

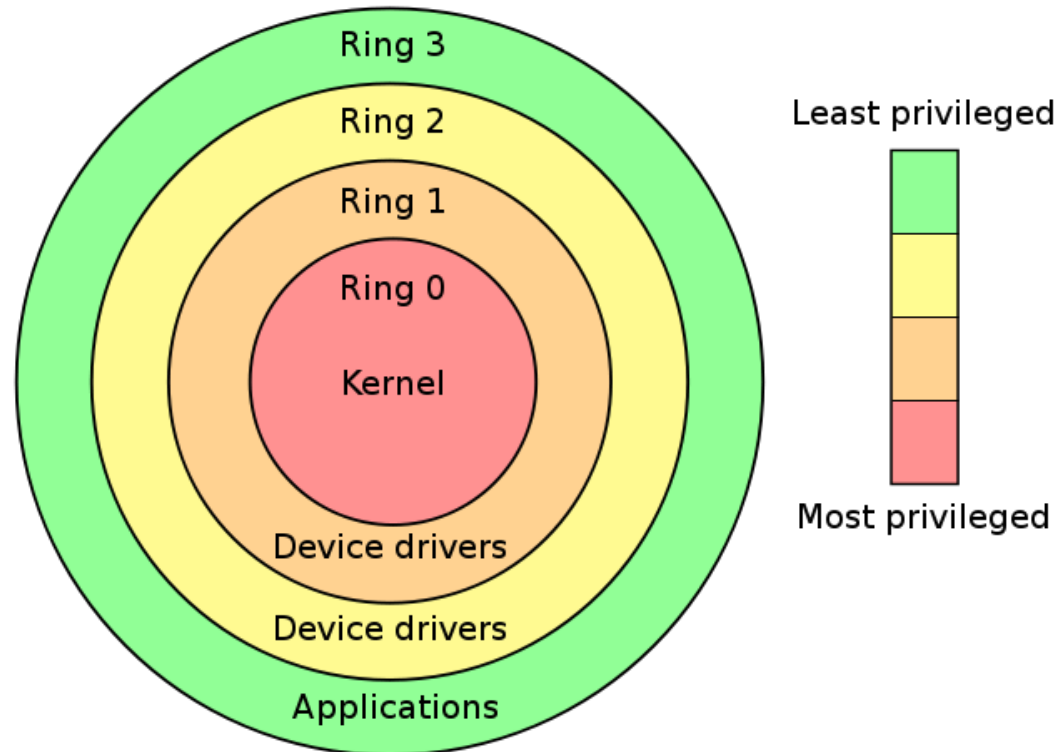
Unidad 6

Sandbox

Se-Linux y otros mecanismos de protección

X86 privilege levels

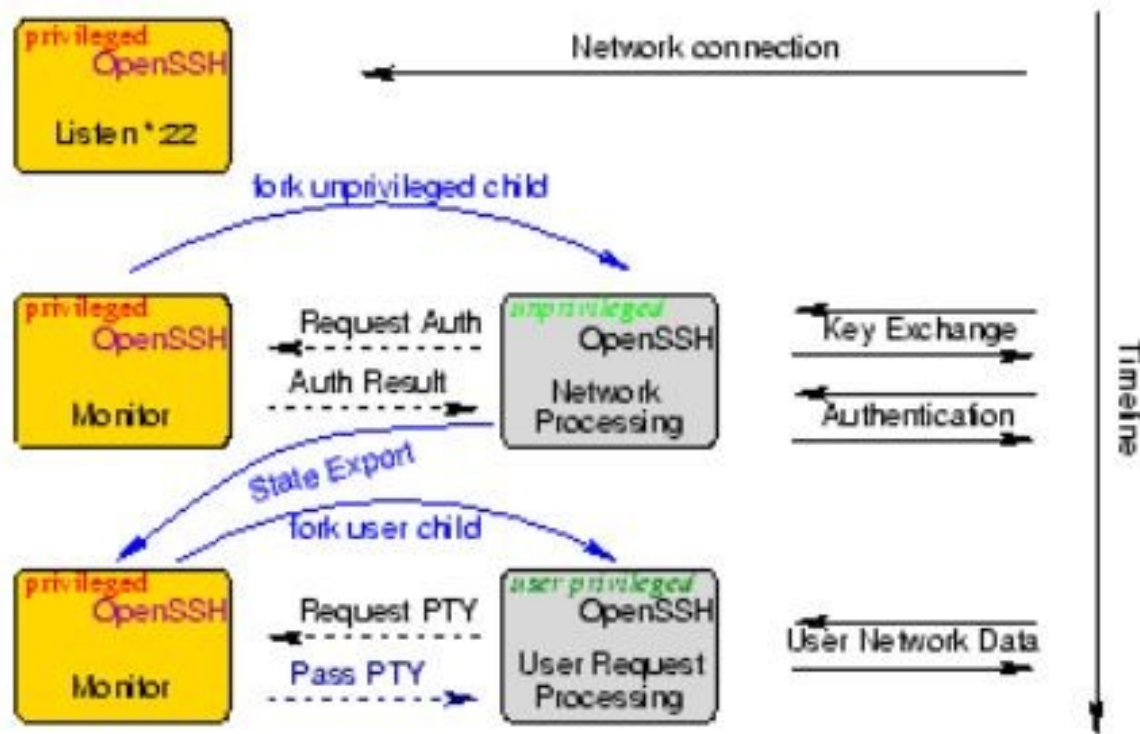
- ¿Cuántos se usan?



