Segurança

Referência:

Silberschatz, Abraham. Sistemas Operacionais com Java. 7 ed. Rio de Janeiro: Elsevier, 2008.

- **O problema da segurança**
- **Ameaças ao programa**
- **Ameaças ao sistema e à rede**
- **Criptografia como uma ferramenta de segurança**
- **Autenticação do usuário**
- **Implementando defesas de segurança**
- **Uso de firewalls para proteger sistemas e redes**
- **Classificações de segurança de computador**

O problema da segurança

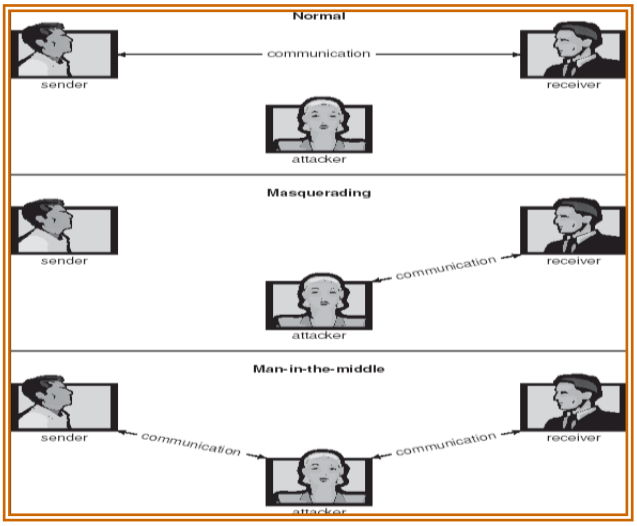
- **A segurança deve considerar o ambiente externo do sistema e proteger os recursos do sistema**
- **Intrusos (crackers) tentam quebrar a segurança**
- **Ameaça é violação de segurança em potencial**
- **Ataque é a tentativa de quebrar a segurança**
- **Ataque pode ser acidental ou malicioso**
- **Mais fácil de proteger contra mau uso acidental do que malicioso**

Violações de segurança

- **Categorias**
  - **Quebra de confidencialidade**
  - **Quebra de integridade**
  - **Quebra de disponibilidade**
  - **Ameaça de serviço**
  - **Negação de serviço**
- **Métodos**
  - **Mascaragem (quebra de autenticação)**

- **Ataque de reprodução**
  - **Modificação da mensagem**
- **Ataque do homem no meio**
- **Sequestro de sessão**

Ataques de segurança padrão



Níveis de medida de segurança

- **A segurança deve ocorrer nos quatros níveis para ser eficaz:**
  - **Físico**
  - **Humano**
    - **Evite engenharia social, phishing, dumpster diving**
  - **Sistema operacional**
  - **Rede**
- **A segurança é tão fraca quanto seu elo mais fraco**

Ameaças ao programa

- **Cavalo de Tróia**
  - **Segmento de código que faz mau uso de seu ambiente**
  - **Explora mecanismos para permitir que os programas escritos pelos usuários sejam executados por outros usuários**
  - **Spyware, janelas popup do navegador, canais cobertos**
- **Porta de armadilha**

- **Identificador ou senha de usuário específico, que contorna os procedimentos de segurança normais**
- **Poderia estar incluída em um compilador**

- **Bomba lógica**
  - **Programa que inicia um incidente de segurança sob certas circunstâncias**
- **Estouro de pilha e buffer**
  - **Explora um bug em um programa (estoura a pilha ou buffers de memória)**

Ameaças ao programa

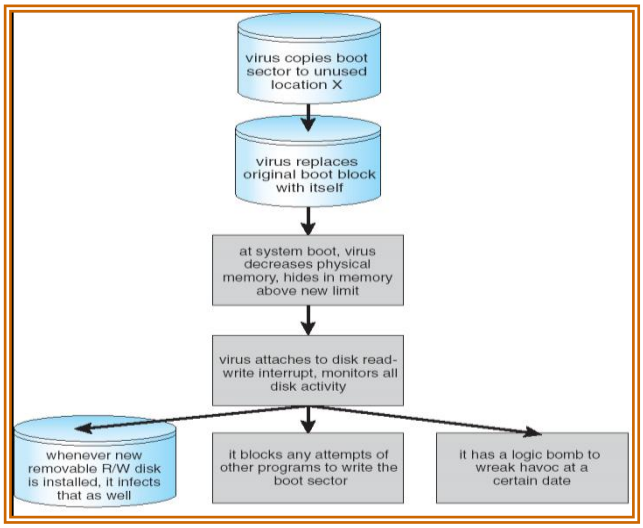
- **Vírus**
  - **Fragmento de código embutido no programa legítimo**
  - **Muito específico à arquitetura de CPU, sistema operacional, aplicações**
  - **Normalmente, vem de e-mail ou como uma macro**
    - **Macro do Visual Basic para reformatar disco rígido**

```
Sub AutoOpen()
Dim oFS
Set oFS = CreateObject(''Scripting.FileSystemObject'')
vs = Shell(''c:command.com /k format c:'',vbHide)
End Sub
```

