Unidad 5: PROTECCIÓN vs SEGURIDAD

BLOQUE II – Sistemas, aplicaciones y personas

CONTENIDOS

1. Introducción.
2. Mecanismos de control de acceso.
3. Autenticación de usuario.
4. Sistemas operativos de confianza.

1. Introducción

- Como ya sabemos, las funcionalidades de un sistema operativo pueden resumirse en dos:

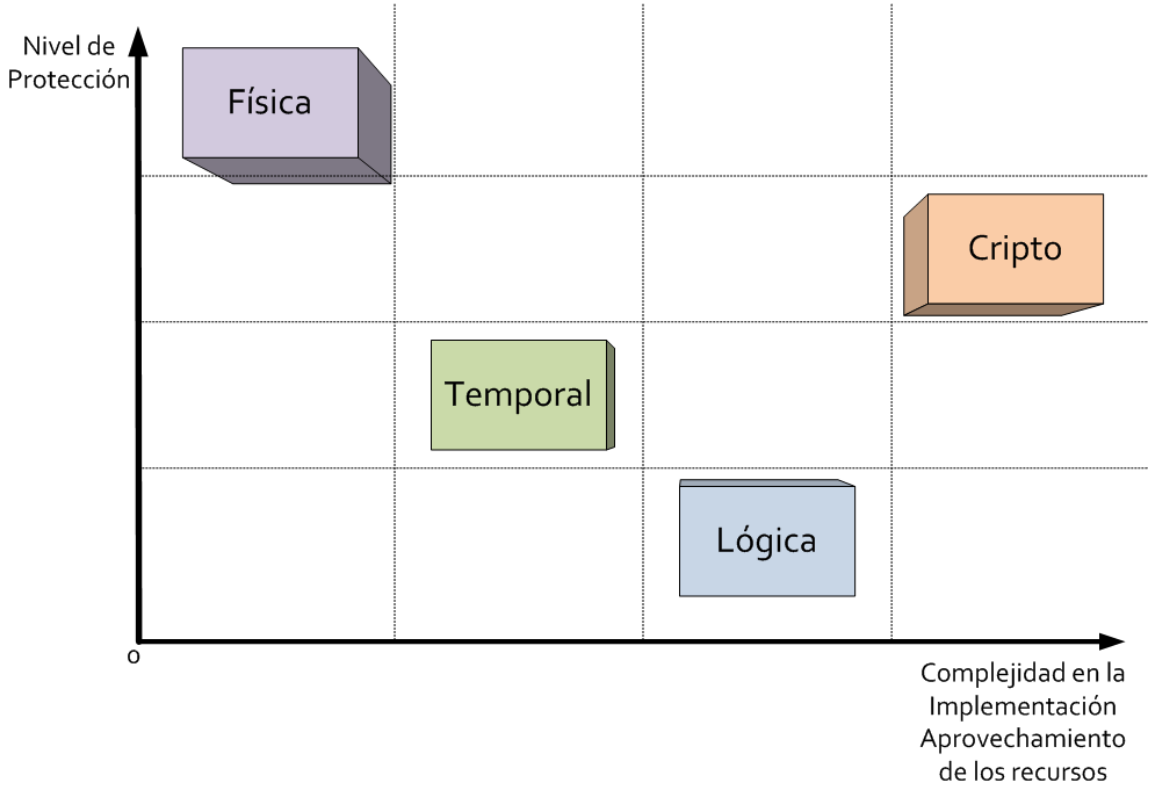
Intermediar en el a los recursos hardware disponibles	Proporcionar el interfaz que posibilita este acceso a usuarios y aplicaciones

- La primera de estas funciones está directamente relacionada con lo que denominamos protección.

1. Introducción

- En primer lugar es necesario comprender la diferencia entre protección y seguridad:
  - **Protección** – Supone el diseño e implementación de técnicas y mecanismos orientados a la protección delos recursos destinados a un usuario/procesos frente a la intervención inadvertida o maliciosa de otros usuarios/procesos.
  - **Seguridad** – Supone el diseño, implementación y adopción de técnicas, mecanismos, herramientas y modelos que aseguren la confidencialidad, la integridad y la disponibilidad de los activos de una organización.

- La protección empleada en la mayor parte de los sistemas operativos actuales se basa en la separación, que se puede enfocar de diferentes formas:



1. Introducción

- Pregunta: ¿Puedes poner un ejemplo de cada uno de estos tipos de separación? Piensa en el sistema de ficheros y el disco duro, por ejemplo.

