Análise Comparativa de Segurança em SO

Karolina Cristina Pacheco
Tiago Costa

- Introdução
- Segurança em Sistemas Operacionais de uso corrente
- Sistemas Operacionais voltados a Segurança
- Mandatory Access Control (MAC)
- SELinux (Security Enhanced Linux)

Introdução

- O surgimento dos Sistemas Operacionais tem **dois objetivos principais**
  - Camada de abstração entre hardware/software
  - Gerenciamento eficiente de recursos

- A preocupação com a segurança era **mínima**
  - Computadores isolados
  - Invasões eram somente presenciais

Introdução

- Surgem as **redes de computadores**
  - Execução de múltiplas tarefas é suportada
  - Tempo de processamento compartilhado

- A **internet** é criada
  - Grande quantidade de usuários
  - Dados compartilhados

Introdução

- É necessário **aprimorar a segurança** dos computadores
  - A segurança diz respeito à garantia de algumas propriedades associadas às **informações** e aos **recursos** do SO

SO de uso corrente

- São os mais utilizados em organizações e computadores pessoais
  - **Windows**, **MacOS**, derivados do **UNIX** (Ubuntu, Fedora, Solaris)
- A segurança reside nos **direitos de acesso**
- Três paradigmas de segurança
  - Usuário como limitador de segurança
  - Concentração e isolamento dos direitos privilegiados do computador
  - Cifragem das informações

SO de uso corrente

Usuário como limitador de segurança

- É responsabilidade do usuário restringir os arquivos/recursos

```
drwxr-xr-x 2 root root     1024 Ago 12 13:42 vim
drwxr-xr-x 2 root root     1024 Ago 12 13:56 w3m
-rw-r--r-- 1 root root     4496 Set  5  2010 wgetrc
drwxr-xr-x 3 root root     1024 Ago 12 13:42 X11
drwxr-xr-x 2 root root     1024 Ago 12 13:56 xml
```

- Domínio
  - Quando uma aplicação é executada, ela tem os mesmos privilégios de acesso que o usuário que a executou

SO de uso corrente

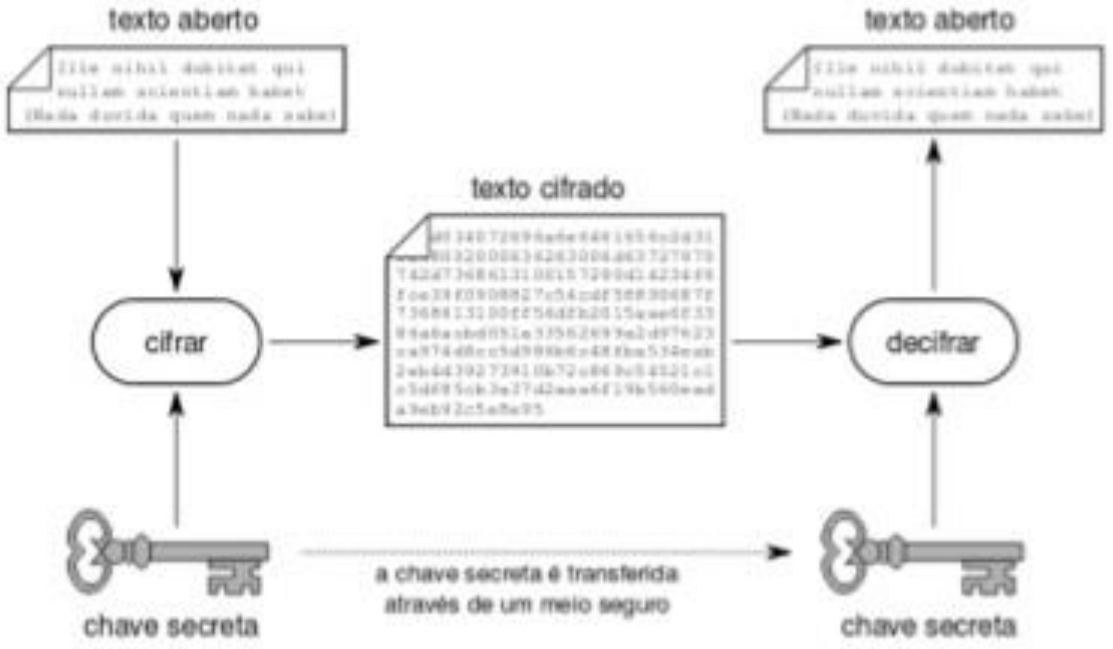
Concentração e isolamento dos direitos privilegiados do computador

- Superusuário ou administrador
- Domínio do superusuário
- Domínio dos usuários comuns
  - Limitado pelo S.O. através de API (chamadas de sistema)
  - Acesso às operações privilegiadas é total ou nenhum

SO de uso corrente

Cifragem das informações

- Garantia de que o arquivo será **ilegível**



SO voltado a Segurança

- Faz-se necessário que o **sistema operacional garanta a segurança** dos arquivos, aplicações e recursos do computador
- São introduzidas **novas propriedades**
  - Controle de acesso mandatório **(MAC)**
    - Conjunto de políticas de segurança que **limitam as permissões que o usuário dispõe** sobre arquivos e aplicações
    - Surge o **administrador de políticas de segurança** cuja responsabilidade é definir quais as políticas a serem aplicadas
    - O **superusuário também se submete às políticas**

SO voltado a Segurança

- Privilégio Mínimo
  - Um usuário ou aplicação não recebe **nenhum privilégio a mais do que o necessário** para realizar suas atividades