- **Colocador de vírus insere vírus no sistema**
- **Muitas categorias de vírus, literalmente muitos milhares de vírus**
  - **Arquivo**
  - **Boot**
  - **Macro**
  - **Códiogo fonte**
  - **Polimórfico**
  - **Criptografado**
  - **Furtivo**
  - **Tunelamento**
  - **Multipartite**

- **Blindado**

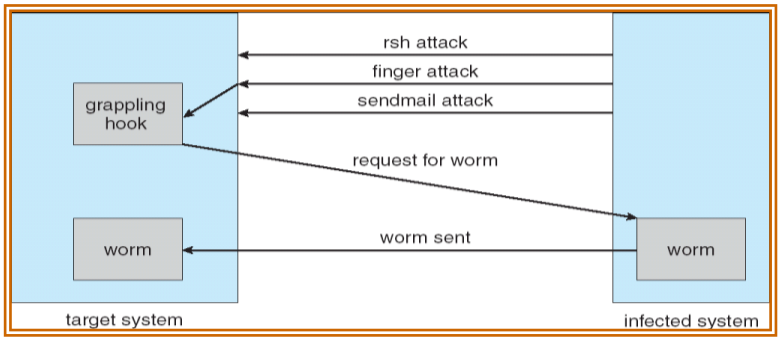
Vírus de computador do setor de boot



Ameaças ao sistema e à rede

- **Vermes – usa mecanismo de procriação; programa isolado**
- **Verme da Internet**
  - **Explora recursos de rede do UNIX (acesso remoto) e bugs nos programas *finger* e *sendmail***
  - **Programa gancho de atracação faz o upload do programa de verme principal**
- **Varredura de porta**
  - **Tentativa automatizada de conectar a um grupo de portas em um ou vários endereços IP**
- **Negação de serviço**
  - **Sobrecarrega o computador vítima, impedindo que realize qualquer trabalho útil**
  - **Negação de serviço distribuída (DDOS) vem de vários locais ao mesmo tempo**

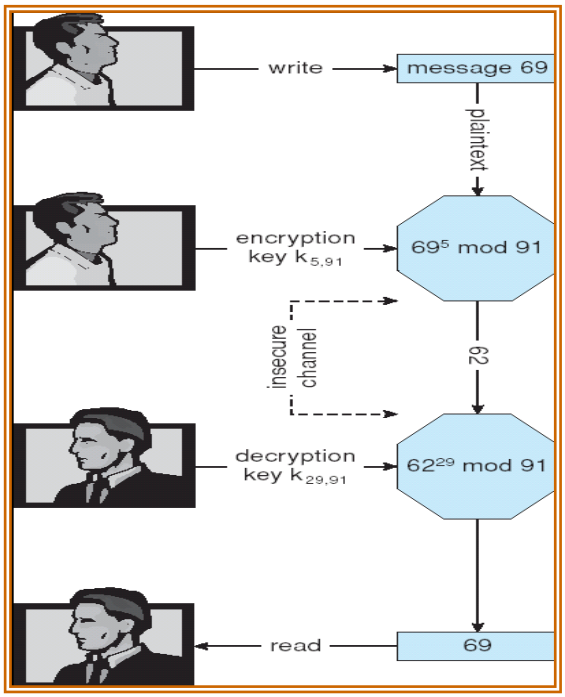
O verme Morris da Internet



Criptografia como ferramenta de segurança

- **Principal ferramenta de segurança disponível**
  - **Origem e destino das mensagens não podem ser confiados sem criptografia**
  - **Meios de restringir emissores (*origens*) e/ou receptores (*destinos*) em potencial das *mensagens***
- **Baseado em segredos (chaves)**

