PERTEMUAN 3
OPERATING SYSTEM SECURITY

A. TUJUAN PEMBELAJARAN

Pada Pertemuan ini akan dijelaskan mengenai keamanan sistem operasi. Setelah mempelajari materi ini mahasiswa diharapkan mampu untuk:

1. Memahami konsep sistem operasi
2. Meningkatkan pertahanan sistem operasi
3. Membuat mekanisme perlindungan tambahan pada sistem operasi
4. Memahami keamanan program aplikasi

B. URAIAN MATERI

1. Memahami Konsep Sistem Operasi

Merancang dan menerapkan virus komputer dan jenis perangkat lunak berbahaya lainnya biasanya memerlukan pengetahuan yang baik tentang operasi, prosedur, dan variabel internal serta tabel sistem operasi yang akan diserang virus. Sebaliknya, mereka yang hanya ingin memahami virus hanya membutuhkan pemahaman umum terkait sistem operasi dan cara kerjanya.

Sistem operasi adalah sekumpulan rutinitas yang memberikan layanan kepada pengguna dan memudahkan mereka untuk menggunakan komputer. Dalam komputer multiuser, sistem operasi juga mengawasi pengguna, melindungi setiap pengguna dari pengguna lain, juga melindungi dirinya dari kerusakan yang tidak disengaja dan disengaja oleh pengguna.

Sistem operasi merupakan garis pertahanan pertama terhadap segala macam perilaku yang tidak diinginkan. Ini melindungi satu pengguna dari yang lain, memastikan bahwa area kritis memori atau penyimpanan tidak ditimpa oleh proses yang tidak sah, melakukan identifikasi dan otentikasi orang dan operasi jarak jauh, dan memastikan pembagian yang adil dari sumber daya perangkat keras penting.

Sistem operasi adalah pengendali fundamental dari semua system sumber daya, juga menjadikannya target utama serangan**.** Ketika sistem operasi menginisialisasi pada waktu boot sistem, ia memulai tugas dalam urutan yang

teratur, seperti, pertama, fungsi primitif dan driver perangkat, kemudian pengontrol proses, diikuti oleh rutinitas manajemen file dan memori, dan terakhir, antarmuka pengguna. Untuk membangun keamanan, tugas awal membentuk pertahanan yang kokoh untuk membatasi tugas selanjutnya. Fungsi sistem operasi primitif, seperti komunikasi antarproses serta input dan output dasar, harus mendahului struktur yang lebih kompleks seperti file, direktori, dan segmen memori, sebagian karena fungsi primitif ini diperlukan untuk mengimplementasikan konstruksi yang terakhir. Dan juga karena komunikasi dasar diperlukan agar fungsi sistem operasi yang berbeda dapat berkomunikasi satu sama lain. Aplikasi antivirus biasanya terlambat dimulai karena merupakan add-on pada sistem operasi; tetap saja, kode antivirus harus dikontrol sebelum sistem operasi mengizinkan akses ke objek baru yang mungkin berisi virus. Jelas, perangkat lunak pencegahan dapat melindungi hanya jika aktif sebelum kode yang berbahaya (*malicious code*).

Tetapi bagaimana jika malware menyematkan dirinya di sistem operasi, sehingga aktif sebelum komponen sistem operasi yang mungkin mendeteksi atau memblokirnya? Atau bagaimana jika malware dapat mengakali atau mengambil alih bagian lain dari sistem operasi? Pengurutan ini mengarah pada kerentanan yang penting: Mendapatkan kendali sebelum pelindung berarti bahwa kekuatan pelindung terbatas. Dalam kasus tersebut, penyerang memiliki kendali penuh atas sistem: Kode berbahaya tidak dapat terdeteksi dan tidak dapat dihentikan. Karena malware beroperasi dengan hak istimewa dari root sistem operasi, itu disebut rootkit. Meskipun menyematkan rootkit di dalam sistem operasi itu sulit, upaya yang berhasil pasti sepadan.

