px

Segurança em Sistemas Operacionais

Caio Mariano, Gustavo Santos
e Ygor Henrique

Instituto Federal de Brasília, Campus
Taguatinga

Sumário

px

Motivação

- Muitas vezes informações valiosas são guardadas em computadores.
- Essas informações variam de tipo dependendo do ente que o está a guardando(empresas ou usuários domésticos).
- Em virtude do valor dessas informações, a segurança em sistemas operacionais se fez necessária.

Ambiente de Segurança

- Muitos textos de segurança decompõem a segurança de um sistema de informação em três componentes:
  1. confidencialidade
  2. integridade
  3. disponibilidade
- "CIA" (confidentiality, integrity, availability).

Componentes de segurança - Confidencialidade

- Diz respeito a fazer com que dados secretos assim permaneçam.
- O sistema deve garantir que a liberação de dados para pessoas não autorizadas jamais ocorra.

Componentes de segurança - Integridade

- Usuários não autorizados não devem ser capazes de modificar dado algum sem a permissão do proprietário.
- O sistema tem que garantir que os dados depositados nele permaneçam inalterados até que o proprietário decida modificá-los.

Componentes de segurança - Disponibilidade

- Ninguém pode perturbar o sistema para torná-lo inutilizável.
- Denial-of-service.

Ataques a Sistemas Operacionais

- Existe uma variedade enorme de maneiras pelas quais atacantes podem comprometer a máquina de um usuário, como:
  1. Backdoor
  2. Ataque DoS
  3. Ataque DMA
  4. Eavesdropping
  5. e muitos outros..
- Botnets e atividades criminosas na internet.

Ataques a Sistemas Operacionais

- Quando o computador está sob o controle do atacante, ele é conhecido com um bot ou zumbi.
- Tipicamente, nada disso é visível para o usuário.

Segurança em Sistemas Operacionais

- Existem dois tipos de ataques que tentam roubar informações, que são:
  1. Ataques Passivos.
  2. Ataques Ativos.

Segurança em Sistemas Operacionais

- As duas principais formas de proteção que existem são:
  1. A criptografia
  2. O endurecimento de software.

px

É possível construir um SO seguro?

- Sim, é possível, porem quanto mais complexo, mais difícil é de manter o SO seguro.
- As diferentes funcionalidades de um SO geram mais complexidade, mais códigos, mais defeitos e mais erros de segurança.

Controlando o acesso aos recursos

- Um sistema computacional contém muitos recursos, ou "objetos", que precisam ser protegidos.
- Esses objetos podem ser hardware ou software.
- Cada objeto tem um nome único pelo qual ele é referenciado, assim como um conjunto finito de operações que os processos têm permissão de executar.

Domínio

- Um domínio é um conjunto de pares (objetos, direitos).
- Cada par especifica um objeto e algum subconjunto das operações que podem ser desempenhadas nele.
- Um direito nesse contexto significa a permissão para desempenhar uma das operações.

Domínio

- Como objetos são alocados para domínios depende das questões específicas relativas a quem precisa saber o quê
- *"POLA"* ou necessidade de saber.

px

Armamento de matrizes

- Existem dois métodos práticos para armazenar matrizes, por linhas ou por colunas, e então armazenar somente os elementos não vazios.
- O primeiro método é a lista de Controle de Acesso:

1. Para cada objeto é definida uma lista de controle de acesso (ACL - Access Control List), que contém: a relação de sujeitos que podem acessá-lo e suas respectivas permissões.
2. Cada lista de controle de acesso corresponde a uma coluna da matriz de controle de acesso.
3. É simples de implementar
4. É bastante robusta.
5. É a mais usada em sistemas operacionais.

px

Armazenamento de matrizes

- O segundo método é a Lista de Capacidades :

1. Uma lista de objetos que um dado sujeito pode acessar e suas respectivas permissões sobre os mesmos.
2. Cada lista de capacidades corresponde a uma linha da matriz de acesso.
3. Uma capacidade pode ser vista como uma ficha ou token: sua posse dá ao proprietário o direito de acesso ao objeto em questão.
4. Capacidades são pouco usadas em sistemas operacionais, devido à sua dificuldade de implementação e possibilidade de fraude.
5. Outra dificuldade inerente às listas de capacidades é a administração das autorizações.

Matrizes de Proteção

- Matrizes de proteção não são estáticas.
- O conjunto de todas as matrizes pode ser dividido no conjunto de todos os estados autorizados e o de todos os não autorizados.

Modelos formais de sistemas seguros

	Objeto							
	Arquivo1	Arquivo2	Arquivo3	Arquivo4	Arquivo5	Arquivo6	Impressora1	Plotter2
Domínio 1	Leitura	Leitura Escrita						
2			Leitura	Leitura Escrita Execução	Leitura Escrita		Escrita	
3						Leitura Escrita Execução	Escrita	Escrita

Figura: Matriz de Proteção

px

Comandos de proteção

- Existem seis operações primitivas na matriz de proteção que podem ser usadas para modelar qualquer sistema de proteção. Elas são:
  1. Create object;
  2. Delete object;
  3. Create domain;
  4. Delete domain;
  5. Insert right;
  6. Remove right;

Comandos de proteção

- São os comandos de proteção que fazem alterações na matriz.
- Eles não podem executar as primitivas diretamente.

Segurança multinível

- O modelo de segurança multinível mais amplamente usado é o modelo Bell-LaPadula.
- Esse modelo foi projetado para lidar com a segurança militar.
- Basicamente os documentos e os usuários tem um nível de segurança.
- Um processo executando em um certo nível de segurança só pode ter acesso a objetos do seu nível ou menor.

Segurança multinível

- Mesmo em um sistema com um modelo de segurança apropriado podem acontecer vazamentos de segurança por meio dos canais ocultos..
- Canal oculto é um canal de comunicação onde o servidor manda um fluxo de bits binário para o intruso.

px

Conclusão

- O sistema operacional é o software mais importante do computador e por isso deve ser o mais seguro para que não haja prejuízos, ou até mesmo perda da integridade do sistema.
- Para isso é necessário investir em uma rígida politica de segurança que detenha mecanismos atualizados para detectar e interromper as ameaças, pois os ataques virtuais mudam e se sofisticam a cada dia.

Referências

- Tanenbaum, A. S. e Bos, H.. Sistemas Operacionais Modernos. 4.ed. Pearson/Prentice-Hall. 2016.

px