Criptografia e decriptografia usando criptografia assimétrica RSA



Autenticação

- **Conjunto restritivo de emissores em potencial de uma mensagem**
  - **Complementar e às vezes redundante à codificação**
  - **Também pode provar mensagem não modificada**
- **Componentes do algoritmo**
  - **Um conjunto *K* de chaves**
  - **Um conjunto *M* de mensagens**
  - **Um conjunto *A* de autenticadores**
  - **Uma função $S : K \rightarrow (M \rightarrow A)$**
    - **Ou seja, para cada $k \in K$, $S(k)$ é uma função para gerar autenticadores de mensagens**
    - **Tanto *S* quanto $S(k)$ para qualquer *k* devem ser funções calculáveis de modo eficiente**
  - **Uma função $V : K \rightarrow (M \times A \rightarrow \{true, false\})$. Ou seja, para cada $k \in K$, $V(k)$ é uma função para verificar autenticadores em mensagens**
  - **Tanto *V* quanto $V(k)$ para qualquer *k* devem ser funções calculáveis de modo eficiente**
- **Para uma mensagem *m*, um computador pode gerar um autenticador $a \in A$ tal que $V(k)(m, a) = true$ somente se possuir $S(k)$**
- **Assim, o computador mantendo $S(k)$ pode gerar autenticadores em mensagens de modo que qualquer outro computador possuindo $V(k)$ possa verificá-las**
- **O computador que não mantém $S(k)$ não pode gerar autenticadores em mensagens que possam ser verificadas usando $V(k)$**
- **Como os autenticadores geralmente são expostos (por exemplo, são enviados na rede com as próprias mensagens), não deverá ser viável derivar $S(k)$ a partir dos autenticadores**

Autenticação – Funções de hash

- **Base de autenticação**
- **Cria bloco de dados pequeno, de tamanho fixo (síntese de mensagem, valor de hash) a partir de *m***

- **Função de hash *H* deve ser resistente a colisão em *m***
  - **Deve ser inviável achar um *m' ≠ m* tal que *H*(*m*) = *H*(*m'*)**
- **Se *H*(*m*) = *H*(*m'*), então *m* = *m*'**
  - **A mensagem não foi modificada**
- **Funções comuns de síntese de mensagem são MD5, que produz um hash de 128 bits, e SHA-1, que gera um hash de 160 bits**

Autenticação – MAC

- **Codificação simétrica usada no algoritmo de autenticação Message-Authentication Code (MAC)**
- **Exemplo simples:**
  - **MAC define *S*(*k*)(*m*) = *f* (*k, H*(*m*))**
    - **Onde *f* é uma função que é unidirecional no seu primeiro argumento**
      - ***k* não pode ser derivado de *f* (*k, H*(*m*))**
    - **Devido a resistência à colisão na função de hash, razoavelmente garantido que nenhuma outra mensagem poderia criar o mesmo MAC**
    - **Um algoritmo de verificação adequado é *V*(*k*)(*m, a*) ≡ (*f* (*k,m*) = *a*)**
    - **Note que *k* é necessário para calcular *S*(*k*) e *V*(*k*), e quem puder calcular um, poderá calcular o outro**

Autenticação – Assinatura digital

- **Baseada em chaves assimétricas e algoritmo de assinatura digital**
- **Autenticadores produzidos são assinaturas digitais**
- **Em um algoritmo de assinatura digital, computacionalmente inviável derivar $S(k_s)$ de $V(k_v)$**
  - ***V* é uma função unidirecional**
  - **Assim, k_v é a chave pública e k_s é a chave privada**
- **Considere o algoritmo de assinatura digital RSA**
  - **Semelhante ao algoritmo de codificação RSA, mas o uso da chave é invertido**

- **Assinatura digital da mensagem $S(k_s)(m) = H(m)^{ks} \bmod N$**
- **A chave k_s novamente é um par d, N, onde N é o produto de dois números primos grandes, p e q, escolhidos aleatoriamente**
- **Algoritmo de verificação é $V(k_v)(m, a) \equiv (a^{kv} \bmod N = H(m))$**
  - **onde k_v satisfaz $k_v k_s \bmod (p - 1)(q - 1) = 1$**