2. Meningkatkan Keamanan Sistem Operasi

Sebagian besar pekerjaan dalam keamanan dan perlindungan yang berkaitan dengan OS secara kasar dapat dikelompokkan menjadi tiga kategori:

a. Access Control. Berkaitan dengan pengaturan akses pengguna ke sistem total, sub-sistem, dan data, dan pengaturan akses proses ke berbagai sumber daya dan objek di dalam sistem.
b. Kontrol arus informasi. Mengatur aliran data dalam sistem dan pengirimannya ke pengguna.

c. Sertifikasi. Berkaitan dengan pembuktian bahwa mekanisme kontrol akses dan aliran bekerja sesuai dengan spesifikasinya dan bahwa mereka menegakkan kebijakan perlindungan dan keamanan yang diinginkan.

a. Aset-aset Sistem Komputer

Aset sistem komputer dapat dikategorikan sebagai perangkat keras, perangkat lunak, dan data.

1) Hardware. Ancaman perangkat keras sistem komputer ada di area ketersediaan (*availability*). Perangkat keras paling rentan terhadap serangan, juga tidak dapat menerima kontrol otomatis. Ancaman terbilang kerusakan yang tidak disengaja dan disengaja pada peralatan serta pencurian. Perkembangan komputer pribadi dan workstation serta peningkatan penggunaan jaringan area lokal (LAN) meningkatkan potensi kerugian pada area ini. Tindakan pengamanan fisik dan administratif diperlukan untuk menghadapi ancaman ini.
2) Software. OS, utilitas, dan program aplikasi adalah yang membuat perangkat keras sistem komputer berguna bagi bisnis dan individu. Beberapa ancaman berbeda perlu dipertimbangkan. Ancaman perangkat lunak ialah serangan terhadap ketersediaan (*availability*). Perangkat lunak, terutama aplikasi, secara mengejutkan mudah dihapus. Perangkat lunak pun bisa diubah atau rusak untuk membuatnya tidak berguna atau berbahaya. Manajemen konfigurasi *software* yang cermat, termasuk membuat *backup* versi terbaru, dapat menjaga ketersediaan tinggi. Perkara yang kian sulit dihadapi yakni alterasi *software* yang membuahkan program yang masih berfungsi dengan *behaviour* berbeda dari sebelumnya. Masalah terakhir adalah kontrol atau kepemilikan perangkat lunak. Sekalipun upaya pencegahan tertentu tersedia, pada umumnya perkara penyalinan *software* yang tidak sah belum terpecahkan.
3) Data. Keamanan perangkat keras dan perangkat lunak biasanya menjadi perhatian profesional pusat komputasi atau masalah individu pengguna komputer pribadi. Masalah yang jauh lebih luas adalah keamanan data, yang melibatkan file dan bentuk data lain yang dikendalikan oleh individu, kelompok, dan organisasi bisnis. Masalah keamanan sehubungan dengan data bersifat luas, mencakup kerahasiaan, kendali atau

kepemilikan, integritas, keaslian, ketersediaan, dan utilitas. Untuk perlakuan teoritis yang baik dari atribut informasi yang harus dilindungi melalui tindakan pengamanan

b. Prinsip desain keamanan

Saltzer dan Schroeder mengidentifikasi sejumlah prinsip untuk desain langkah-langkah keamanan untuk berbagai ancaman terhadap sistem komputer.

1) Hak istimewa terkecil (least privilege). Setiap program dan setiap pengguna sistem harus beroperasi menggunakan hak istimewa paling sedikit yang diperlukan untuk menyelesaikan pekerjaan. Hak akses harus diperoleh hanya dengan izin eksplisit; defaultnya adalah "tidak ada akses".
2) Mekanisme ekonomis. Mekanisme keamanan harus sekecil dan sesederhana mungkin, membantu verifikasi mereka. Ini biasanya berarti bahwa mereka harus menjadi bagian integral dari desain daripada mekanisme tambahan untuk desain yang ada.
3) Dapat diterima. Mekanisme keamanan tidak boleh terlalu mengganggu pekerjaan pengguna. Pada saat yang sama, mekanisme tersebut harus memenuhi kebutuhan mereka yang memberi otorisasi akses. Jika mekanismenya tidak mudah digunakan, kemungkinan besar mekanisme tersebut tidak digunakan atau digunakan secara tidak benar.
4) Mediasi lengkap. Setiap akses harus diperiksa terhadap informasi kontrol akses, termasuk akses yang terjadi di luar operasi normal, seperti dalam pemulihan atau pemeliharaan.
5) Desain terbuka (open design). Keamanan sistem tidak boleh bergantung pada kerahasiaan desain mekanismenya. Dengan demikian, mekanismenya dapat ditinjau oleh banyak ahli, dan pengguna dapat memiliki kepercayaan yang tinggi terhadapnya.

