Controle de Acesso

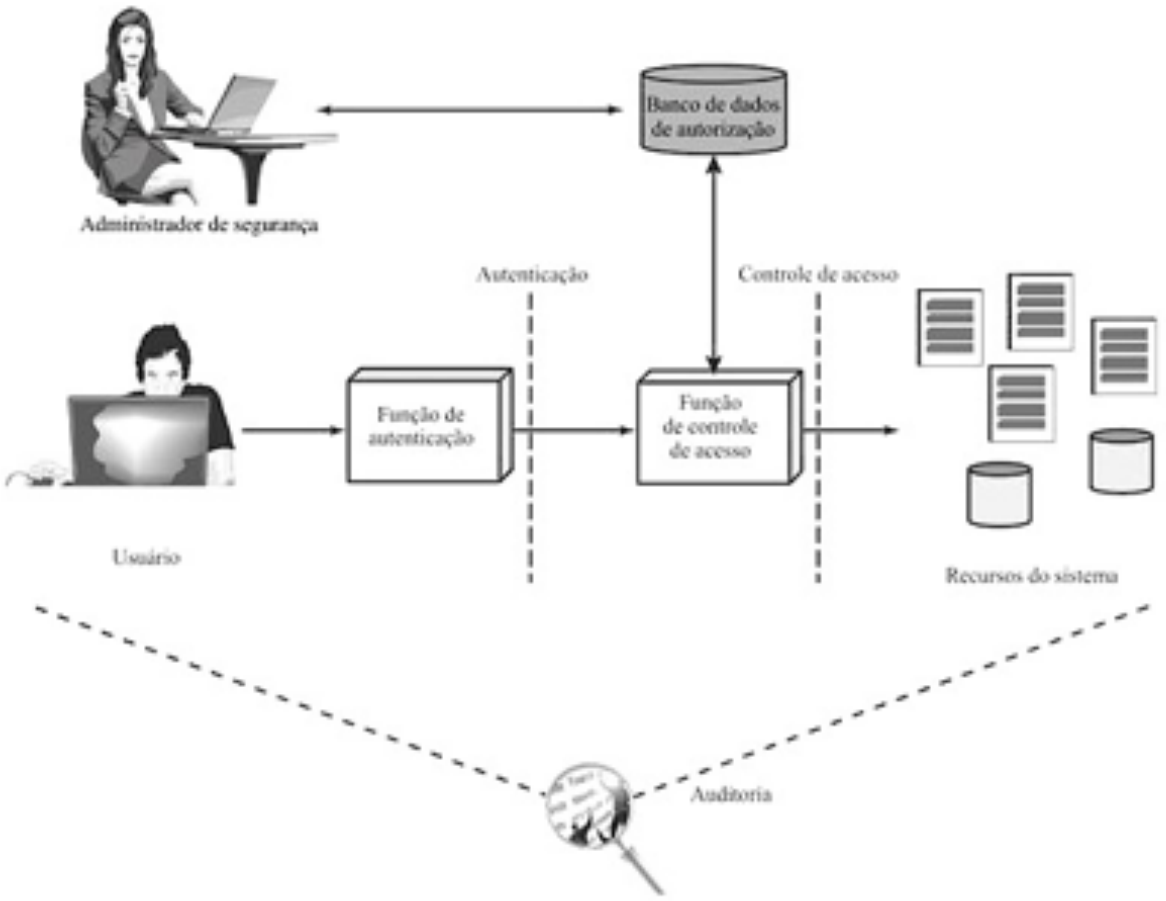
Capítulo 4

Introdução

Controle de Acesso

- "Prevenção do uso não autorizado de um recurso, incluindo a prevenção do uso de um recurso de maneira não autorizada"
- Aspecto central da segurança de computadores
- Assume a existência de usuários e grupos
  - Autenticação no sistema
  - Associação de direitos de acesso a determinados recursos do sistema

Controle de Acesso e Outras Funções de Segurança



Controle de Acesso e Outras Funções de Segurança

- Autenticação
  - verificação de que as credenciais de um usuário ou outra entidade são válidas
- Autorização
  - fornecer o direito ou permissão de acesso a um recurso
- Auditoria
  - revisão independente dos registros e atividades de um sistema com o bjetivo de testar a adequação de controles, promover conformidade às políticas, detectar falhas/brechas e recomendar mudanças

Políticas de Controle de Acesso

- Controle de acesso discricionário (Discretionary Access Control — DAC): entidade pode ter direitos de acesso que lhe permitem, por sua própria vontade, habilitar outra entidade a acessar algum recurso
- Controle de acesso mandatório (Mandatory Access Control — MAC): entidade que está autorizada a acessar um recurso não pode habilitar outra entidade a acessar aquele recurso
- Controle de acesso baseado em papéis (RBAC): Controla o acesso com base nos papéis que os usuários desempenham dentro do sistema

Políticas de Controle de Acesso

- Controle de acesso discricionário (Discretionary Access Control — DAC): entidade pode ter direitos de acesso que lhe permitem, por sua própria vontade, habilitar outra entidade a acessar algum recurso
- Controle de acesso mandatório (Mandatory Access Control — MAC): entidade que está autorizada a acessar um recurso não pode habilitar outra entidade a acessar aquele recurso
- Controle de acesso baseado em papéis (RBAC): Controla o acesso com base nos papéis que os usuários desempenham dentro do sistema