- **Consiste en cambiar la visión del sistema de archivos disponible para el proceso.**
- **Todas las operaciones sobre archivo (open, etc.) sobre el directorio “/” se corresponden con operaciones sobre un subdirectorio (la jaula) del sistema de archivos real.**
- **Es útil para aislar procesos de otra información sensible presente en el sistema**
- **No enjaula las demás syscalls (se puede ver la lista de procesos de todo el sistema, etc, etc)**
- **Si estoy con usuario root, puedo salir de la jaula.**

- **Cada Jail provee:**
- **Un entorno virtual con sus propios archivos, procesos, usuarios y administrador. Desde un jail, el sistema no se distingue de un sistema real.**
- **Cada Jail es separada del resto.**
- **Facilidad de delegación de tareas sin dar control completo sobre el sistema.**
- **Tecnologías similares: Solaris Containers, Linux Vserver**

OpenSSH Privilege separation



BSD security level

-1 Permanently insecure mode – Siempre correr el sistema en modo 0.

0 Insecure mode – flags de immutable y append-only pueden ser apagados. Todos los dispositivos pueden ser leídos o modificados en base a sus permisos.

1 Secure mode – No se pueden apagar los flags immutable y append-only. No se puede escribir directamente a dispositivos montados, ni a /dev/mem, o /dev/kmem. Los módulos de kernel no pueden ser cargados o descargados.

2 Highly secure mode – se agrega que los discos solo pueden ser abiertos para escritura por el mount. Tampoco se pueden crear nuevos filesystems mientras el sistema está en modo multi-usuario. Los cambios a la hora del kernel se restringen a menos de un segundo.

3 Network secure mode – como el anterior, pero además no se puede cambiar la configuración del firewall.

Windows User Account Control

- Cuando nos logueamos a Vista/7 como un usuario estándar, se crea un token de sesión conteniendo privilegios básicos.
- Cuando nos logueamos a Vista/7 con un usuario administrador, se crean dos tokens de sesión. Uno contiene todos los privilegios del administrador, y el otro está restringido con permisos parecidos a los de un usuario estándar.
- Las aplicaciones normalmente son iniciadas con el token restrictivo, y cuando una aplicación requiere mayores privilegios, el UAC pide confirmación y, si se da consentimiento, el nuevo proceso se inicia con el token que no posee restricciones.

Funciones que disparan la ventana del UAC

- Ejecutar una aplicación como administrador
- Cambios a seteos que afecten a todo el sistema, o archivos en %SystemRoot% o %ProgramFiles%
- Instalar o desinstalar aplicaciones
- Instalar drivers de dispositivos
- Instalar controles ActiveX
- Cambiar la configuración del firewall de windows
- Cambiar configuración del UAC
- Configurar el Windows Update
- Administrar cuentas de usuario
- Configurar el control parental
- Ejecutar el task scheduler
- Restaurar backups
- Ver o modificar archivos de otros usuarios
- Ejecutar el defragmentador de discos

Vista Windows Integrity Control

- **Conocido inicialmente como Mandatory Integrity control.**
- **Se basa en el modelo Biba de control de Integridad.**
- **Define seis niveles de integridad**
- Trusted Installer
- System (operating system processes)
- **High (administrators)**
- **Medium (non-administrators)**
- **Low (temporary Internet files)**
- Untrusted
- **Achivos, carpetas, usuarios, procesos, todos tienen niveles de integridad.**
- **El nivel medio es el nivel por defecto para usuarios estándar y objetos sin etiquetas.**
- **El usuario no puede darle a un objeto un nivel de integridad más alto que el suyo.**

Servicios y mínimos privilegios en 2003

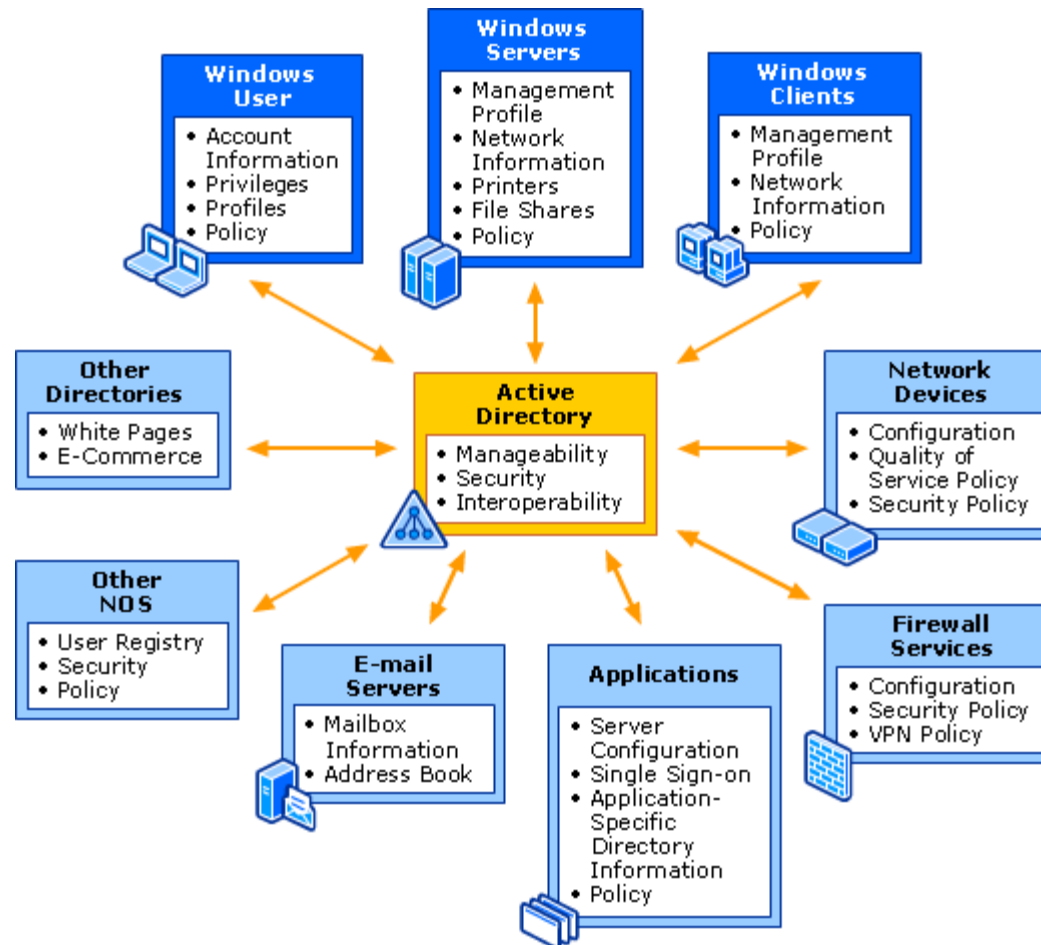
- **Local Service.** Esta cuenta tiene privilegios mínimos en el equipo local. Los servicios que inicien sesión como Local Service acceden a los recursos de red utilizando una sesión nula con credenciales anónimas.
- **Network Service.** Esta cuenta también tiene privilegios mínimos en el equipo local. Los servicios que inicien sesión como Network Service acceden a la red mediante las credenciales de la cuenta de equipo.
- **Unique user account.** Un servicio debe ejecutarse como una cuenta de usuario único sólo si no es práctico ejecutarlo como servicio local o servicio de red.
- **Local System.** Esta cuenta tiene amplios privilegios en el equipo local. Los servicios que inicien sesión como local system acceden a los recursos de red del sistema mediante las credenciales de la cuenta de equipo.
- **Local administrator account.** Debe ejecutar un servicio como una cuenta de administrador local sólo si no es práctico ejecutarlo como Local Service, Network Service, Unique user account o Local System.
- **Domain administrator account.** La ejecución de un servicio utilizando una cuenta de administrador de dominio es el peor escenario de seguridad.