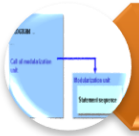
1. Introducción

- Los recursos que se comparten y que por lo tanto es imprescindible proteger se pueden resumir en:

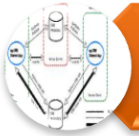

Memoria


Dispositivos E/S compartidos en serie


Dispositivos E/S compartidos en paralelo


Programas y subrutinas


Redes


Datos e información

1. Introducción

- Aunque la separación constituye una medida efectiva de protección, analizando los recursos involucrados se comprende que no puede ser una separación completa en todos los casos.
  - Un sistema operativo proveerá diferentes niveles de protección para diferentes objetos, usuarios o situaciones.
- Por este motivo se contemplan distintos grados de separación/compartición.
  - Estáticos y dinámicos.

1. Introducción

1. Introducción

- Pero no basta con garantizar esta separación entre usuarios/procesos, además es necesario proporcionar:
  - Autenticación, gestión de identidades y credenciales.
  - Gestión de logs y procedimientos de auditoría.
  - Cuantificación de la utilización de los recursos compartidos.
  - Etc.

2. Mecanismos de control de acceso

- El sistema operativo funciona como autoridad central que controla el acceso (siguiendo el principio de separación/compartición ya mencionado) a los recursos compartidos.
- En algunos casos, como ocurre con la memoria, este control de acceso está muy apoyado en el propio hardware.

Asignaturas relacionadas con Sistemas Operativos y con Arquitectura de Computadores a lo largo de todo el Grado

2. Mecanismos de control de acceso

- Pero el control de acceso a otros objetos por parte del “sujeto” o “usuario” (que puede ser una persona, un programa, un fichero, en general, cualquier otro objeto) recae completamente sobre el SO.
- Que además debe garantizar el acceso con mínimo privilegio a estos objetos.
  - Principio de mínimo privilegio: Postulado que requiere que los sujetos o usuarios tengan habilitado, exclusivamente, el derecho de acceso a los objetos que ineludiblemente requieran para cumplir sus funciones, y en el modo (lectura, escritura, etc.) más conservador que sea necesario. ¿Recuerdas?

2. Mecanismos de control de acceso

Discretionary Access Control (DAC)

- Las políticas que definen el control de acceso pueden ser modificadas por los propietarios de los objetos.
- Por ejemplo, la asignación de permisos R-W-X en un sistema de tipo Unix.

Mandatory Access Control (MAC)

- Las políticas se gestionan de manera centralizada por un administrador de seguridad.
- Los usuarios no tienen posibilidad de modificarlas.