Definições mutuamente não-excludentes

Políticas de Controle de Acesso

Requisitos/Princípios de Controle de Acesso

- **Entrada Confiável**: necessidade de autenticação
- **Especificações mais ou menos detalhadas**: diferentes níveis de granularidade
- **Privilégio mínimo**: menor conjunto possível de recursos
- **Separação de deveres**: etapas de uma função crítica distribuídas entre diferentes usuários
- **Políticas abertas e fechadas**: white vs black list
- **Combinações de políticas e resolução de conflitos**: cenários de aplicação de mais de uma política
- **Políticas administrativas**: quem pode adicionar, eliminar ou modificar as regras de autorização
- **Controle dual**: duas pessoas precisam atuar para completar um processo (cofre com duas chaves)

Elementos de Controle de Acesso

- Sujeito: entidade capaz de acessar objetos
  - Classes típicas de acesso: proprietário, grupo e global
- Objeto: recurso cujo acesso é controlado
  - Exemplos: arquivos, registros, programas etc.
- Direito de acesso: modo pelo qual o objeto é acessado
  - Exemplos: leitura, escrita, execução, remoção, criação, busca

Controle de Acesso Discricionário

Controle de Acesso Discricionário

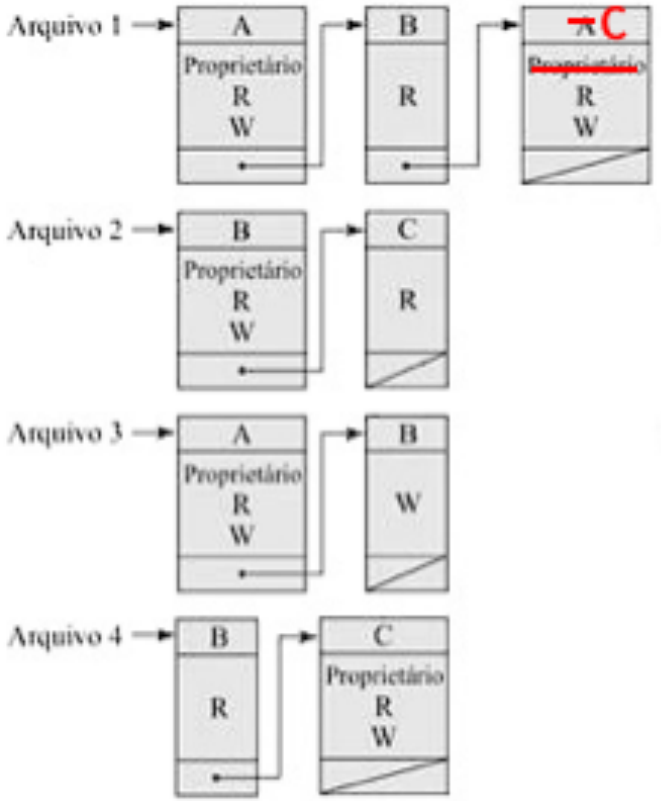
- Todo objeto possui um proprietário que define os direitos de acesso a este objeto
- Direitos de acesso frequentemente definidos por meio de uma matriz de acesso
  - sujeitos em uma dimensão (linhas)
  - objetos em outra dimensão (colunas)
  - cada célula especifica os direitos de acesso do sujeito àquele objeto
- Em geral, a matriz de acesso é esparsa

Estruturas de dados de DAC

		OBJETOS			
		Arquivo 1	Arquivo 2	Arquivo 3	Arquivo 4
SUJEITOS	Usuário A	Proprietário Leitura Escrita		Proprietário Leitura Escrita	
	Usuário B	Proprietário	Proprietário Leitura Escrita	Escrita	Leitura
	Usuário C	Leitura Escrita	Leitura		Proprietário Leitura Escrita