Mejoras en windows 2008

Service	Previous Context	New Context
COM+ Event System	SYSTEM	LOCAL SERVICE
Windows Security	SYSTEM	LOCAL SERVICE
Windows Event Log	SYSTEM	LOCAL SERVICE
Windows Audio	SYSTEM	LOCAL SERVICE
Workstation Service	SYSTEM	LOCAL SERVICE
Windows Image Acquisition	SYSTEM	LOCAL SERVICE
Windows Time	SYSTEM	LOCAL SERVICE
DHCP Client	SYSTEM	LOCAL SERVICE
Telephony	SYSTEM	NETWORK SERVICE
Cryptographic Services	SYSTEM	NETWORK SERVICE
Policy Agent	SYSTEM	NETWORK SERVICE
Terminal Services	SYSTEM	NETWORK SERVICE

Table 12-1 Vista Services that Have Now Run Under Lower-privileged Accounts

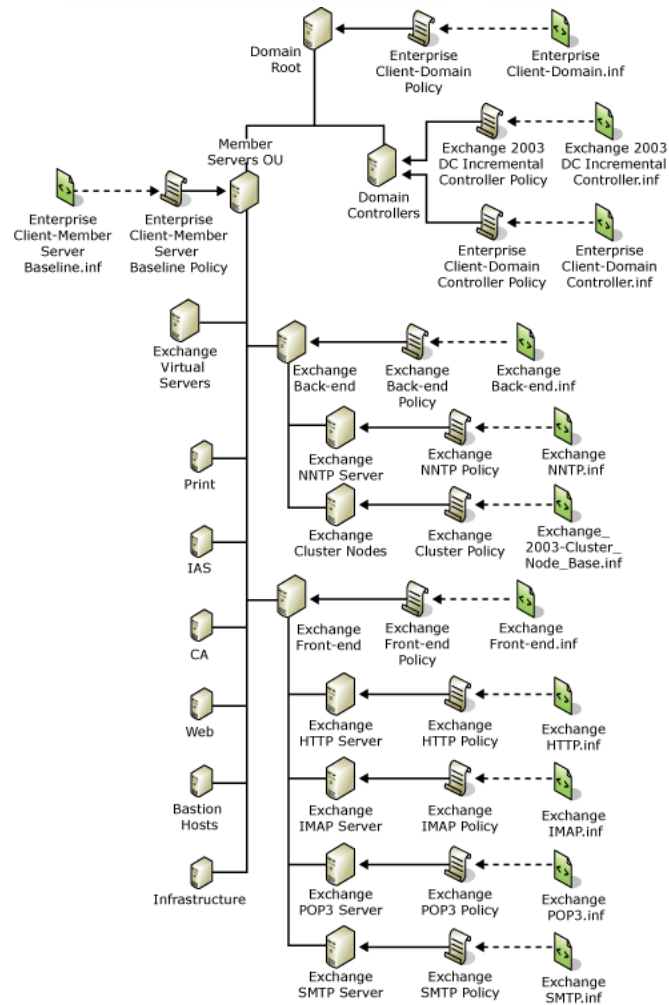
Active Directory



- **Las Directiva de grupo proporcionan una forma para que los administradores puedan aplicar configuraciones coherentes a los grupos de usuarios y equipos.**
- **Las directivas de grupo puede ayudar a hacer cumplir las políticas por escrito de su organización. Por ejemplo, la política de seguridad de su organización puede requerir que todos los equipos del departamento de investigación muestren un mensaje cuando los usuarios inician sesión, informándoles de un mayor seguimiento de seguridad en ese departamento. La Directiva de grupo le permite configurar de forma centralizada, implementar y administrar un mensaje de advertencia, y aplicarlo a los equipos necesarios.**