c. Memperkuat Ketahanan Sistem Operasi

Langkah penting pertama dalam mengamankan sistem adalah mengamankan sistem operasi dasar yang menjadi sandaran semua aplikasi dan layanan lain. Landasan keamanan yang baik membutuhkan sistem operasi yang dipasang, ditambal (*patched)*, dan dikonfigurasi dengan benar. Sayangnya, konfigurasi default untuk banyak sistem operasi seringkali

memaksimalkan kemudahan penggunaan dan fungsionalitas, daripada keamanan. Juga, setiap organisasi memiliki kebutuhan keamanannya sendiri, profil keamanan yang sesuai, dan karenanya konfigurasinya juga akan berbeda.

NIST SP 800-123 menyarankan langkah-langkah dasar berikut yang harus digunakan untuk mengamankan sistem operasi:

1) Instal dan *patch* sistem operasi.
2) Perkuat dan konfigurasikan sistem operasi untuk memenuhi kebutuhan keamanan sistem yang teridentifikasi secara memadai dengan:
   a) Menghapus layanan, aplikasi, dan protokol yang tidak perlu.
   b) Konfigurasi pengguna, grup, dan izin.
   c) Konfigurasi kontrol sumber daya.
3) Instal dan konfigurasikan kontrol keamanan tambahan, seperti anti-virus, firewall berbasis host, dan *intrusion detection system* (IDS), jika diperlukan.
4) Uji keamanan sistem operasi dasar untuk memastikan bahwa langkah-langkah yang diambil telah memenuhi kebutuhan keamanannya secara memadai.

3. Membuat mekanisme perlindungan tambahan pada sistem operasi

Pengenalan multiprogramming menghasilkan kemampuan untuk berbagi (*sharing*) sumber daya di antara pengguna. *Sharing* disini tidak hanya melibatkan prosesor tetapi juga:

a. Memory
b. I/O devices, seperti disk dan printer
c. Program
d. Data

Kemampuan untuk berbagi sumber daya ini memperkenalkan kebutuhan akan perlindungan. Pfleeger dan Pfleeger menunjukkan bahwa OS mungkin menawarkan perlindungan sepanjang spektrum ini:

a. Tidak ada proteksi. Ini sesuai jika prosedur sensitif dijalankan pada waktu yang berbeda.

b. Isolasi. Pendekatan ini menyiratkan bahwa setiap proses beroperasi secara terpisah dari proses lain, tanpa berbagi atau komunikasi. Setiap proses memiliki ruang alamat, file, dan objek lainnya sendiri.
c. Bagikan semua atau jangan bagikan apa pun. Pemilik suatu objek (misalnya, file atau segmen memori) mendeklarasikannya sebagai publik atau privat. Dalam kasus sebelumnya, proses apa pun dapat mengakses objek; pada yang terakhir, hanya proses pemilik yang dapat mengakses objek.
d. Bagikan melalui batasan akses. OS memeriksa izin setiap akses oleh pengguna tertentu ke objek tertentu. Oleh karena itu, OS bertindak sebagai penjaga, atau penjaga gerbang, antara pengguna dan objek, memastikan bahwa hanya akses resmi yang terjadi.
e. Berbagi melalui kemampuan dinamis. Ini memperluas konsep kontrol akses untuk memungkinkan kreasi dinamis dari hak berbagi untuk objek.
f. Batasi penggunaan suatu objek. Bentuk perlindungan ini tidak hanya membatasi akses ke suatu objek tetapi penggunaan tempat objek tersebut dapat diletakkan. Misalnya, pengguna mungkin diizinkan untuk melihat dokumen sensitif tetapi tidak dapat mencetaknya. Contoh lain adalah bahwa pengguna mungkin diizinkan mengakses database untuk memperoleh ringkasan statistik tetapi tidak untuk menentukan nilai data tertentu.

Daftar diatas dicantumkan secara kasar dalam urutan tingkat kesulitan untuk diterapkan, tetapi juga dalam urutan peningkatan kehalusan perlindungan yang mereka berikan. OS tertentu dapat memberikan tingkat perlindungan yang berbeda untuk objek, pengguna, atau aplikasi yang berbeda.