(a) Matriz de acesso

Listas de Controle de Acesso

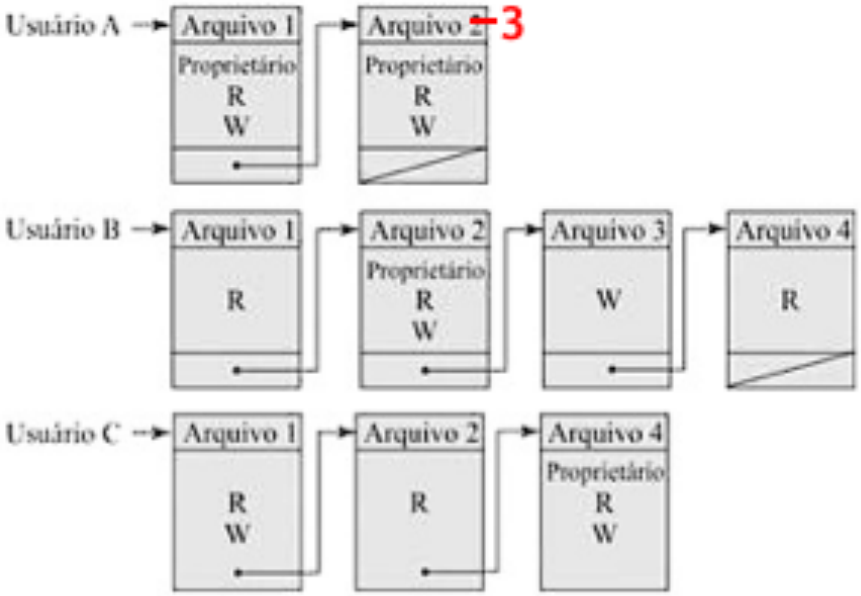


OBJETOS

SUJEITOS	Arquivo 1	Arquivo 2	Arquivo 3	Arquivo 4
Usuário A	Proprietário Leitura Escrita		Proprietário Leitura Escrita	
Usuário B	Proprietário	Proprietário Leitura Escrita	Escrita	Leitura
Usuário C	Leitura Escrita	Leitura		Proprietário Leitura Escrita

(a) Matriz de acesso

Listas de Capacidade



OBJETOS

SUJEITOS	Arquivo 1	Arquivo 2	Arquivo 3	Arquivo 4
Usuário A	Proprietário Leitura Escrita		Proprietário Leitura Escrita	
Usuário B	Proprietário	Proprietário Leitura Escrita	Escrita	Leitura
Usuário C	Leitura Escrita	Leitura		Proprietário Leitura Escrita

(a) Matriz de acesso

Tabela de Autorização

Sujeito	Modo de acesso	Objeto
A	Proprietário	Arquivo 1
A	Leitura	Arquivo 1
A	Escrita	Arquivo 1
A	Proprietário	Arquivo 3
A	Leitura	Arquivo 3
A	Escrita	Arquivo 3
B	Leitura	Arquivo 1
B	Proprietário	Arquivo 2
B	Leitura	Arquivo 2
B	Escrita	Arquivo 2
B	Escrita	Arquivo 3
B	Leitura	Arquivo 4
C	Leitura	Arquivo 1

(a) Matriz de acesso

Modelo de Controle de Acesso

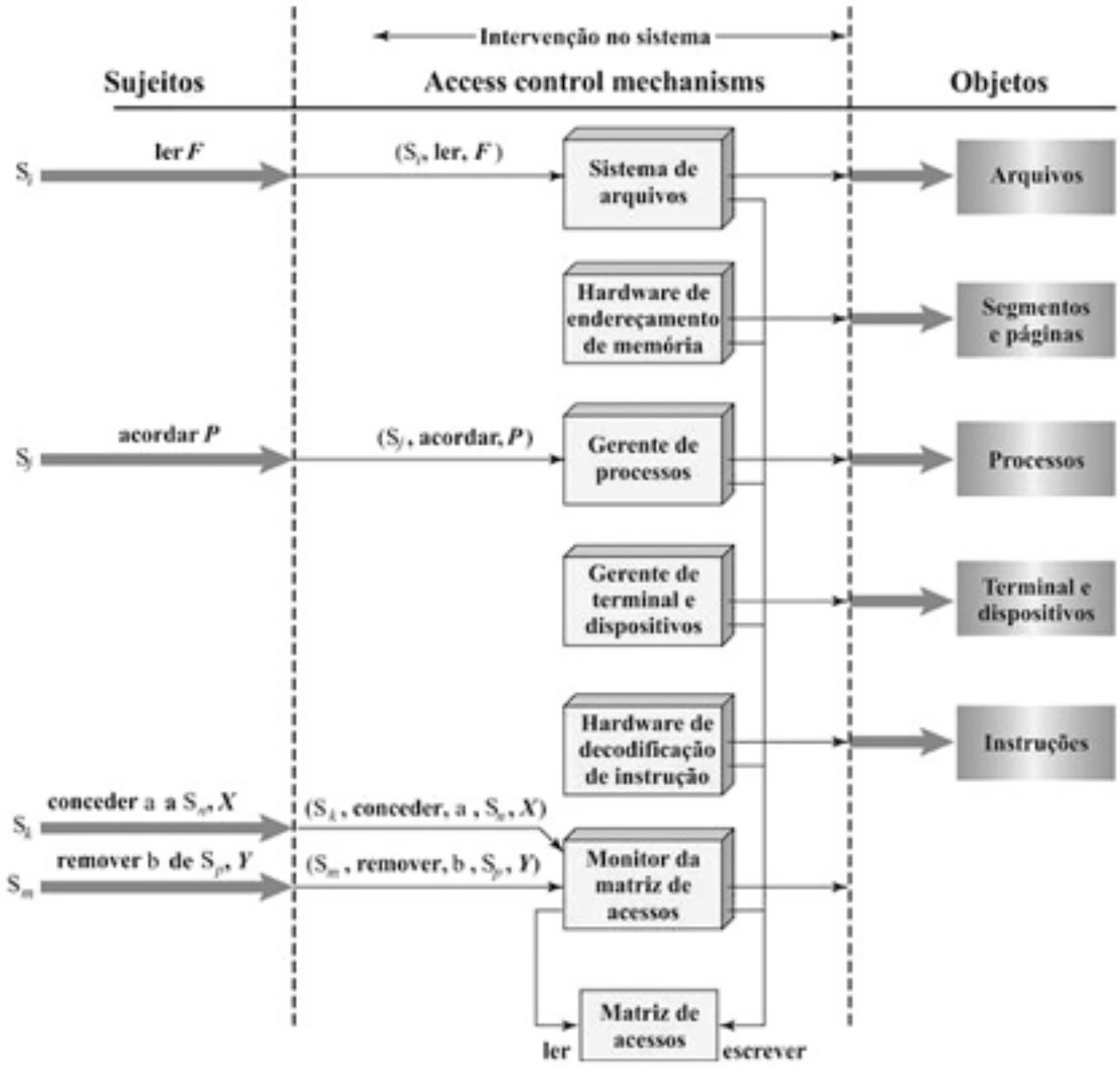
Objetos!