- **Una de las mayores características de seguridad de las Directivas de grupo es la capacidad de desplegar plantillas de seguridad en una organización. Las plantillas de seguridad, hacen posible un paquete de configuración de seguridad todo en un solo archivo (la plantilla). Por ejemplo, puede crear una plantilla de seguridad para los equipos cliente en su organización y luego utilizar Directiva de grupo para implementar la plantilla de seguridad a los equipos cliente.**
- **De esta manera, puede configurar los equipos para tener una configuración de seguridad coherente. Debido a que las plantillas se pueden administrar de forma centralizada, se puede actualizar, revisar y mejorar la configuración de seguridad a través del tiempo como lo requiera su organización.**

Ous y políticas



Windows applocker

- **Controlar archivos ejecutables (.exe y .com), scripts (.js, .ps1, .vbs, .cmd y .bat), archivos de Windows Installer (.msi y .msp) y archivos DLL (.dll y .ocx).**
- **Definir reglas basadas en atributos de archivo derivados de la firma digital, incluido el publicador, el nombre de producto, el nombre de archivo y la versión del archivo. Por ejemplo, puede crear reglas basadas en el atributo de publicador que se ha conservado a lo largo de las actualizaciones, o puede crear reglas para una versión específica de un archivo.**
- **Asignar una regla a un grupo de seguridad o a un usuario individual.**
- **Crear excepciones a las reglas. Por ejemplo, puede crear una regla que permita la ejecución de todos los procesos de Windows salvo el Editor del Registro (Regedit.exe).**
- **Utilizar el modo de solo auditoría para implementar la directiva y conocer su impacto antes de aplicarla.**
- **Simplificar la creación y administración de reglas de AppLocker utilizando cmdlets de Windows PowerShell.**

- **Permite dar permisos privilegiados en forma restringida.**
- **Ejemplo:**

```
$ getcap /usr/bin/dumpcap  
/usr/bin/dumpcap =  
    cap_net_admin,cap_net_raw+eip
```

- **Man capabilities para ver lista y funcionalidad**

- El programa sudo (de las siglas en inglés de *superuser* -o *substitute user- do*) es una utilidad de los sistemas operativos tipo Unix, como Linux, BSD, o Mac OS X, que permite a los usuarios ejecutar programas con los privilegios de seguridad de otro usuario (normalmente el usuario root) .

- Todo un mundo. Distintos niveles de abstracción.
- Como evitar que desde el contenedor o la virtual se pueda atacar al host.
- Secure Supply Chain
- Cuidado con las múltiples VLANs.



SE-Linux

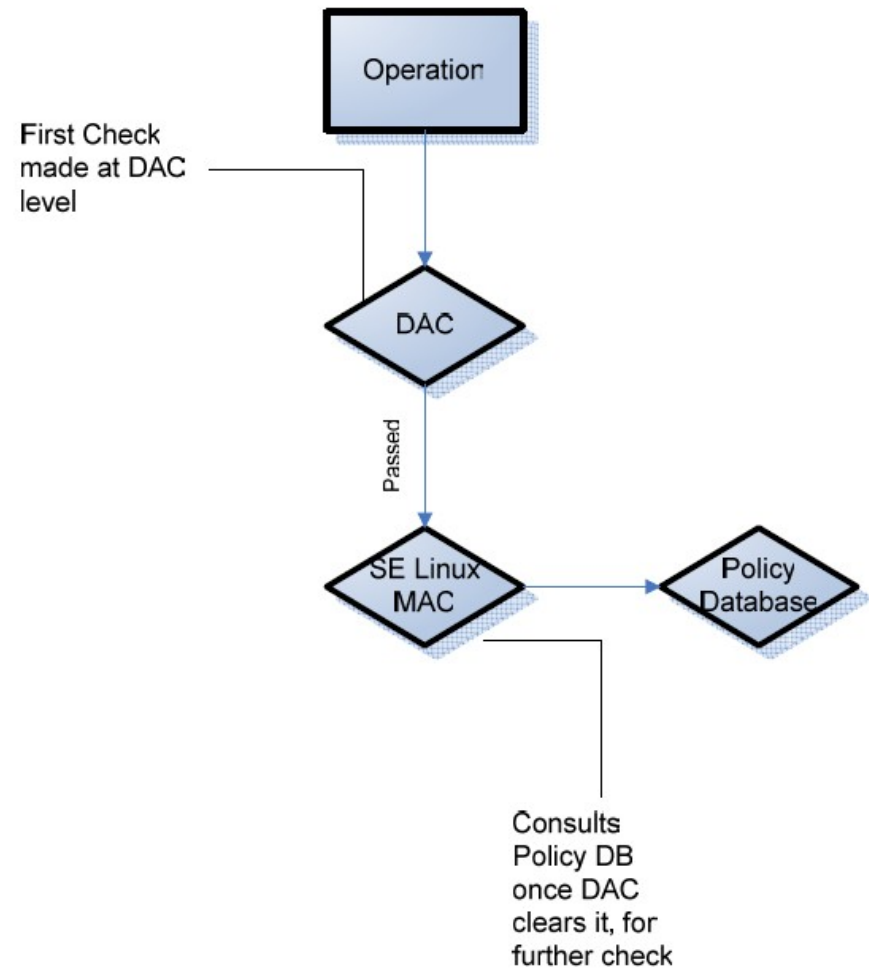
- **Security Enhanced Linux (SELinux) es una extensión al Kernel de Linux que fue diseñada para forzar políticas de control de acceso estrictas (MAC).**
- **Difundido por la NSA en el 2000, y creado con el aporte de, entre otros, NAI Labs, Secure Computing Corporation y MITRE Corporation.**
- **El código es liberado a la comunidad Open Source.**
- **Integrado al Kernel 2.6 de Linux.**

- **DAC (Discretionary Access Control)**
 - usado por unix standard,
 - Las decisiones de acceso se basan en identidad y propiedad.
 - Cada usuario tiene control absoluto sobre sus procesos y archivos. Los procesos heredan los derechos del usuario.
 - No se controla el flujo de datos si los usuarios tienen poder absoluto sobre sus objetos.