OS perlu menyeimbangkan kebutuhan untuk memungkinkan berbagi, yang meningkatkan utilitas sistem komputer, dengan kebutuhan untuk melindungi sumber daya pengguna individu. Bagian ini membahas beberapa mekanisme yang digunakan OS untuk menerapkan perlindungan untuk objek ini.

a. Perlindungan Memori

Dalam lingkup multiprogram, perlindungan memori utama (*random-access memory*, atau RAM) sangat penting. Perhatian di sini bukan hanya keamanan tetapi fungsi yang benar dari berbagai proses yang aktif. Jika satu proses secara tidak sengaja dapat menulis ke dalam ruang memori dari proses lain, maka proses terakhir mungkin tidak dapat dijalankan dengan benar.

Pemisahan ruang memori dari berbagai proses diselesaikan dengan mudah dengan skema memori virtual. Baik segmentasi atau paging, atau keduanya dalam kombinasi, memberikan cara yang efektif untuk mengelola memori utama. Jika isolasi lengkap dicari, maka OS harus memastikan bahwa setiap segmen atau halaman hanya dapat diakses melalui proses yang ditetapkan. Ini dilakukan dengan mudah dengan mengharuskan tidak ada entri duplikat di tabel segmen dan/atau halaman.

Jika *sharing* diizinkan, segmen atau halaman yang sama dapat muncul di lebih dari satu tabel. Jenis berbagi ini dilakukan paling mudah dalam sistem yang mendukung segmentasi atau kombinasi segmentasi dan *paging*. Dalam kasus ini, struktur segmen dapat dilihat oleh aplikasi, dan aplikasi dapat mendeklarasikan segmen individu sebagai yang dapat dibagikan atau tidak dapat dibagikan. Dalam lingkungan *paging* murni, menjadi lebih sulit untuk membedakan antara dua jenis memori, karena struktur memori transparan terhadap aplikasi.

Segmentasi, khususnya, cocok untuk implementasi kebijakan perlindungan dan pembagian. Karena setiap entri tabel segmen menyertakan panjang dan juga alamat dasar, program tidak dapat secara tidak sengaja mengakses lokasi memori utama di luar batas segmen. Untuk mencapai *sharing*, segmen dapat direferensikan dalam tabel segmen lebih dari satu proses. Mekanisme yang sama tersedia dalam sistem *paging*. Namun, dalam hal ini struktur halaman program dan data tidak terlihat oleh programmer, membuat spesifikasi perlindungan dan persyaratan berbagi menjadi lebih canggung.

b. Kontrol Akses Berorientasi Pengguna

Langkah-langkah yang diambil untuk mengontrol akses dalam sistem pemrosesan data terbagi dalam dua kategori: yang terkait dengan pengguna dan yang terkait dengan data.

Teknik yang paling umum untuk kontrol akses pengguna pada sistem atau server bersama adalah logon pengguna, yang memerlukan pengenal pengguna (ID) dan beberapa bentuk otentikasi, seperti memberikan sandi, token, atau atribut biometrik. Otentikasi mengacu pada pengikatan identitas dunia nyata (misalnya, karyawan bernama atau peran bernama dalam organisasi) dan ID yang ditunjukan. Sistem akan mengizinkan pengguna

untuk masuk hanya jika ID pengguna tersebut diketahui oleh sistem dan jika pengguna mengetahui kata sandi yang dikaitkan oleh sistem dengan ID tersebut.

Setelah pengguna membuat sesi, sistem operasi kemudian dapat mengotorisasi berbagai bentuk akses (misalnya: membaca, menulis, menambahkan, mengunci, atau mengeksekusi) ke berbagai jenis data (misalnya: file, database, perangkat, atau komunikasi tertentu).

Kontrol akses pengguna dalam lingkup terdistribusi dapat terpusat atau terdesentralisasi. Dalam pendekatan terpusat, jaringan menyediakan layanan logon, menentukan siapa yang diizinkan untuk menggunakan jaringan dan kepada siapa pengguna diizinkan untuk terhubung.

Kontrol akses pengguna yang terdesentralisasi memperlakukan jaringan sebagai tautan komunikasi transparan, dan host tujuan melakukan prosedur logon biasa. Masalah keamanan untuk mengirimkan kata sandi melalui jaringan harus tetap diatasi.

