14: Protección

Sistemas Operativos 2
Ing. Alejandro León Liu

- **Protección**
- Dominios de protección
- Matriz de Accesos
- Implementación de matriz de accesos
- Roles
- Protección en compiladores / lenguajes de programación

PROTECCIÓN

- Acceso controlado de procesos y usuarios a los recursos del sistema
- Proteger un proceso de otro
  - Multiprogramación
  - Aumentar la confiabilidad del sistema
- Uso adecuado de los recursos
- Mecanismos vrs. políticas

- Principios
  - Propiedades y comportamientos de un sistema
  - Guía en el diseño de un sistema
  - Principle of least priviledge
    - Usuario o proceso únicamente tiene acceso a información y recursos necesarios para cumplir su propósito
    - Minimizar consecuencias de uso indebido
    - Modo kernel
    - Root
  - Auditar accesos

- Protección
- **Dominios de protección**
- Matriz de Accesos
- Implementación de matriz de accesos
- Roles
- Protección en compiladores / lenguajes de programación

DOMINIO DE PROTECCIÓN

- ¿Qué queremos proteger?: objetos
  - Recursos (hardware): CPU, memoria, I/O
  - Software: semáforos, archivos, programas, etc…
- Conjunto de Permisos
  - <objeto, acciones>
- Implementado a nivel de
  - Usuario
  - Proceso
  - Procedimiento

- UNIX
  - Dominio asociado a usuario
  - Cambio de dominio: cambio de usuario
    - Su, sudo
  - Acciones sobre archivos
    - Leer
    - Escribir
    - Ejecutar
  - Por cada archivo, especificar tres diferentes dominios
    - Owner: Dueño del archivo. RWX
    - Group: Usuarios del mismo grupo que el archivo RWX.
    - Universe: Todos los demás usuarios RWX.

- Protección
- Dominios de protección
- **Matriz de Accesos**
- Implementación de matriz de accesos
- Roles
- Protección en compiladores / lenguajes de programación

MATRIZ DE ACCESOS

- Filas: Dominios
- Columnas: Objetos
- Políticas: dictadas por los usuarios
- Mecanismos: Matriz de acceso

domain \ object	F_1	F_2	F_3	printer
D_1	read		read	
D_2				print
D_3		read	execute	
D_4	read write		read write	

- Cambio de dominio
  - Agregar cada dominio como una columna

object / domain	F_1	F_2	F_3	laser printer	D_1	D_2	D_3	D_4
D_1	read		read			switch		
D_2				print			switch	switch
D_3		read	execute					
D_4	read write		read write		switch			

- Permisos para manipular accesos
  - Owner: Puede manipular accesos sobre determinado objeto
  - Copy: Puede extender sus accesos sobre un objeto a otros dominios.
    - Puede incluso copiar el permiso de copy
  - Control (sobre un dominio): permite alterar accesos de este dominio

Copy

object / domain	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute		

(a)

object / domain	F_1	F_2	F_3
D_1	execute		write*
D_2	execute	read*	execute
D_3	execute	read	

(b)

Owner

object / domain	F_1	F_2	F_3
D_1	owner execute		write
D_2		read* owner	read* owner write
D_3	execute		

(a)

object / domain	F_1	F_2	F_3
D_1	owner execute		write
D_2		owner read* write*	read* owner write
D_3		write	write

(b)

- Protección
- Dominios de protección
- Matriz de Accesos
- **Implementación de matriz de accesos**
- Roles
- Protección en compiladores / lenguajes de programación

IMPLEMENTACIÓN

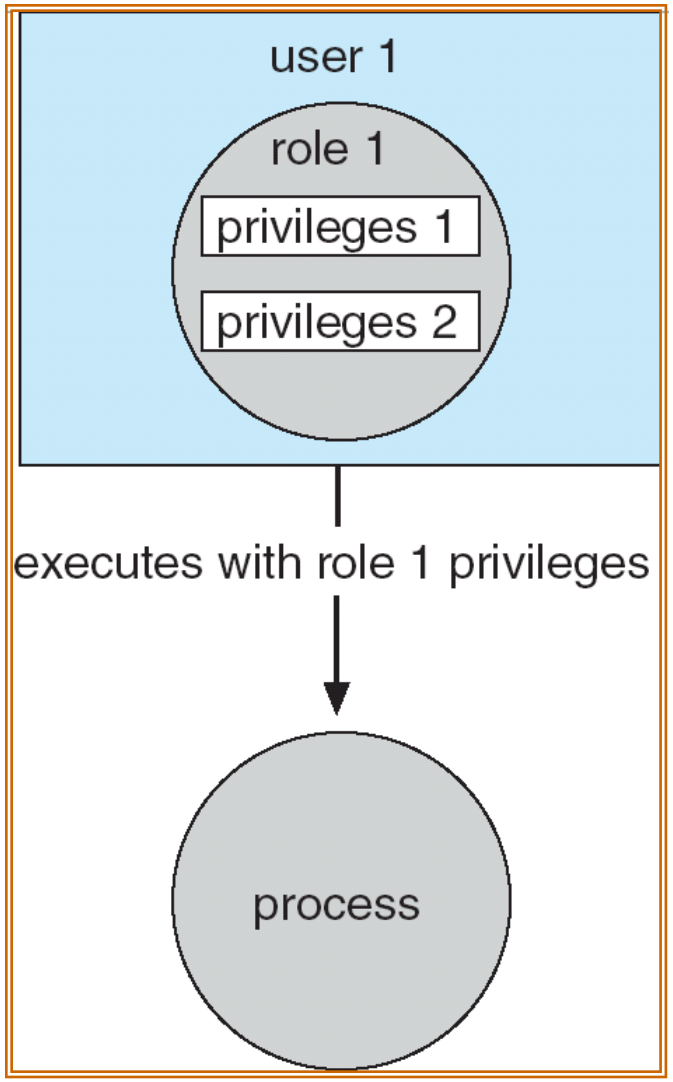
- Tabla global
  - Lista de <dominio, objeto, permisos>
  - Tabla grande
  - Difícil agrupar dominios
- Lista de accesos por objeto
  - Lista de <dominio, permisos> por objeto
  - Los dominios no encontrados, tienen permisos por defecto

- Lista de capacidades por dominio
  - Lista de <objeto, permisos> por dominio
  - Manejada por el S.O., no por procesos de ese dominio
  - Difícil revocar permisos

- Protección
- Dominios de protección
- Matriz de Accesos
- Implementación de matriz de accesos
- **Roles**
- Protección en compiladores / lenguajes de programación

ROLES

- Anteriormente, manejar permisos en file system
- Permisos de ejecución de un proceso
- Conjunto de permisos para ejecutar system calls
- Solaris



- Protección
- Dominios de protección
- Matriz de Accesos
- Implementación de matriz de accesos
- Roles
- **Protección en compiladores / lenguajes de programación**

PROTECCIÓN EN COMPILADORES

- Capa de abstracción sobre system calls
  - Validación de parámetros
  - Abstracción
- Constantes
- Scope de clases, variables, métodos
- Protección de memoria
  - Punteros
- Protección de buffer overflow
- Tipos de variables
- Control de asignación de memoria dinámica
- Garbage collector