Segurança e Auditoria de Sistemas

Conceitos básicos

Conceitos básicos

- Propriedades e princípios de segurança;
- Ameaças;
- Vulnerabilidades;
- Ataques;
- Tipos de malware;
- Infraestrutura de segurança.

Propriedades e princípios de segurança

Fundamentais

- **Confidencialidade** :
  - os recursos presentes no sistema só podem ser **consultados** por usuários devidamente autorizados;
- **Integridade** :
  - os recursos do sistema só podem ser **modificados ou destruídos** pelos usuários autorizados;
- **Disponibilidade** :
  - os recursos devem estar disponíveis para os usuários que tiverem direito de usa-los, a qualquer momento.

Propriedades e princípios de segurança

Complementares

- **Autenticidade** :
  - todas as entidades do sistema são autênticas ou genuínas;
- **Irretratabilidade** :
  - Todas as ações realizadas no sistema são conhecidas e não podem ser escondidas ou negadas por seus autores;
  - esta propriedade também é conhecida como *irrefutabilidade* ou *não-repudiação*.

Princípios para construção do sistema 1/4

- **Privilégio mínimo** :
  - **todos** os usuários e programas devem operar com o **mínimo** possível de privilégios ou permissões de acesso.
  - Dessa forma, os danos provocados por erros ou ações maliciosas intencionais serão minimizados.
- **Mediação completa** :
  - **todos** os acessos a recursos, tanto diretos quanto indiretos, **devem** ser verificados pelos mecanismos de segurança.
  - Eles devem estar dispostos de forma a ser impossível contorna-los.

Princípios para construção do sistema 2/4

- **Default seguro** :
  - o mecanismo de segurança deve identificar claramente os acessos permitidos;
  - caso um certo acesso não seja explicitamente permitido, ele deve ser negado.
    - Este princípio impede que acessos inicialmente não-previstos no projeto do sistema sejam inadvertidamente autorizados.
- **Economia de mecanismo** :
  - o projeto de um sistema de proteção deve ser pequeno e simples,
    - para que possa ser facilmente e profundamente analisado, testado e validado.
- **Separação de privilégios** :
  - sistemas de proteção baseados em mais de um controle são mais robustos, pois se o atacante conseguir burlar um dos controles, mesmo assim não terá acesso ao recurso.
    - Um exemplo típico é o uso de mais de uma forma de autenticação para acesso ao sistema (como um cartão e uma senha, nos sistemas bancários).

Princípios para construção do sistema 3/4

- **Compartilhamento mínimo** :
  - o uso de mecanismos compartilhados deve ser minimizado, sobretudo se envolver áreas de memória compartilhadas.
    - Ex, caso uma certa funcionalidade do sistema operacional possa ser implementada como chamada ao núcleo ou como função de biblioteca, deve-se preferir esta última forma, pois envolve menos compartilhamento.
- **Projeto aberto** :
  - a robustez do mecanismo de proteção não deve depender da ignorância dos dissatacantes;
    - ao invés o, o projeto deve ser público e aberto, dependendo somente do segredo de poucos itens, como listas de senhas ou chaves criptográficas.
- **Proteção adequada** :
  - cada recurso computacional deve ter um nível de proteção coerente com seu valor intrínseco.
    - Ex, o nível de proteção requerido em um servidor Web de serviços bancário é bem distinto daquele de um terminal público de acesso à Internet.

Princípios para construção do sistema 4/4

- **Facilidade de uso** :
  - o uso dos mecanismos de segurança deve ser fácil e intuitivo, caso contrário eles serão evitados pelos usuários.
- **Eficiência** :
  - os mecanismos de segurança devem ser eficientes no uso dos recursos computacionais, de forma a não afetar significativamente o desempenho do sistema ou as atividades de seus usuários.
- **Elo mais fraco** :
  - a segurança do sistema é limitada pela segurança de seu elemento mais vulnerável, seja ele o sistema operacional, as aplicações, a conexão de rede ou o próprio usuário.

Ameaças

- **Ameaça** ➔ pode ser considerada qualquer ação que coloque em risco as propriedades de segurança do sistema descritas na seção anterior.

- Alguns exemplos de ameaças às propriedades básicas de segurança seriam:
  - ***Ameaças à confidencialidade***: um processo vasculhar as áreas de memória de outros processos, arquivos de outros usuários, tráfego de rede nas interfaces locais ou áreas do núcleo do sistema, buscando dados sensíveis como números de cartão de crédito, senhas, e-mails privados, etc.;
  - ***Ameaças à integridade***: um processo alterar as senhas de outros usuários, instalar programas, *drivers* ou módulos de núcleo maliciosos, visando obter o controle do sistema, roubar informações ou impedir o acesso de outros usuários;
  - ***Ameaças à disponibilidade***: um usuário alocar para si todos os recursos do sistema, como a memória, o processador ou o espaço em disco, para impedir que outros usuários possam utiliza-lo.

Vulnerabilidades

- Uma vulnerabilidade é um defeito ou problema
  - presente na especificação, implementação, configuração ou operação de um software ou sistema, que possa ser explorado para violar as propriedades de segurança do mesmo.
    - Alguns exemplos de vulnerabilidades são descritos a seguir:
      - Um erro deprogramação no serviço de compartilhamento de arquivos;
      - Conta de usuário sem senha, ou com uma senha pré-definida pelo fabricante;
      - Ausência de quotas de disco, permitindo a um único usuário alocar todo o espaço em disco para si e assim impedir os demais usuários de usar o sistema.
- A grande maioria das vulnerabilidades ocorre devido a erros de programação.