- **MAC (Mandatory Access Control)**
 - Reglas de acceso explícitas.
 - Definidas por el administrador de manera global y forzadas por el sistema.
 - Reglas de separación
 - Fuerzan restricciones de acceso a los datos
 - Establecen roles bien definidos
 - Reglas de Integridad
 - Evitan modificaciones no autorizadas a datos o aplicaciones

Como Funciona

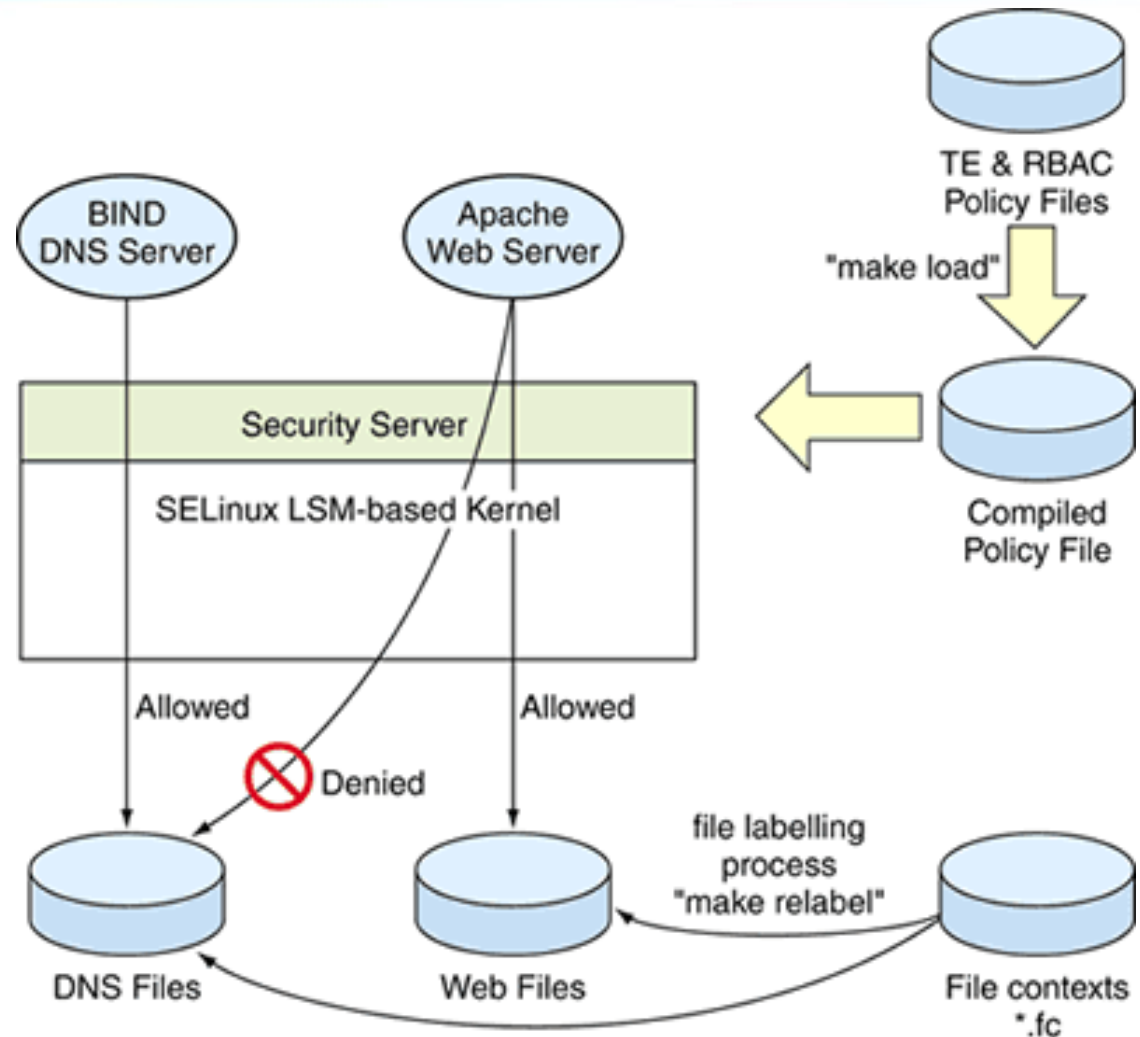
- **Conceptos: Sujetos, Objetos y Operaciones. Los objetos a su vez se pueden agrupar en “Object classes”.**
- **El DAC original de Linux tiene prioridad sobre MAC**
- **Si el acceso es denegado por los permisos tradicionales, SELinux ni entra en juego.**



- **SE Linux se implementa como un “Linux Security Module” (LSM) en el cual se insertan hooks en el kernel que hacen que cada operación de un proceso sea validada contra SELinux.**
- **Para ejecutar un programa en forma segura en SELinux el administrador debe conocer todos los archivos y subprocesos afectados por el mismo y de que manera actúa contra ellos.**

Como Funciona

- Dominios. Las aplicaciones se confinan en estrictos dominios separados del resto.