Di banyak jaringan, dua tingkat kontrol akses dapat digunakan. Masing-masing host dapat diberikan fasilitas logon untuk melindungi sumber daya dan aplikasi khusus host. Selain itu, jaringan secara keseluruhan dapat memberikan perlindungan untuk membatasi akses jaringan kepada pengguna yang berwenang. Fasilitas dua tingkat ini diinginkan untuk kasus umum, saat ini, di mana jaringan menghubungkan host yang berbeda dan hanya menyediakan cara yang nyaman untuk akses terminal-host. Dalam jaringan host yang lebih seragam, beberapa kebijakan akses terpusat dapat diterapkan di pusat kendali jaringan.

c. Kontrol Akses Berorientasi Data

Setelah berhasil masuk, pengguna diberikan akses ke satu atau sekumpulan host dan aplikasi. Ini umumnya tidak cukup untuk sistem yang menyertakan data sensitif dalam database-nya. Melalui prosedur kontrol akses pengguna, pengguna dapat diidentifikasi ke sistem. Terkait dengan setiap pengguna, mungkin ada profil yang menentukan operasi yang diizinkan dan akses file. OS kemudian dapat memberlakukan aturan berdasarkan profil pengguna. Sistem manajemen database, bagaimanapun, harus mengontrol akses ke catatan tertentu atau bahkan bagian dari catatan. Misalnya,

seseorang dalam administrasi mungkin diizinkan untuk mendapatkan daftar personel perusahaan, tetapi hanya individu terpilih yang dapat memiliki akses ke informasi gaji. Masalahnya lebih dari satu tingkat detail. Sementara OS dapat memberikan izin kepada pengguna untuk mengakses file atau menggunakan aplikasi, setelah itu tidak ada pemeriksaan keamanan lebih lanjut, sistem manajemen database harus membuat keputusan pada setiap upaya akses individu. Keputusan itu tidak hanya akan bergantung pada identitas pengguna tetapi juga pada bagian tertentu dari data yang diakses dan bahkan pada informasi yang telah diungkapkan kepada pengguna.

d. Perlindungan berdasarkan Mode Sistem Operasi

Salah satu teknik yang digunakan di semua OS untuk memberikan perlindungan didasarkan pada mode eksekusi prosesor. Kebanyakan prosesor mendukung setidaknya dua mode eksekusi: mode yang biasanya terkait dengan OS dan yang biasanya terkait dengan program pengguna. Instruksi tertentu hanya dapat dijalankan dalam mode yang lebih diistimewakan. Ini termasuk membaca atau mengubah register kontrol, seperti kata status program; instruksi I / O primitif; dan instruksi yang berhubungan dengan manajemen memori. Selain itu, wilayah memori tertentu hanya dapat diakses dalam mode yang lebih diistimewakan.

Mode yang kurang memiliki hak istimewa sering disebut sebagai mode pengguna, karena program pengguna biasanya akan dijalankan dalam mode ini. Mode yang lebih diistimewakan disebut sebagai mode sistem, mode kontrol, atau mode kernel. Istilah terakhir ini mengacu pada kernel OS, yang merupakan bagian dari OS yang mencakup fungsi sistem penting. Daftar berikut mencantumkan fungsi-fungsi yang biasanya ditemukan di kernel OS.

1) Manajemen proses
   a) Proses *creation* dan *termination*
   b) Proses *scheduling* dan *dispatching*
   c) Proses *switching*
   d) Proses sinkronisasi dan dukungan untuk komunikasi antar proses
   e) Manajemen proses blok control
2) Manajemen memori
   a) Alokasi ruang alamat ke proses
   b) Swapping

c) Manajemen halaman dan segmen

3) Manajemen I/O
   a) Manajemen penyangga (*buffer*)
   b) Alokasi saluran dan perangkat I/O ke proses
4) Fungsi pendukung
   a) Penanganan interupsi
   b) Accounting
   c) Monitoring

Alasan menggunakan dua mode harus jelas. Hal ini diperlukan untuk melindungi OS dan tabel OS utama, seperti blok kontrol proses, dari gangguan oleh program pengguna. Dalam mode kernel, perangkat lunak memiliki kendali penuh atas prosesor dan semua instruksi, register, dan memori. Tingkat kendali ini tidak diperlukan, dan untuk keamanan tidak diinginkan, untuk program pengguna.