SUJEITOS

Sujeitos! OBJETOS	Sujeitos			Arquivos		Processos		Unidade de disco	
	S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
S_1	controle	proprietário	proprietário controle	leitura*	leitura proprietário	acordar	acordar	buscar	proprietário
S_2		controle		escrita*	execução			proprietário	buscar*
S_3			controle		escrita	parar			

* = copiar conjunto de sinalizador (flag)

Função de Controle de Acesso



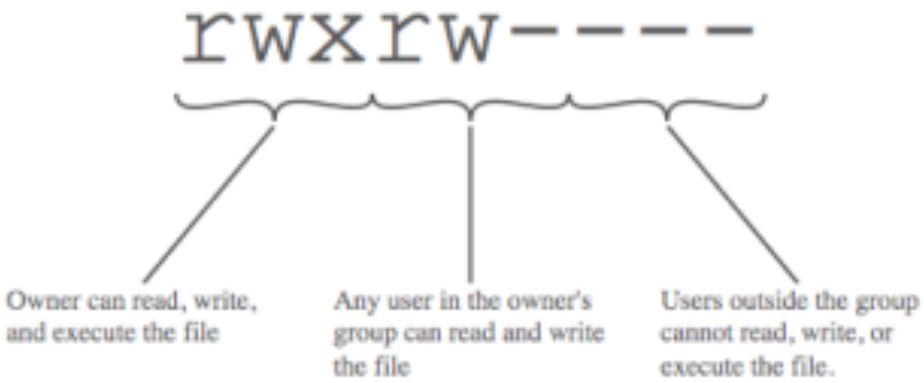
Protection Domains

- set of objects with associated access rights
- in access matrix view, each row defines a protection domain
  - but not necessarily just a user
  - may be a limited subset of user's rights
  - applied to a more restricted process
- may be static or dynamic

UNIX File Concepts

- UNIX files administered using inodes
  - control structure with key info on file
    - attributes, permissions of a single file
  - may have several names for same inode
  - have inode table / list for all files on a disk
    - copied to memory when disk mounted
- directories form a hierarchical tree
  - may contain files or other directories
  - are a file of names and inode numbers

UNIX File Access Control



UNIX File Access Control

- "set user ID"(SetUID) or "set group ID"(SetGID)
  - system temporarily uses rights of the file owner / group in addition to the real user's rights when making access control decisions
  - enables privileged programs to access files / resources not generally accessible
- sticky bit
  - on directory limits rename/move/delete to owner
- superuser
  - is exempt from usual access control restrictions

UNIX Access Control Lists

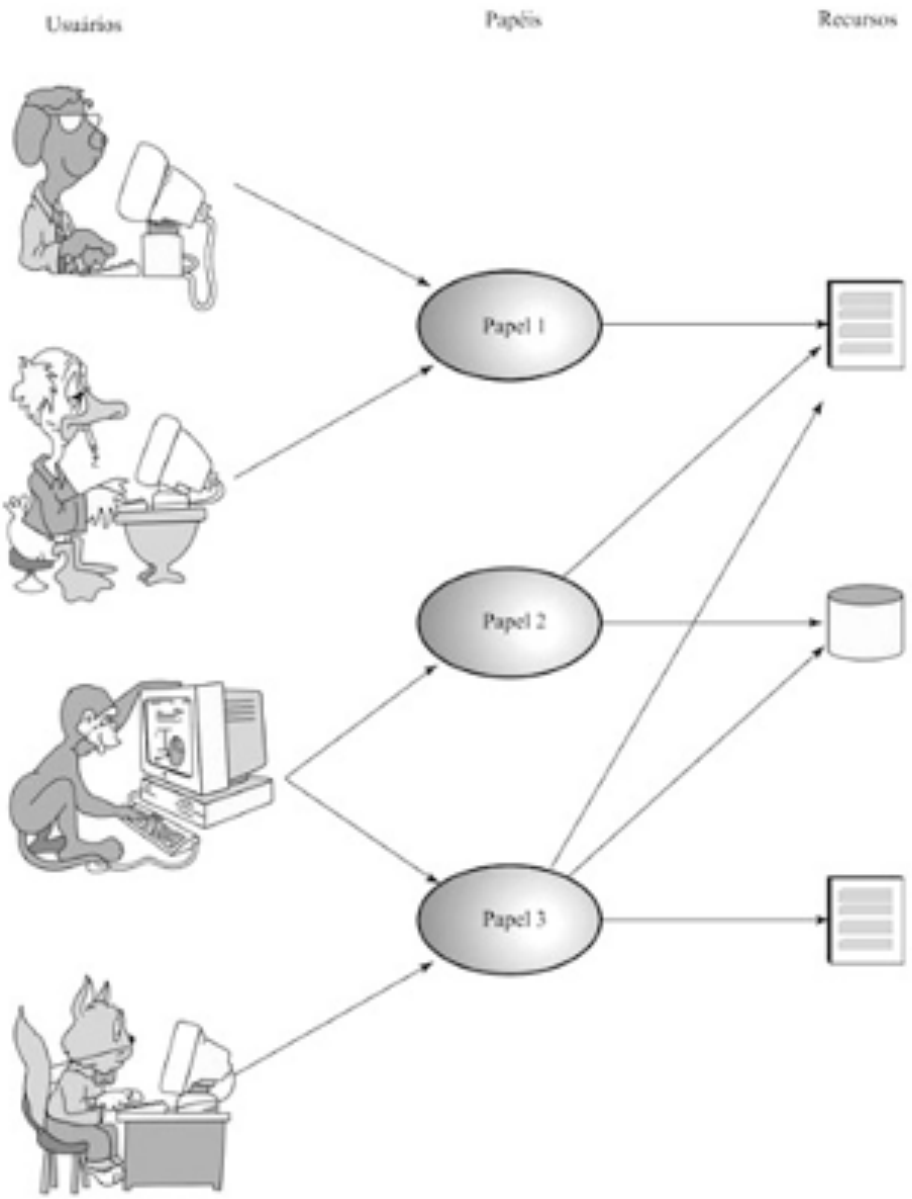
- modern UNIX systems support ACLs
- can specify any number of additional users / groups and associated rwx permissions
- ACLs are optional extensions to std perms
- group perms also set max ACL perms
- when access is required
  - select most appropriate ACL
    - owner, named users, owning / named groups, others
  - check if have sufficient permissions for access