- Type Enforcement (TE) – Mecanismo Primario
- RBAC
- Multi-Level Security (MLS)
- Los procesos y archivos tienen un contexto de seguridad:

kmacmill:staff_r:firefox_t:s0

kmacmill:object_r:user_home_t:s0

usuario:rol:tipo:nivel

- **El tipo se aplica a procesos y recursos. Cuando se aplica a procesos se habla de “dominios”**
- **Representa toda la información relevante de seguridad**
- Apache processes > httpd_t
- /var/www/html/index.html > httpd_sys_content_t
- **El acceso es permitido entre tipos**
- Ej: allow httpd_t httpd_sys_content_t : file read;

Transición

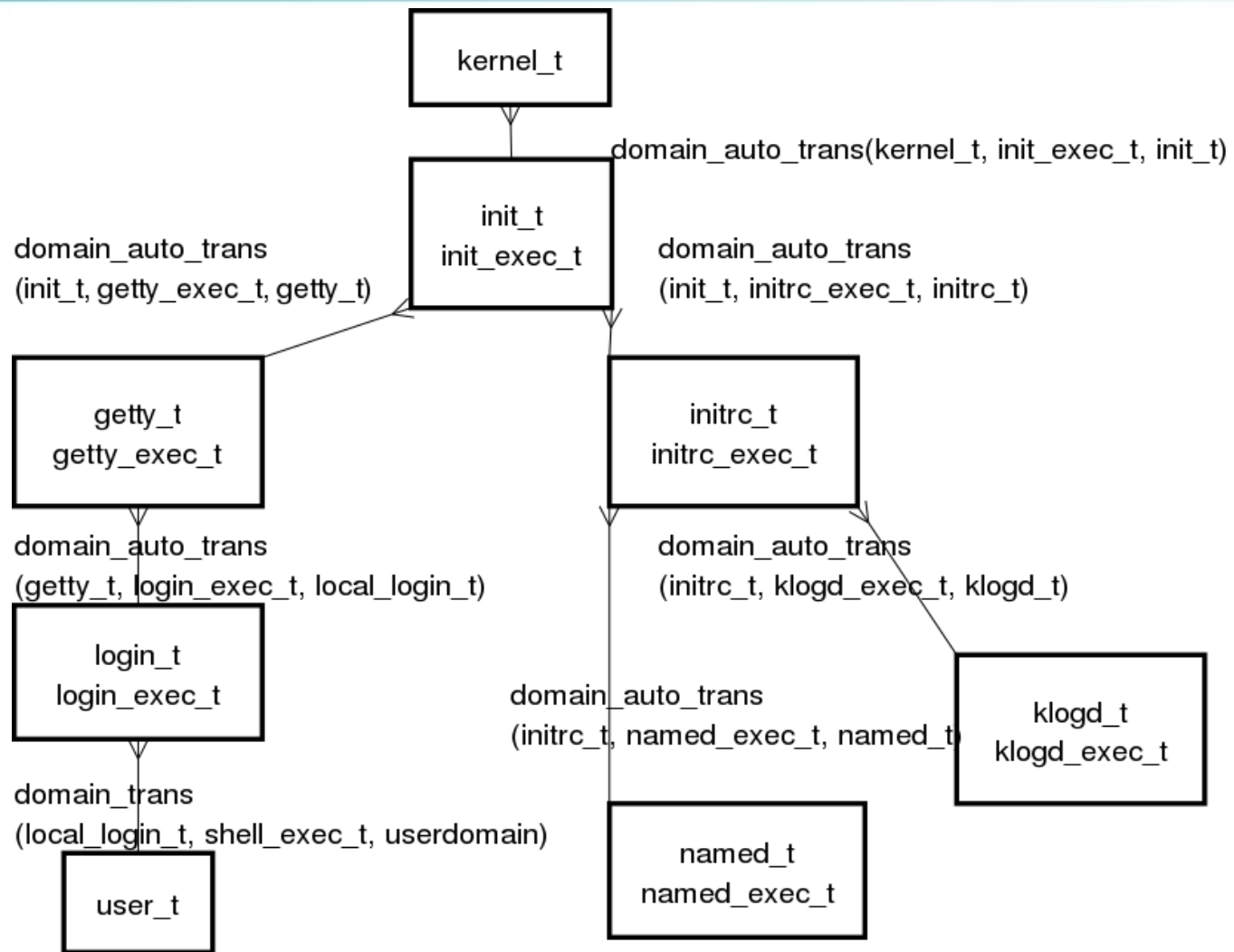
- **de Proceso**

- igual al padre: por ejemplo un comando desde un editor.
- Nuevo dominio, de acuerdo a la política: por ej cuando init lanza un nuevo demonio ftpd.

- **de archivo**

- igual al directorio en que se crea.
- Nuevo dominio, de acuerdo a la política. (temporarios creados por un proceso pertenecen a un tipo dado)