e. Perlindungan berdasarkan Virtualisasi

Dengan ketersediaan memori yang terus meningkat (misalnya, puluhan hingga ratusan gigabyte RAM), ruang disk (penyimpanan terabyte hingga petabyte), prosesor yang lebih cepat (puluhan gigahertz), dan sistem multicore (berpotensi ribuan prosesor bekerja secara paralel), virtualisasi komputer telah berkembang menjadi kegunaan praktis dan luas. Instansiasi lingkungan operasi dapat hidup berdampingan menggunakan sumber daya bersama tanpa mengizinkan komunikasi langsung di antara mereka. Setiap instansiasi dikemas dan dilindungi sepenuhnya dari gangguan atau gangguan dari proses yang berjalan pada mesin virtual lain yang berbagi sumber daya fisik yang sama.

f. File Sharing

Sistem multipengguna hampir selalu mengharuskan file dapat dibagikan di antara sejumlah pengguna. Dua masalah muncul: hak akses dan pengelolaan akses simultan.

1) Hak Akses

Sistem file harus menyediakan alat yang fleksibel untuk memungkinkan berbagi file secara ekstensif di antara pengguna. Sistem file harus menyediakan sejumlah opsi sehingga cara mengakses file

tertentu dapat dikontrol. Biasanya, pengguna atau kelompok pengguna diberikan hak akses tertentu ke file. Berbagai hak akses telah digunakan. Daftar berikut menunjukkan hak akses yang dapat diberikan kepada pengguna tertentu untuk file tertentu.

a) None. Pengguna bahkan mungkin tidak mengetahui keberadaan file tersebut, apalagi mengaksesnya. Untuk menegakkan batasan ini, pengguna tidak akan diizinkan untuk membaca direktori pengguna yang menyertakan file ini.
b) Knowledge. Pengguna dapat menentukan bahwa file tersebut ada dan siapa pemiliknya. Pengguna kemudian dapat mengajukan petisi kepada pemilik untuk mendapatkan hak akses tambahan.
c) Execution. Pengguna dapat memuat dan menjalankan program tetapi tidak dapat menyalinnya. Program berpemilik sering kali dapat diakses dengan pembatasan ini.
d) Locking. Pengguna dapat mengubah status dari tanda logical yang menunjukkan pembatasan sementara pada akses ke data. Sistem manajemen database menyediakan penguncian untuk mengontrol akses bersamaan ke catatan sehingga proses yang berbeda dapat menghindari penimpaan modifikasi satu sama lain.
e) Reading. Pengguna dapat membaca file untuk tujuan apa pun, termasuk menyalin dan mengeksekusi. Beberapa sistem dapat menerapkan perbedaan antara melihat dan menyalin. Dalam kasus sebelumnya, konten file dapat ditampilkan kepada pengguna, tetapi pengguna tidak memiliki sarana untuk membuat salinan.
f) Appending. Pengguna dapat menambahkan data ke file, seringkali hanya di bagian akhir, tetapi tidak dapat mengubah atau menghapus konten file apa pun. Hak ini berguna untuk mengumpulkan data dari berbagai sumber.
g) Updating. Pengguna dapat mengubah, menghapus, dan menambahkan ke data file. Ini biasanya termasuk menulis file pada awalnya, menulis ulang seluruhnya atau sebagian, dan menghapus semua atau sebagian data. Beberapa sistem membedakan berbagai tingkat pembaruan.
h) Changing protection. Pengguna dapat mengubah hak akses yang diberikan kepada pengguna lain. Biasanya, hanya pemilik file yang

memegang hak ini. Dalam beberapa sistem, pemilik dapat memberikan hak ini kepada orang lain. Untuk mencegah penyalahgunaan mekanisme ini, pemilik file biasanya dapat menentukan hak mana yang dapat diubah oleh pemegang hak yang diperpanjang ini.

i) Deletion. Pengguna dapat menghapus file dari sistem file.

Hak-hak ini dapat dianggap sebagai suatu hierarki, dengan masing-masing hak mengartikan hak yang mendahuluinya. Jadi, jika pengguna tertentu diberikan hak memperbarui untuk file tertentu, maka pengguna tersebut juga diberikan hak-hak ini: pengetahuan, eksekusi, membaca, dan menambahkan.