2. Mecanismos de control de acceso

Directorios

Listas de control

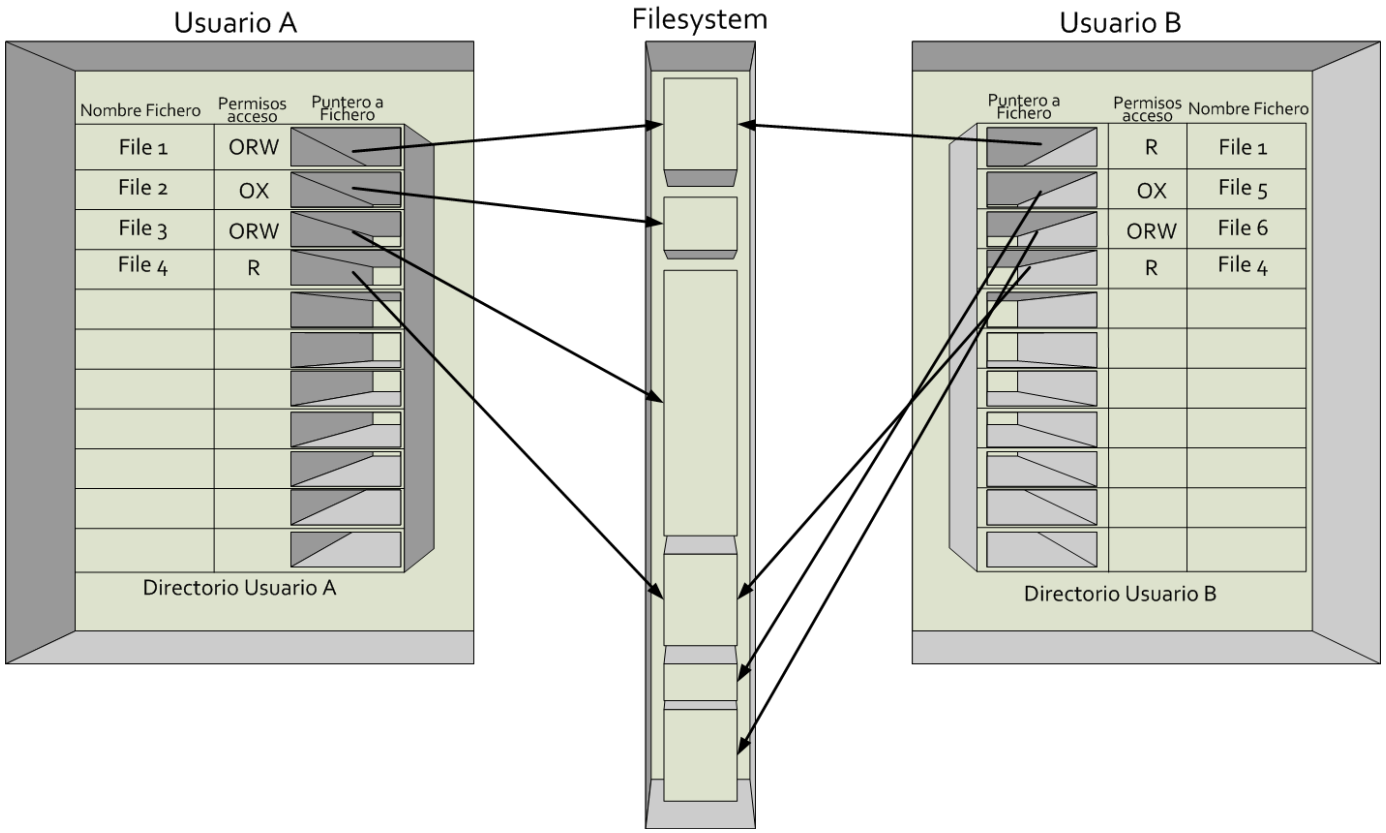
Matrices de control

Roles

2. Mecanismos de control de acceso

- **Directorios:**
  - Se puede utilizar un mecanismo sencillo similar al de los directorios de ficheros.
  - La idea es que cada objeto tiene un usuario propietario (Owner, O) con derecho de gestionar el control de acceso.
    - Puede dar permisos y retirarlos a otros usuarios.
  - Cada usuario tiene un directorio con la lista de objetos a los que tiene acceso con el modo de acceso permitido.
  - El SO mantiene estos directorios y toda su información actualizada.
    - Los usuarios no pueden escribir sobre ellos directamente.