Controle de Acesso Baseado em Papéis (RBAC)

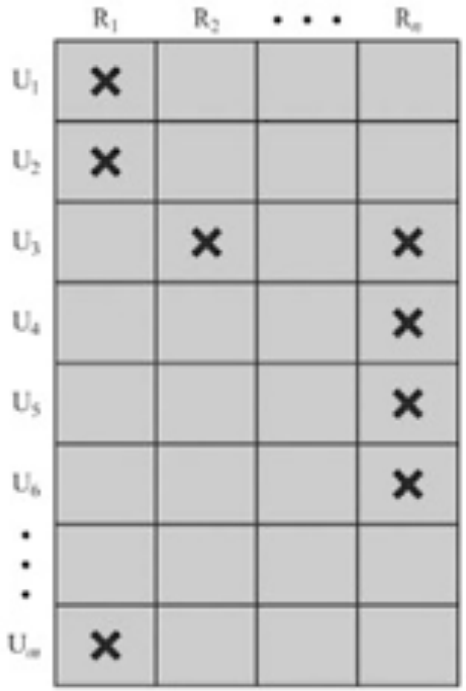
Controle de Acesso Baseado em Papéis

- Baseado nos "papéis" que usuários podem assumir em um sistema
  - Não depende diretamente da identidade
- Os direitos de acesso estão atribuídos a "papéis" – e não a usuários individuais
  - Matriz de acesso do RBAC para "papéis" é similar à matriz de acesso de DAC para "sujeitos"
  - Papéis podem ser tratados como objetos – hierarquia de papéis
- Relações "muitos para muitos" entre papéis e usuários
- RBAC tem uso comercial disseminado, pesquisa ativa e reconhecimento técnico
  - FIPS 140-2 exige suporte a RBAC