- Proteção dos mecanismos da API (acesso ao hardware)
  - Garantia de que os **mecanismos** como canais de comunicação, criptografia, controle de acesso, acesso a dispositivos de entrada e saída, etc. **não tenham seu comportamento alterado** por qualquer aplicação ou vulnerabilidade do sistema

SO voltado a Segurança

Microkernel

- Preveem mecanismos primitivos de proteção
- Visam suportar a construção de uma arquitetura de segurança flexível e de alto nível

Exokernel

- Alto grau de flexibilidade e validação
- Limitado em relação ao suporte de controle de acesso mandatório (MAC)

Flask e DTOS

- Implementação coerente do gerenciador de políticas e restritor de segurança

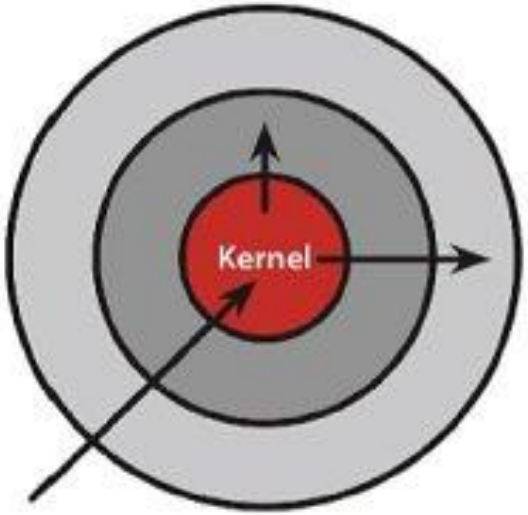
Mandatory Access Control (MAC)

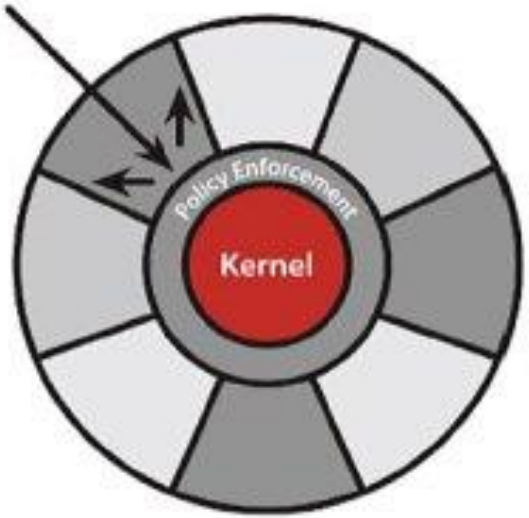
- Restringe acesso a objetos baseado na **sensibilidade da informação** contida
- Autorizações são baseadas em pré-requisitos que são atingidos, resultando na aquisição de acessos individuais
- Capaz de negar acesso a usuários
- Sistema define a política de acessos
- Usuários não podem conceder direitos
- Programas não podem conceder direitos

Mandatory Access Control (MAC)

DAC vs MAC

- MAC não pode ser superado pelo dono do objeto
- MAC pode ser aplicado a objetos não protegidos pelo DAC unix-like, tais como sockets e processos
- Outra vantagem é que o MAC torna possível controle no fluxo de dados





Mandatory Access Control (MAC)

Aplicações MAC recentes:

- **SELinux (Linux)**
- DTE Linux (Linux)
- AppArmor (Linux)
- Mandatory Integrity Control (Vista)
- Trusted BSD
- Trusted Solaris

SELinux (Security Enhanced Linux)

- Desenvolvido pela **National Security Agency** (NSA) e Secure Computing Corporation para uso das tecnologias **MAC**
- Provê uma política de segurança sobre **todos os processos e objetos** do sistema
- Baseia suas decisões em **etiquetas**, contendo uma variedade de informações relevantes a segurança
- Toma como base o princípio do mínimo privilégio ao extremo
- Implementado no **kernel Linux** usando o framework **LSM** (Linux Security Modules)

SELinux (Security Enhanced Linux)

Objetivos de segurança primários do SELinux:

- **Isolamento das Aplicações**: busca o nível do menor privilégio no uso de aplicações. Um problema de segurança em uma aplicação isolada não influencia o sistema como um todo
- **Fluxo de Informações**: garantia de que a informação deve seguir caminhos pré definidos para acesso entre os processos
- **Confidencialidade**: a informação não estará disponível ou será divulgada a indivíduos, entidades ou processos sem a devida autorização
- **Integridade**: disponibilidade de informações confiáveis, corretas e dispostas em formato compatível com o de sua utilização

SELinux (Security Enhanced Linux)

Objetivos de segurança primários do SELinux:

- **Auto-proteção**: além de proteger as políticas de segurança, ele também tem por objetivo proteger o próprio SO (binários, configurações, recursos, etc) para se auto proteger
- **Menor privilégio**: garantir que as políticas aplicadas estão corretas e de que os processos possuem apenas o acesso necessário para realizar a sua função, nada mais do que isso
- **Separação de papéis**: definir permissões de usuários e processos para evitar a elevação de privilégios e suas consequências

EXPERIMENTO

Link Vídeo

Conclusão

- Os SO de uso corrente são eficientes no contexto em que foram desenvolvidos
  - O usuário nem sempre faz a configuração correta das permissões
  - Depender das aplicações para garantir a segurança é perigoso
- Substituição dos sistemas para sistemas voltados a segurança é difícil
  - Adaptação os sistemas existentes
  - SELinux é uma boa alternativa, pois pode ser integrado ao sistemas atuais e já vem configurado em alguns sistemas (Fedora por exemplo)

PERGUNTAS?

karolinacp@edu.unisinos.br
tiagocsilva@edu.unisinos.br