Ejemplo con objetos ficheros, dos usuarios y modos de acceso O, R, W y X



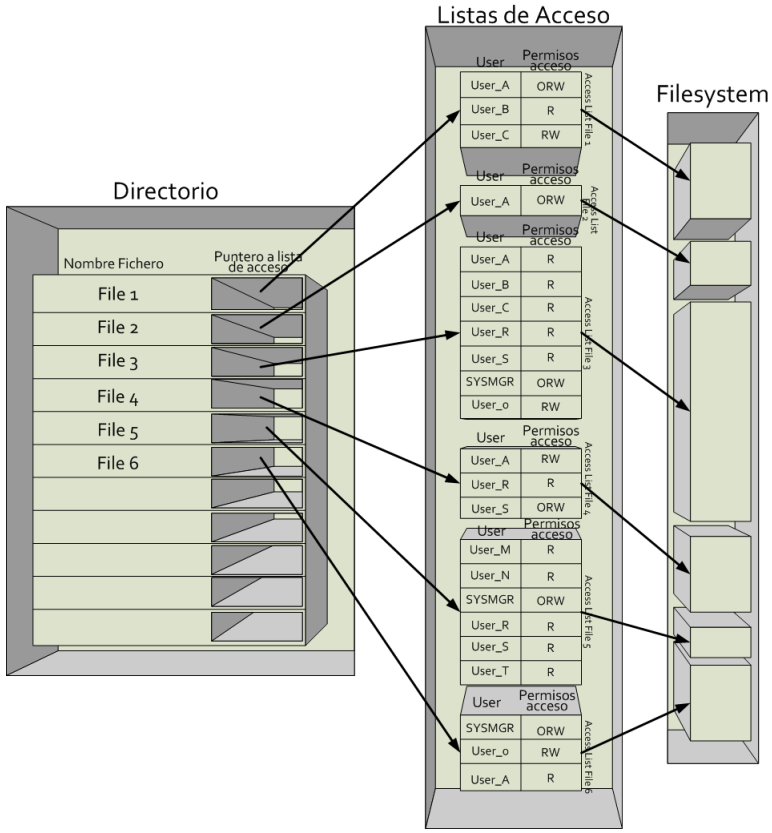
2. Mecanismos de control de acceso

- Esta solución es fácil de implementar pero tiene serias desventajas:
  - Para mantener la coherencia de la solución cada directorio debe tener una entrada por cada objeto del sistema, aunque el usuario correspondiente no tenga la intención de acceder a él.
    - Esto aumenta excesivamente el tamaño de las estructuras necesarias para el control del acceso.
  - Las latencia de ciertas operaciones son demasiado altas.
    - Por ejemplo, si el SO tiene que revocar los permisos de acceso de todos los usuarios a un objeto determinado.
  - Surgen problemas de nombrado (pseudónimos, aliasing, etc.).

2. Mecanismos de control de acceso

- **Listas de control (Access Control List o ACL):**
  - Se puede un mecanismo similar pero que emplee una estructura por objeto en lugar de una por usuario.
  - La idea en este caso es mantener una lista de control de acceso que almacene la información sobre los usuarios que pueden acceder a cada objeto y su modo de acceso permitido.
  - Este mecanismo es más manejable que el de directorios (en cuanto tamaño de las estructuras) pero todavía es demasiado inflexible en muchos casos de uso.

Si seguimos con el ejemplo de los ficheros pero añadimos más usuarios:



2. Mecanismos de control de acceso

- **Matrices de control:**
  - Los mecanismos de directorio y lista de control son complementarios y cada uno de ellos puede dar un rendimiento mejor en situaciones diferentes.
  - Las matrices de control combinan ambas representaciones de la información.
  - Pero se utilizan poco ya que son matrices dispersas y por lo tanto, su utilización suele ser poco eficiente.
    - Muchas celdas de la matriz está vacías.

Matriz de Control de Accesos								
	File 1	File 2	File 3	File 4	File 5	File 6	File 7	File 8
USER A	ORW	ORW	ORW	R	X	X	R	W
USER B	R	-	-	R	X	X	R	W
USER S	RW	-	R	R	X	X	R	W
USER T	-	-	-	R	X	X	R	W
SYS_MGR	-	-	-	RW	OX	OX	ORW	O
USER 0	-	-	-	O	X	X	R	W

Triplas <Usuario, Objeto, Modo de acceso permitido>

2. Mecanismos de control de acceso

- Roles (Role Based Access Control o RBAC):
  - Este mecanismo de control de acceso es complementario a cualquiera de las soluciones estudiadas hasta ahora.
  - Permite distinguir entre diferentes tipos de usuarios, agruparlos y asignar privilegios en función de la pertenencia a dichos grupos.
    - Esto puede simplificar enormemente la gestión del control de acceso.

2. Mecanismos de control de acceso

- Los sistemas operativos con los que trabajáis habitualmente (Linux y Windows) funcionan:

Objetos internos

ACLs y MAC (administrador de seguridad)

En el caso de Windows, RBAC

Objetos en red

Capacidades (Kerberos)

Asignaturas relacionadas con las Redes

3. Autenticación de usuarios

- La autenticación es el proceso que permite verificar la identidad digital del remitente de un mensaje o de una petición.
  - Imprescindible en los actuales SO multiusuario.
- Los mecanismos de autenticación deben ser fiables, económicamente factibles, y aceptables para los usuarios (que normalmente tienen que emplearlos con cierta frecuencia).
  - Hoy en día los entornos son en mucho casos single-sign-on.

3. Autenticación de usuarios

- Los métodos de autenticación suelen dividirse en tres categorías:
  - Sistemas basados en algo conocido (palabra clave, por ejemplo).
  - Sistemas basados en algo poseído (una llave o tarjeta, por ejemplo).
  - Sistemas basados en algo que se es (huella dactilar o iris ocular, por ejemplo) o en lo que se hace.
    - Esto es más bien una identificación que una autenticación.

3. Autenticación de usuarios

- Dentro de la primera categoría suelen encontrarse todos los sistemas de autenticación "informática" habituales.
  - Aunque en algunos entornos se utilizan con esta finalidad smartcards, llaves USB o mecanismos biométricos, todavía no suele ser lo habitual.
    - Se van incorporando en SO móviles o empotrados.
- Y la mayor parte de los SO utilizan una palabra clave o contraseña.