Controle de Acesso baseado em papéis



Controle de Acesso baseado em papéis

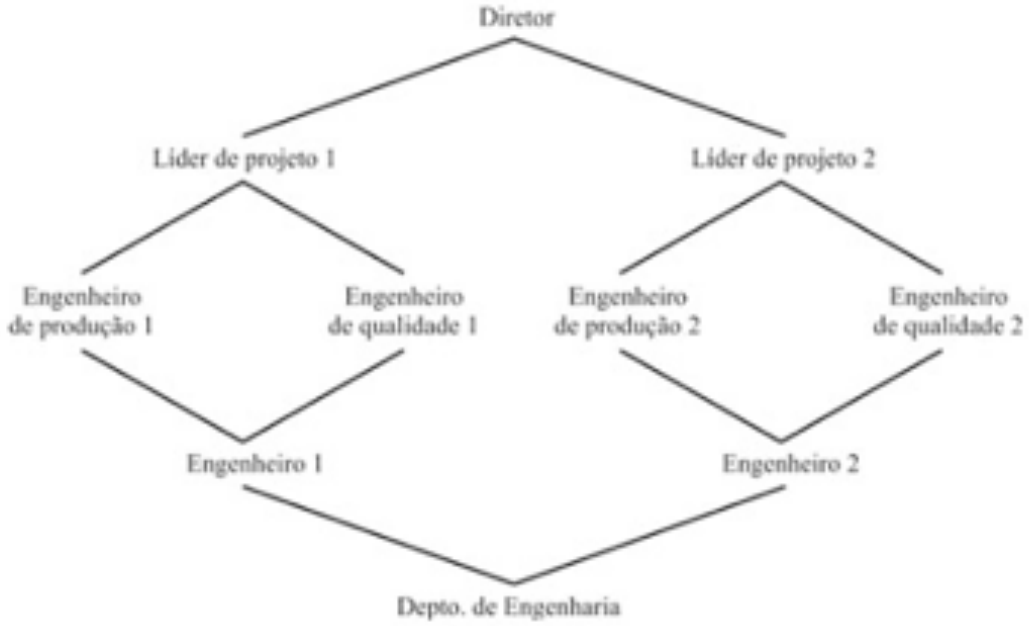


OBJETOS

PAPÉIS	R1	R2	Rn	F1	F2	P1	P2	D1	D2
R1	controle	proprietário	proprietário controle	leitura*	leitura proprietário	acordar	acordar	buscar	proprietário
R2		controle		escrita*	execução			proprietário	buscar*
. . .									
Rn			controle		escrita	parar			

Hierarquia de papéis no RBAC

- Hierarquia da organização pode se refletir nos papéis
- O conjunto de direitos de cada papel contém a união dos conjuntos de direitos de seus "filhos"

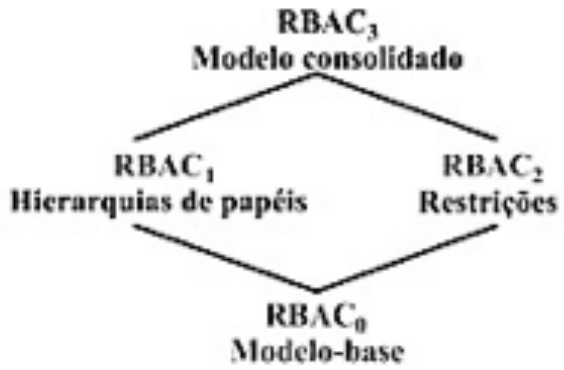


Restrições no modelo RBAC

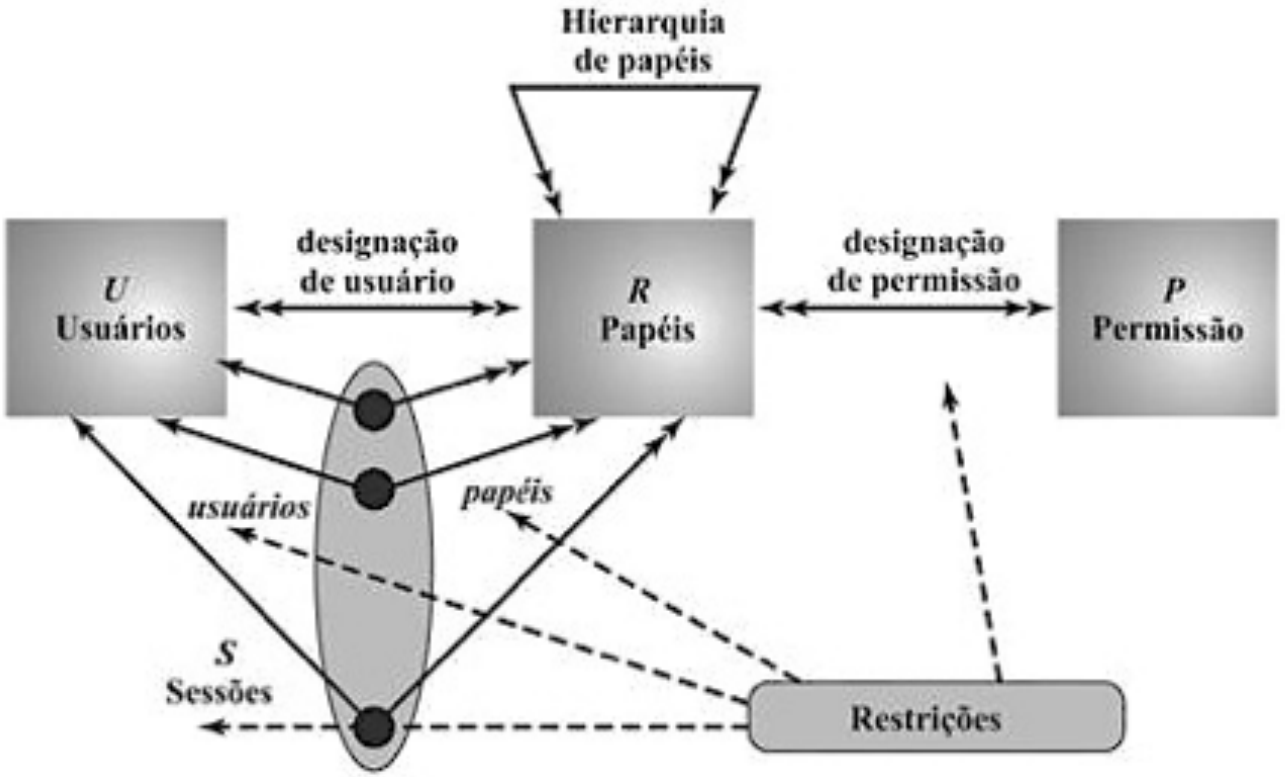
- Envolvem restrições a relacionamento de papéis ou condições sobre papéis
- Papéis mutuamente exclusivos: usuário só pode assumir um papel num conjunto de papéis
  - Conceito de segregação de deveres
- Cardinalidade: restrições sobre quantidades associadas a papéis
  - máximo número de usuários associados a papel
  - máximo número de papéis assumidos por usuário
  - máximo número de papéis com determinada permissão
- Pré-requisitos: condições para assumir pape
  - usuário só pode assumir papel sênior se possuir papel júnior