Satu pengguna ditetapkan sebagai pemilik file tertentu, biasanya ini adalah orang yang pertama kali membuat file tersebut. Pemilik memiliki semua hak akses yang terdaftar sebelumnya dan dapat memberikan hak kepada orang lain. Akses dapat diberikan ke berbagai kelas pengguna:

a) Specific user. Pengguna individu yang ditentukan oleh ID pengguna.
b) User groups. Sekumpulan pengguna yang tidak ditentukan satu per satu. Sistem harus memiliki beberapa cara untuk melacak keanggotaan grup pengguna.
c) All. Semua pengguna yang memiliki akses ke sistem ini. Ini adalah file publik.

2) Akses Simultan

Ketika akses diberikan untuk menambahkan atau memperbarui file ke lebih dari satu pengguna, OS atau sistem manajemen file harus menegakkan disiplin. Pendekatan brute force adalah mengizinkan pengguna mengunci seluruh file saat akan diperbarui. Butir kontrol yang lebih baik adalah mengunci catatan individu selama pembaruan. Masalah eksklusi bersama dan kebuntuan harus ditangani dalam mendesain kapabilitas akses bersama.

g. File Descriptor

Agar proses dapat bekerja dengan file, mereka memerlukan cara singkat untuk merujuk ke file tersebut, selain selalu menuju sistem file dan menentukan jalur ke file tersebut. Untuk membaca dan menulis file yang

disimpan di disk secara efisien, sistem operasi modern mengandalkan mekanisme yang dikenal sebagai deskriptor file. Deskriptor file pada dasarnya adalah nilai indeks yang disimpan dalam tabel, yang dikenal sebagai tabel deskriptor file. Saat program perlu mengakses file, panggilan dilakukan ke panggilan sistem terbuka, yang mengakibatkan kernel membuat entri baru dalam tabel deskriptor file yang memetakan ke lokasi file di disk. Deskriptor file baru ini dikembalikan ke program, yang sekarang dapat mengeluarkan perintah *read* atau *write* menggunakan deskriptor file tersebut. Saat menerima panggilan sistem *read* atau *write*, kernel mencari deskriptor file dalam tabel dan melakukan *read* atau *write* di lokasi yang sesuai pada disk. Akhirnya, setelah selesai, program harus mengeluarkan panggilan sistem tertutup untuk menghapus deskriptor file yang terbuka.

4. Memahami Keamanan Program Aplikasi

Setelah sistem operasi dasar diinstal dan diamankan dengan benar, layanan dan aplikasi yang diperlukan selanjutnya harus diinstal dan dikonfigurasi. Persoalannya, seperti dengan sistem operasi dasar, adalah hanya menginstal perangkat lunak pada sistem yang diperlukan untuk memenuhi fungsionalitas yang diinginkan, untuk mengurangi jumlah tempat kerentanan dapat ditemukan. Pada sistem klien, perangkat lunak seperti Java, PDF viewer, Flash, browser Web, dan Microsoft Office merupakan target yang diketahui dan perlu diamankan. Pada sistem server, perangkat lunak yang menyediakan akses atau layanan jarak jauh, termasuk Web, basis data, dan server akses file, menjadi perhatian khusus, karena penyerang mungkin dapat memanfaatkan ini untuk mendapatkan akses jarak jauh ke sistem.

Setiap layanan atau aplikasi yang dipilih harus diinstal, dikonfigurasi, dan kemudian ditambah ke versi aman yang didukung terbaru yang sesuai untuk sistem. Ini mungkin dari paket tambahan yang disediakan dengan distribusi sistem operasi, atau dari paket pihak ketiga yang terpisah. Seperti sistem operasi dasar, lebih disukai menggunakan jaringan build yang terisolasi dan aman.

a. Konfigurasi Program Aplikasi

Konfigurasi khusus aplikasi apa pun kemudian dilakukan. Ini mungkin termasuk membuat dan menentukan area penyimpanan data yang sesuai

untuk aplikasi, dan membuat perubahan yang sesuai pada detail konfigurasi default aplikasi atau layanan.

Beberapa aplikasi atau layanan mungkin menyertakan data default, skrip, atau akun pengguna. Ini harus ditinjau, dan hanya disimpan jika diperlukan, dan diamankan dengan sesuai. Contoh terkenal dari hal ini ditemukan dengan server Web, yang sering kali menyertakan sejumlah contoh skrip, beberapa di antaranya diketahui tidak aman. Ini tidak boleh digunakan seperti yang disediakan, tetapi harus dilepas kecuali diperlukan dan diamankan.