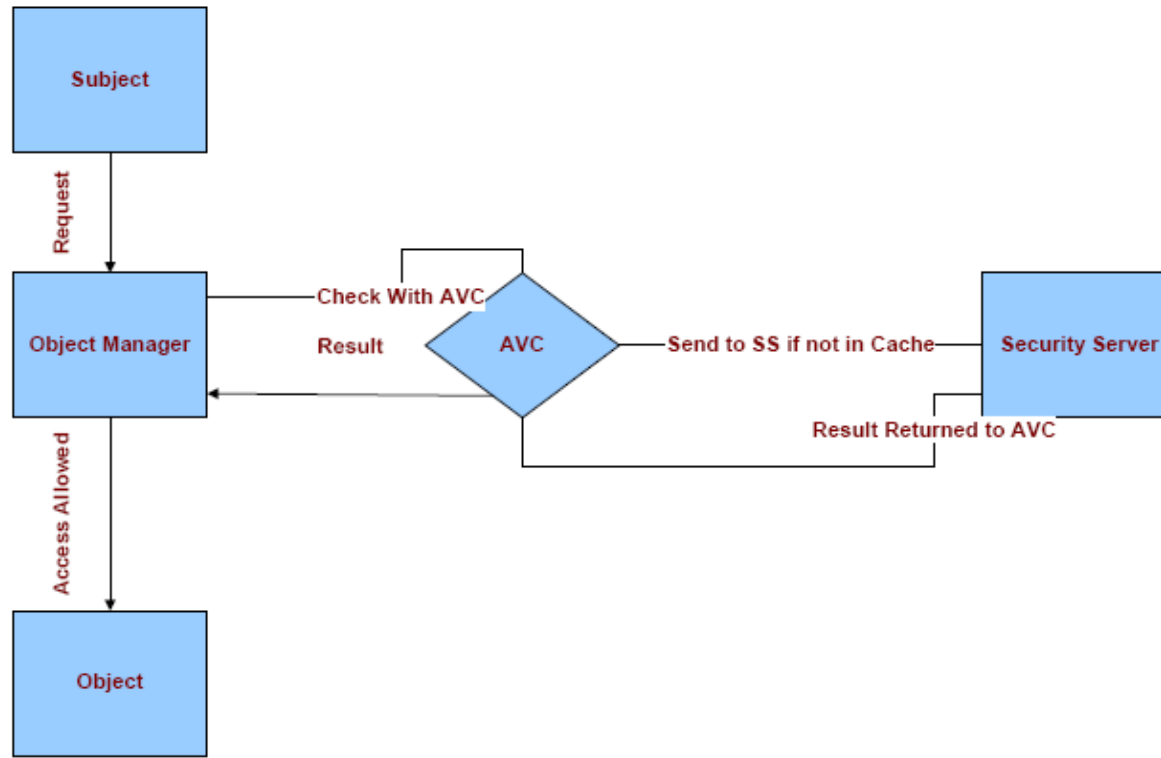
Transiciones:



- **Security Server: componente que fuerza el cumplimiento de las políticas**
 - toma decisiones de seguridad basado en el contexto de seguridad (conjunto de atributos asociados a un objeto)
- **Object Managers: manejan los atributos de seguridad de los objetos**

- **El Security context (SC) de un objeto tiene la forma:**
 - usuario:rol:tipo:nivel
 - “system_u:object_r:inetd_exec_t:s0”
 - El campo clave es el tipo!
- **El security server asigna un contexto a cada proceso, compuesto por:**
 - reglas, proceso padre y user ID
 - las reglas dicen que es lo que puede hacer.
 - Ante cada acceso, el Sec.Server. chequea los permisos. A diferencia del unix standard que es solo al abrir.

- Como el SS es consultado en cada acceso se usa el Access Vector Cache (AVC) para optimizar.



- **SELinux es el mismo independientemente de la distribución de Linux que utilice.**
- **Cambia la forma de especificar/configurar reglas**
- **Cambian las reglas incluidas**
- **Cambian las herramientas/comandos que son SELinux-aware**

- **APLICACIONES**
- **Gran parte de aplicaciones y servidores no cambian**
- SELinux aware
 - Aplicaciones que manipulan o cambian contextos
 - Programas que setean contextos de sesiones:
login/sshd, ls, cp, ps, setfilecon, logrotate, cron ...

- **POLITICAS**
- Strict – (no soportada en general)
- **Todo es negado por default**
- **Se deben especificar reglas para dar privilegios.**
- **SELinux está diseñado para ser Strict.**
 - Las reglas no manejan el concepto de deny
 - privilegios mínimos por cada daemon
 - dominios separados por programas GPG, X, ssh, etc
- **Muy difícil de forzar en un sistema operativo de uso general.**

- **POLITICAS**
- Targeted
- **Definida como una alternativa al strict.**
- **Apunta a restringir los servicios más importantes o potencialmente vulnerables**
- **Por default los procesos corren en unconfined_t**
 - tienen los mismos permisos como si SELinux no existiese
- **Los dominios con políticas definidas hacen una transición a un dominio definido:**
 - httpd comienza con unconfined_t y pasa a httpd_t que tiene accesos más limitados

Dominios predefinidos

- **En RHEL4**
 - **15 dominios definidos**
 - **httpd, squid, Mailman, Named, dhcpd, mysqld, nscd, ntpd, portmap, postgresql, snmpd, syslogd, winbindd, etc.**
- **En RHEL5**
 - **200 dominios definidos**
 - **Todo programa incluido por Red Hat y que se inicia al bootear, debe tener un dominio definido.**
 - **Limitadas restricciones a los usuarios**

- SELinux guarda la configuración en /etc/selinux

```
ls -l /etc/selinux
-rw-r--r-- 1 root root 515 Jan 18 11:46 config
drwxr-xr-x 7 root root 4096 Jan 23 14:06 strict
drwxr-xr-x 7 root root 4096 Jan 23 14:06 targeted
```

- /etc/selinux/config identifica la política y el modo

```
more /etc/selinux/config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=targeted
```

- **enforcing** Fuerza las políticas.
- **permissive** Se avisa el incumplimiento de reglas, pero se deja continuar.
- **disabled** No se activan las reglas de SELinux.