Modelos de Referência para RBAC

- $RBAC_0$: Funcionalidades mínimas para sistema RBAC
- $RBAC_1$: $RBAC_0$ + hierarquia de papéis
- $RBAC_2$: $RBAC_0$ + restrições sobre configurações
- $RBAC_3$: $RBAC_1$ + $RBAC_2$



Modelo consolidado RBAC(3)



Modelo-base: $RBAC_0$

- Quatro tipos de entidades em um sistema $RBAC_0$
  - Usuário: indivíduo com acesso a sistema computacional - cada indivíduo tem um ID de usuário a ele associado.
  - Papel: função definida no sistema computacional ou organização que o controla - normalmente, associada a cada papel há uma descrição da autoridade e da responsabilidade conferidas a esse papel e a qualquer usuário que assuma esse papel
  - Permissão: aprovação de modo de acesso em particular a um ou mais objetos - termos equivalentes são direito de acesso, privilégio e autorização.
  - Sessão: mapeamento entre um usuário e um subconjunto ativado do conjunto de papéis atribuídos a um usuário.

Hierarquia de papéis: $RBAC_1$

› O $RBAC_1$ permite reproduzir a hierarquia de papéis típica das organizações
  – Funções com maior responsabilidade agregam/acumulam permissões para acesso a recursos

Restrições: $RBAC_2$

- Restrições fornecem uma forma de adaptar o RBAC às políticas administrativas e de segurança específicas de uma organização
- Uma restrição é uma relação definida entre papéis ou uma condição relacionada a papéis
- Tipos de restrições
  - papéis mutuamente exclusivos
  - cardinalidade
  - papéis com pré-requisitos

Tipos de restrições

- Papéis mutuamente exclusivos
  - um usuário só pode ser designado para um único papel do conjunto de papéis
  - limitação poderia ser estática ou dinâmica
  - provê suporte a uma separação de deveres e capacidades dentro de uma organização
- Cardinalidade
  - número máximo com relação a papéis
  - Ex.: número máximo de usuários que podem ser designados a determinado papel - o papel de líder de projeto ou o papel de chefe de departamento poderia ser limitado a um único usuário
  - também pode impor uma restrição ao número de papéis aos quais um usuário é designado ou ao número de papéis que um usuário pode ativar para uma única sessão
- Pré-requisito
  - usuário só pode ser designado a determinado papel se já estiver designado a algum outro papel especificado
  - pode ser usado para estruturar a implementação do conceito do privilégio mínimo.

Controle de Acesso Mandatório (MAC)

Controle de Acesso Mandatório

- Baseado no conceito de Multi-Level Security (MLS)
- Cada usuário possui um nível de acesso (clearance)
- Cada objeto possui um nível de classificação
- Exemplo (ambiente diplomático/militar):

  top secret > secret > confidential > restricted > unclassified

- Requisitos para confidencialidade:
  - No Read Up
  - No Write Down
- Níveis definidos pelo administrador
- Modelo de Bell-LaPadula

Implementações de MAC

- SELinux: Linux kernel modules available to most Linux distributions (RedHat, Debian, Ubuntu, SuSE, ...)
- AppArmor: some Linux distributions (Ubuntu, SuSE)
- TrustedBSD: FreeBSD, OpenBSD, OSX, ...
- Mandatory Integrity Control: Vista, Windows 7, Windows 8

Pontos-chave sobre controle de acesso

- Controle de acesso previne o uso não-autorizado de recursos (objetos) por usuários (sujeitos)
  - Na prática, em um sistema, os sujeitos são processos agindo em nome de usuários e aplicações
- Classes de sujeitos (Unix): proprietário, grupo, outros
- Tipos de objetos: arquivos, registros de bancos de dados, blocos de disco, segmentos de memória, processos,...
- DAC: direitos de acesso podem ser dados a outros usuários
- RBAC: sujeitos assumem papéis; direitos associados a papéis
- MAC: sujeitos e objetos associados a níveis; sujeitos não podem modificar política de acesso

Estudo de Caso

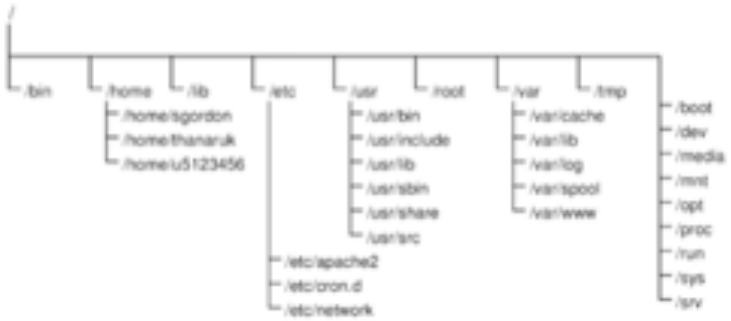
Hierarquia e Permissões no Sistema de Arquivos Unix-type