Certificados digitais

- **Prova de quem ou o que possui uma chave pública**
- **Chave pública assinada digitalmente por uma parte confiável**
- **Parte confiável recebe prova de identificação da entidade e certifica que a chave pública pertence à entidade**
- **Autoridade de certificação é uma parte confiável – suas chaves públicas incluídas com distribuições de navegador Web**
  - **Elas respondem por outras autoridades assinando digitalmente suas chaves, e assim por diante**

Autenticação do usuário

- **Crucial para identificar o usuário corretamente, pois sistemas de proteção depende da ID do usuário**
- **Identidade do usuário normalmente estabelecida por *senhas*, pode ser considerada um caso especial de chaves ou capacidades**
  - **Também pode incluir algo que o usuário tenha e/ou um atributo do usuário**
- **Senhas devem ser mantidas secretas**
  - **Mudança freqüente de senhas**
  - **Uso de senhas "não-adivinháveis"**
  - **Log de todas as tentativas de acesso inválidas**
- **Senhas também podem ser codificadas ou ter permissão para serem usadas apenas uma vez**

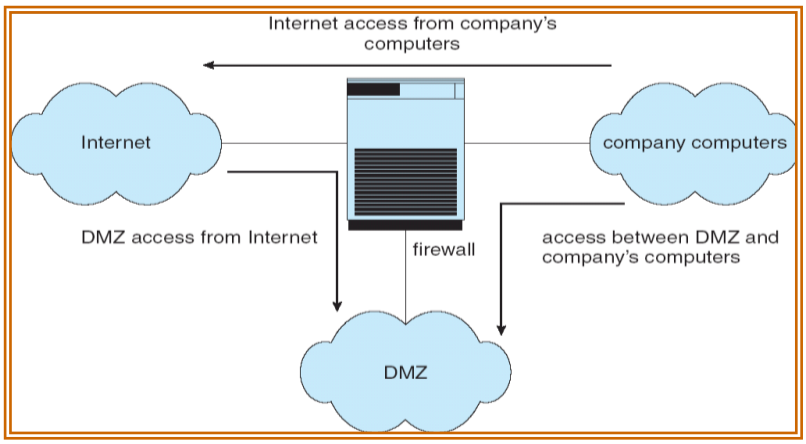
Implementando defesas de segurança

- **Defesa em profundidade é a teoria de segurança mais comum – múltiplas camadas de segurança**
- **Política de segurança descreve o que está sendo protegido**
- **Avaliação de vulnerabilidade compara o estado real do sistema/rede em comparação com a política de segurança**
- **Esforços de detecção de intrusão para detectar intrusões tentadas ou bem sucedidas**
    - **Detecção baseada em assinatura busca padrões de comportamento problemáticos**
    - **Detecção de anomalia busca diferenças do comportamento normal**
        - **Pode detectar ataques do dia zero**
    - **Falsos positivos e falsos negativos: um problema**
- **Proteção contra vírus**
- **Auditoria, contabilidade e logging de todas as atividades da rede ou específicas do sistema**

Uso de firewalls para proteger sistemas e redes

- **Um firewall de rede é colocado entre hosts confiáveis e não confiáveis**
    - **O firewall limita o acesso da rede entre esses dois domínios de segurança**
- **Podem ser tunelados ou forjados**
    - **Tunelamento permite que protocolo não permitido trafegue dentro do protocolo permitido (ou seja, telnet dentro do HTTP)**
    - **Regras de firewall normalmente baseadas no nome de host ou endereço IP que pode ser forjado**
- **Firewall pessoal é camada de software em determinado host**
    - **Pode monitorar/limitar tráfego de e para o host**
- **Firewall de proxy de aplicação entende os protocolos que as aplicações falam pela rede (por exemplo, SMTP)**

- **Firewall de chamada de sistema monitora todas as chamadas do sistema e aplica regras a elas (por exemplo, esse programa pode executar essa chamada do sistema)**



Classificações de segurança de computador

- **O departamento de defesa dos EUA esboça quatro divisões de segurança de computador: A, B, C e D.**
- **D – Segurança mínima.**
- **C – Oferece proteção discricionária por meio de auditoria. Dividido em C1 e C2. C1 identifica usuários em cooperação com o mesmo nível de proteção. C2 permite controle de acesso em nível de usuário.**
- **B – Todas as propriedades de C, porém cada objeto pode ter rótulos de sensibilidade exclusiva. Dividido em B1, B2 e B3.**
- **A – Usa técnicas formais de projeto e verificação para garantir a segurança.**