- **Z** es tu amigo! (conocido como `-context`)
- Core Utilities
 - `ls -Z`
 - `cp /mv/ install`
 - `find / -context=`
 - `id -Z`
 - `ps auxZ`
 - `netstat -Z`
- Login – PAM – password
 - `ssh, su, login, xdm, sudo`
 - `passwd, useradd, groupadd`
- `rpm`

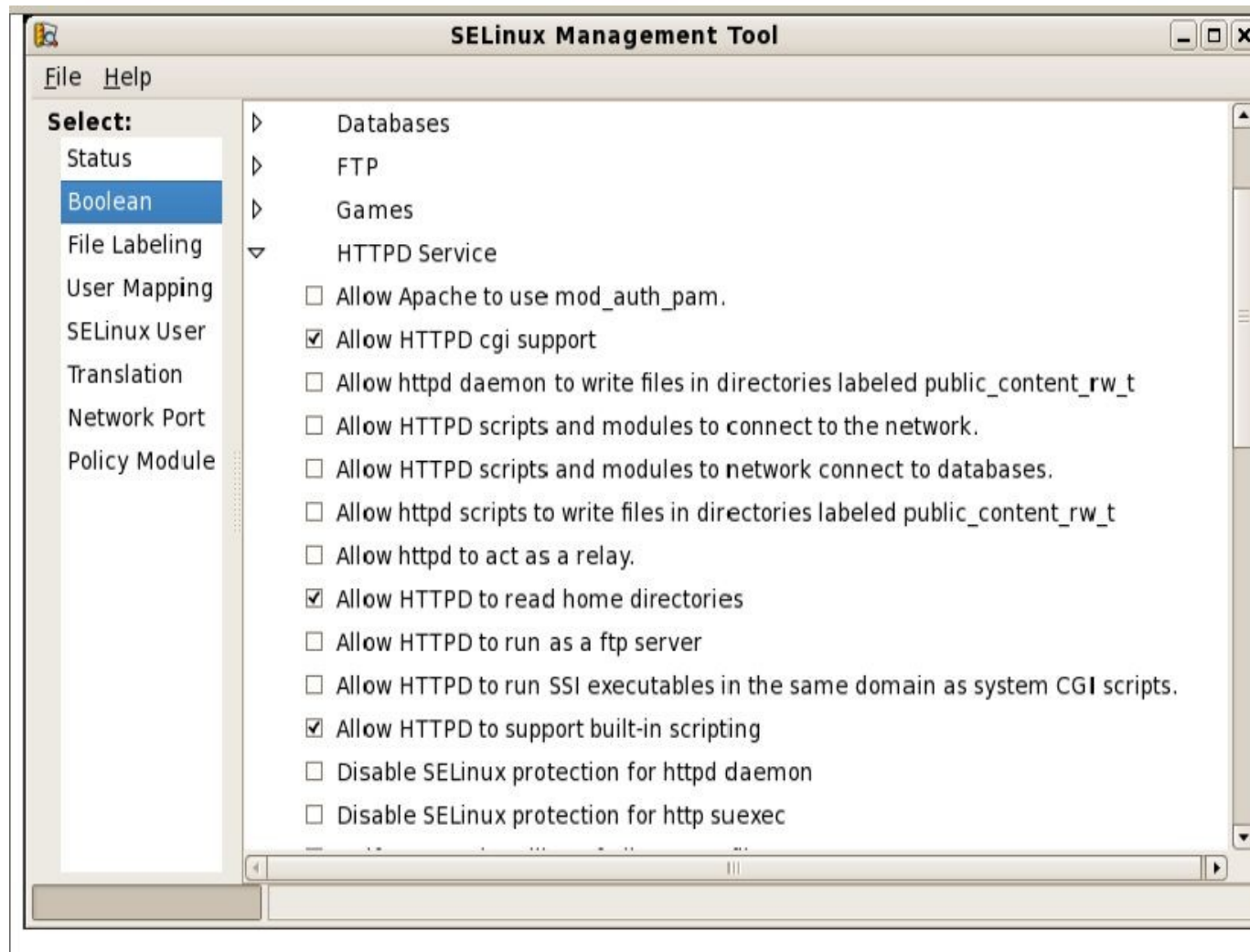
- Policycoreutils
 - newrole
 - run_init
 - audit2allow, audit2why
 - secon
 - sestatus
- libselinux
 - getenforce/setenforce
 - selinuxenabled

- Manejo de Booleans
 - setsebool, getsebool, system-config-securitylevel
 - setsebool -P httpd_enable_homedirs=1
- Manejo file context
 - setfiles, restorecon, fixfiles, genhomedircon, chcon

- audit2allow
 - Genera políticas a partir de mensajes de error del log
 - audit2allow -i /var/log/audit/audit.log
 - `allow system_crond_t null_device_t:chr_file { read write };`
- audit2why
 - Traduce mensajes de error del log.

- **Booleanos son instrucciones if/then/else en la política**
- **Permiten configurar la política sin tener que editarla.**
- **Getsebool para ver los seteos**
 - `getsebool -a`
- **Setsebool para modificar un valor**
 - `setsebool -P httpd_enable_homedirs 1`
- **`man httpd_selinux`**

SELINUX GUI – system-config-selinux



- chcon
 - Utilidad fundamental para cambiar el contexto de un archivo
 - `chcon -R -t httpd_sys_script_rw_t /var/www/myapp/data`
 - `chcon -t httpd_sys_script_t /var/www/cgi-bin/myapp`
 - Parecido al `chmod`
 - `-t` type qualifier
 - `customizable_types`
 - `/etc/selinux/targeted/contexts/customizable_types`

Shellshock

This is perhaps best explained with some lovely pictures. Without SELinux, if your webserver's apache daemon is compromised, your whole server is compromised:



With SELinux, your apache process is compromised, but it is restricted to only accessing the data an apache process should ever want to read:

