkembangan program antivirus dapat diperiode menjadi empat generasi :

1. Generasi pertama : sekedar scanner biasa
2. Generasi kedua : scanner yang pintar (*heuristic scanner)*
3. Generasi ketiga : jebakan-jebakan aktifitas (*activity trap)*
4. Generasi keempat : proteksi penuh (*full featured protection)*

Generasi Pertama

Antivirus men-scan program untuk menemukan penanda (*signature*) virus. Walaupun virus mungkin berisi “karakter-karakter varian” tapi secara esensi mempunyai struktur dan pola bit yuang sama di semua kopiannya. Teknik ini terbatas untuk deteksi viru-virus yang telah dikenal. Tipe lain antivirus generasi pertama adalah mengelola rekaman panjang (ukuran) program dan memeriksa perubahan panjang program.

Generasi Kedua

Antivirus men-scan tidak bergantung pada penanda spesifik. Antivirus menggunakan aturan-aturan pintar (*heuristic rules)* untuk mencari kemungkinan infeksi

virus. Teknik yang dipakai misalnya mencari fragmen-fragmen kode yang sering merupakan bagian virus. Contohnya antivirus mencari awal *loop* enkripsi yang digunakan *polymorphic virus* dan menemukan kunci enkripsi. Begitu kunci ditemukan, antivirus dapat mend-dekripisi virus untuk identifikasi dan kemudian menghilangkan infeksi virus.

Teknik lain adalah pemeriksaan integritas. *Cheksum* dapat ditambahkan di tiap program. Jika virus menginfkesi program tanpa mengubah *cheksum*, maka pemeriksaan integritas akan menemukan perubahan itu. Untuk menanggulangi virus canggih yang mampu mengubah *cheksum* saat menginfeksi program, fungsi *has* ter-enkripsi digunakah. Kunci enkripsi disimpan secara terpisah dari program sehingga program tidak dapat menghasilkan kode *hash* baru dan meng-enkrisinya. Dengan menggunakan fungsi *hash* bukan *cheksum* sederhana maka mencegah virus menyesuaikan program yang menghasilaknkode *hash* yang sama seperti sebelumnya.

Generasi Ketiga

Program antivirus merupakan program yang menetap di memori (*memory resident program)*. Program ini mengidentifikasi virus melalui aksi-aksinya bukan dari struktur program yang diinfeksi. Dengan antivirus semacam ini tak perlu mengembangkan penanda-penanda dan aturan-aturan pintar untuk beragam virus yang sangat banyak. Dengan cara ini yang diperlukan adalah mengidentifikasi kumpulan instruksi yang berjumlah sedikit yang mengidentifikasi adanya usaha infeksi. Kalau muncul kejadian ini, program antivirus segera mengintervensi.

Generasi Keempat

Antivirus generasi ini menggunakan beragam teknik antivirus secara bersamaan. Teknik-teknik ini meliputi *scanning* dan jebakan-jebakan aktifitas. Antivirus juga mempunyai senarai kapabilitas pengaksesan yang membatasi kemampuan virus memasuki sistem dan membatasi kemampuan virus memodifikasi file untuk menginfeksi file.

Pertempuran antara penulis virus dan pembuat antivirus masih berlanjut. Walau beragam strategi lebihlengkap telah dibuat untuk menanggulangi virus, penulis virus pun masih berlanjut menulis virus yang dapat melewati barikade-barikade yang dibuat penulis antivirus. Untuk pengamanan sistem komputer, sebaiknya pengaksesan dan pemakaian komputer diawasi dengan seksama sehingga tidak menjalankan program atau memakai disk yang belum terjamin kebersihan dari infeksi virus. Pencegahan terbaik terhadap ancaman virus adalah mencegah virus memasuki sistemdi saat pertama.