Ataques

- **Ataque**
  - é o ato de utilizar (ou explorar) uma vulnerabilidade para violar uma propriedade de segurança do sistema.
  - De acordo com [Pfleeger and Pfleeger, 2006], existem basicamente quatro tipos de ataques,
- **Interrupção** :
  - consiste em impedir o fluxo normal das informações ou acessos; é um ataque à **disponibilidade** do sistema;
- **Interceptação** :
  - consiste em obter acesso indevido a um fluxo de informações, sem necessariamente modifica-las;
  - é um ataque à **confidencialidade**;
- **Modificação** :
  - consiste em modificar de forma indevida informações ou partes do sistema, violando sua **integridade**;
- **Fabricação** :
  - consiste em produzir informações falsas ou introduzir módulos ou componentes maliciosos no sistema; é um ataque à **autenticidade**.

Tipos de malware 1/3

- ***Malware*** todo programa cuja intenção é realizar atividades ilícitas,
  - como realizar ataques, roubar informações ou dissimular a presença de intrusos em um sistema.

Existe uma grande diversidade de *malwares*, destinados às mais diversas finalidades [Shirey, 2000, Pfleeger and Pfleeger, 2006], como:

- **Vírus** : um vírus de computador é um trecho de código que se infiltra em programas executáveis existentes no sistema operacional, usando-os como suporte para sua execução e replicação.
  - Alguns tipos de vírus são programados usando macros de aplicações complexas, como editores de texto, e usam os arquivos de dados dessas aplicações como suporte. Outros tipos de vírus usam o código de inicialização dos discos e outras mídias como suporte de execução.

- **Worm** : é um programa autônomo, que se propaga sem infectar outros programas.
  - A maioria dos vermes se propaga explorando vulnerabilidades nos serviços de rede, que os permitam invadir e instalar-se em sistemas remotos.
  - Alguns vermes usam o sistema de e-mail como vetor de propagação, enquanto outros usam mecanismos de auto-execução de mídias removíveis (como *pendrives*) como mecanismo de propagação. Uma vez instalado em um sistema, o verme pode instalar *spywares* ou outros programas nocivos.

Tipos de malware

- **Trojan horse** :
  - é um programa com duas funcionalidades: uma funcionalidade lícita conhecida de seu usuário e outra ilícita, executada sem que o usuário a perceba.
  - Muitos cavalos de Tróia são usados como vetores para a instalação de outros *malwares*.
    - Um exemplo clássico é o famoso *Happy New Year 99*, distribuído através de e-mails, que usava uma animação de fogos de artifício como fachada para a propagação de um verme.
  - Para convencer o usuário a executar o cavalo de Tróia podem ser usadas técnicas de *engenharia social* [Mitnick and Simon, 2002].

- **Exploit** :
  - é um programa escrito para explorar vulnerabilidades conhecidas, como prova de conceito ou como parte de um ataque.
  - Os *exploits* podem estar incorporados a outros *malwares* (como vermes e *trojans*) ou constituírem ferramentas autônomas, usadas em ataques manuais.

- **Packet sniffer** :
  - um "farejador de pacotes" captura pacotes de rede do próprio computador ou da rede local, analisando-os em busca de informações sensíveis como senhas e dados bancários.

Tipos de malware

- **Keylogger** :
  - software dedicado a capturar e analisar as informações **digitadas** pelo usuário na máquina local, sem seu conhecimento.

- **Rootkit** :
  - é um conjunto de programas destinado a ocultar a presença de um intruso no sistema operacional.
  - Como princípio de funcionamento, o *rootkit* modifica os mecanismos do sistema operacional que mostram os processos em execução, arquivos nos discos, portas e conexões de rede, etc., para ocultar o intruso.
  - Versões mais elaboradas de *rootkits* substituem bibliotecas do sistema operacional ou modificam partes do próprio núcleo, o que torna complexa sua detecção e remoção.

- **Backdoor** :
  - uma "porta dos fundos" é um programa que facilita a entrada posterior do atacante em um sistema ja invadido.
  - Geralmente a porta dos fundos é criada através um processo servidor de conexões remotas (usando SSH, telnet ou um protocolo ad-hoc).
  - Muitos *backdoors* são instalados a partir de *trojans*, vermes ou *rootkits*.

Infraestrutura de segurança

O sistema operacional emprega várias técnicas complementares para garantir a segurança de um sistema operacional. Essas técnicas estão classificadas nas seguintes grandes áreas:

- **Autenticação** :
  - conjunto de técnicas usadas para identificar inequivocamente usuários e recursos em um sistema;
    - podem ir de simples pares *login/senha* até esquemas sofisticados de biometria ou certificados criptográficos.
- **Controle de acesso** :
  - técnicas usadas para definir quais ações são permitidas e quais são negadas no sistema;
  - para cada usuário do sistema, devem ser definidas regras descrevendo as ações que este pode realizar no sistema, ou seja, que recursos este pode acessar e sob que condições.
  - Normalmente, essas regras são definidas através de uma ***política de controle de acesso***, que é imposta a todos os acessos que os usuários efetuam sobre os recursos do sistema.
- **Auditoria** :
  - técnicas usadas para manter um registro das atividades efetuadas no sistema, visando a contabilização de uso dos recursos, a análise posterior de situações de uso indevido ou a identificação de comportamento suspeitos.