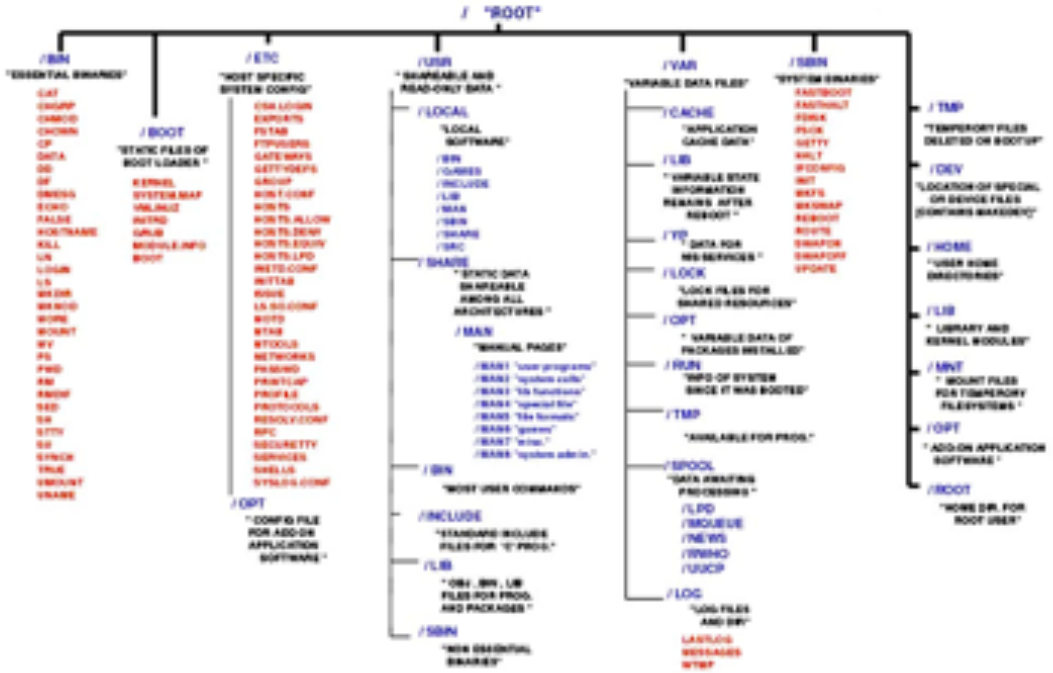
Multiuser and Server Operating System

- Linux systems are commonly used as a multi-user system
  - E.g. students and staff at SIIT have account on ICT
- Linux systems are commonly used as servers
  - Web, email, SSH, database servers
- How to ensure that authorized users can access only designated resources on a Linux system?
  - Understand filesystem organisation
  - Understand access control mechanisms on the filesystem

Hierarquia do sistema de arquivos Unix

- A maioria dos sistemas operacionais Unix-like tem hierarquias de sistemas de arquivos semelhantes
- Arquivos e diretórios
- Diretório raiz é "/"
- Exemplo de hierarquia Linux





Diretórios mais importantes de conhecer

/home/nome_usuario	Arquivos do usuário
/media	Drives externos
/etc	Configuração do S.O.
/var/www	Arquivos de websites
/var/logs	Arquivos de logs
/root	Arquivos do usuário root
/proc	Processos do S.O.
/dev e /sys	Dispositivos do S.O.
/var/mail	Email de entrada
/var/lib	Dados de aplicações

inodes

- Usados pelo S.O. para gerenciar arquivos e diretórios
- Estruturas de dados que armazenam informações sobre arquivos e diretórios
  - modo
  - informação do proprietário
  - tamanho
  - timestamps
  - ponteiros para os blocos de dados
- S.O. mantém lista de inodes em tabela de inodes
- Um diretório é um arquivo que lista os arquivos naquele diretório incluindo:
  - número do inode de cada arquivo
  - comprimento do nome de arquivo
  - nome do arquivo

Conteúdo do inode

- modo: 16 bits
  - 12 bits de proteção: permissões
  - 4 bits de tipo de arquivo: arquivo regular, diretório,...
- proprietário: 16 bits do user ID
- grupo: 16 bits do group ID
- tamanho: tamanho do arquivo em bytes
- timestamps: última vez, em segundos desde a epoch...
  - atime: inode acessado
  - ctime: inode modificado
  - mtime: dados do arquivo modificados
- além de outros campos...

Permissões e Usuários

- Permissões
  - Ler (read) arquivo; listar conteúdo de diretório
  - Escrever (write) arquivo; criar/remover arquivos no diretório
  - Executar (execute) arquivo; acessar arquivos no diretório
- Categorias de usuários
  - Usuário (user) que é proprietário do arquivo
  - Usuários no grupo (group) do proprietário do arquivo
  - Outros (other) usuários
  - * Todos os usuários (all) inclui a união dos três acima

Bits de proteção num inode

- 12 bits de um inode são bits de proteção
  - Primeiros 9 bits indicam permissões de leitura/escrita/execução para proprietário/grupo/outros
  - Últimos 3 bits indicam permissões especiais
- Tipo de arquivo e bits de proteção mostrados em formato amigável com *ls -l*