Sebagai bagian dari proses konfigurasi, pertimbangan yang cermat harus diberikan pada hak akses yang diberikan ke aplikasi. Sekali lagi, ini menjadi perhatian khusus dengan layanan yang diakses dari jarak jauh, seperti Web dan layanan transfer file. Aplikasi server tidak boleh diberikan hak untuk mengubah file, kecuali fungsi itu secara khusus diperlukan. Kesalahan konfigurasi yang sangat umum terlihat pada Web dan server transfer file adalah semua file yang disediakan oleh layanan dimiliki oleh akun "pengguna" yang sama dengan yang dijalankan oleh server. Konsekuensinya adalah bahwa setiap penyerang yang dapat mengeksploitasi beberapa kerentanan baik di perangkat lunak server atau skrip yang dijalankan oleh server mungkin dapat memodifikasi file-file ini. Sejumlah besar serangan "perusakan web" adalah bukti nyata dari jenis konfigurasi yang tidak aman ini. Sebagian besar risiko dari bentuk serangan ini dikurangi dengan memastikan bahwa sebagian besar file hanya dapat dibaca, tetapi tidak ditulis, oleh server. Hanya file yang perlu dimodifikasi, untuk menyimpan data formulir yang diunggah misalnya, atau detail pencatatan, harus dapat ditulisi oleh server. Sebaliknya, sebagian besar file harus dimiliki dan dimodifikasi oleh pengguna di sistem yang bertanggung jawab untuk memelihara informasi.

b. Teknologi Enkripsi

Jika layanan jaringan aman disediakan, kemungkinan besar menggunakan TLS atau IPsec, maka kunci publik dan pribadi yang sesuai harus dibuat untuk masing-masing. Kemudian sertifikat X.509 dibuat dan ditandatangani oleh otoritas sertifikat yang sesuai, yang menghubungkan setiap identitas layanan dengan kunci publik yang digunakan. Jika akses jarak jauh yang aman disediakan menggunakan Secure Shell (SSH), maka server

yang sesuai, dan mungkin kunci klien, harus dibuat. Sistem file kriptografi adalah penggunaan enkripsi lainnya. Jika diinginkan, maka ini harus dibuat dan diamankan dengan kunci yang sesuai.

C. SOAL LATIHAN/ TUGAS

1. Jelaskan konsep keamanan sistem operasi yang Anda ketahui!
2. Bagaimana tindakan dasar perlindungan sistem operasi?
3. Jelaskan tindakan perlindungan kontrol akses pada sistem operasi!
4. Jelaskan keamanan program aplikasi pada sistem operasi!
5. Sebagai seorang staff IT dalam sebuah perusahaan komputer Anda rencananya akan digunakan oleh beberapa user, apakah yang harus ada lakukan agar keamanan komputer Anda dalam sistem operasi dapat terkendali dengan baik!

D. REFERENSI

Nestler, V., Harrison, K., Hirsch, M., & Conklin, W. A. (2015). *Principles of Computer Security Lab Manual*. 4th Edition. New York: McGraw-Hill Education.

Panek, C. (2020). *Security Fundamental*. Canada: Sybex A Wiley Brand.

Gollmann, D. (2011). *Computer Security*. 3rd Edition. India: John Wiley & Sons, Ltd.

Goodrich, M., & Tamassia, R. (2014). *Introduction to Computer Security*. Harlow: Pearson Education Ltd.

Pfleeger, P. C., Pfleeger, S. L., & Margulies, J. (2015). *Security in Computing*. 5th Edition. Westford: Pearson Education Inc.

Salomon, D. (2006). *Foundations of Computer Security*. Springer Science+Business Media.

Stallings, W., & Brown, L. (2018). *Computer Security Principles and Practices*. 4th Edition. New York: Pearson Education Limited.

Bosworth, S., Kabay, M. E., & Whyne, E. (2014). *Computer Security Handbook*. 6th Edition. Canada: John Wiley & Sons, Inc.

Scarfone, K., Jansen, W., & Tracy, M. (2008). National Institute of Standards and Technology [Internet]. Juli 2008. NIST Pubs. Tersedia pada: https://csrc.nist.gov/publications/detail/sp/800-123/final

Paulsen, C., & Byers, R. D. *Glossary of Key Information Security Terms* [Internet]. Juli 2019. NIST Pubs. Tersedia pada: https://www.nist.gov/publications/glossary-key-information-security-terms-2, https://csrc.nist.gov/glossary