3. Autenticación de usuarios

- La elección de este tipo de autenticación está basada sobre todo en el criterio de simplicidad, porque no está exenta de complicaciones y limitaciones.
  - Revelación/robo de contraseñas.
  - Pérdida/olvido de contraseñas.
  - Rotura de contraseña.
    - Contraseñas probables, ataques de diccionario, ataques de fuerza bruta.
- Por eso los administradores fuerzan cada vez más a escoger contraseñas seguras.

Contraseña segura = que contenga todo tipo de caracteres, larga, que no sea una palabra del diccionario, que no tenga nada que ver con el usuario, que se refresque cuando sea necesario, que no se re-utilice, no escrita y no dicha, etc.

What Are the 50 Most Common Passwords?

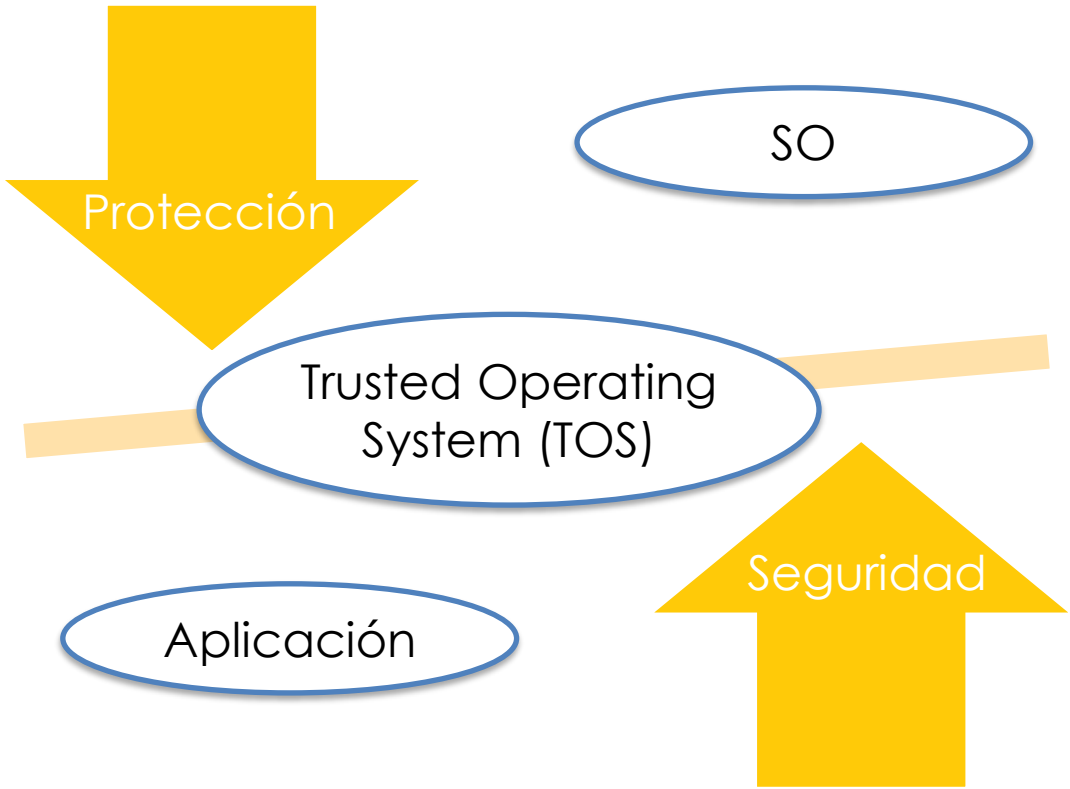
Based on most common duplicate passwords within a breach of over 30 million accounts.

SecurityScorecard

1.	123456	11.	123321	21.	222222	31.	333333	41.	password1
2.	123456789	12.	1q2w3e4r5t	22.	112233	32.	123qwe	42.	q1w2e3r4
3.	qwerty	13.	iloveyou	23.	abc123	33.	159753	43.	qqww1122
4.	password	14.	1234	24.	999999	34.	q1w2e3r4t5y6	44.	sunshine
5.	1234567	15.	666666	25.	777777	35.	987654321	45.	zxcvbnm
6.	12345678	16.	654321	26.	qwerty123	36.	1q2w3e	46.	1qaz2wsx3edc
7.	12345	17.	555555	27.	qwertyuiop	37.	michael	47.	liverpool
8.	1234567890	18.	gfhjkm	28.	888888	38.	lovely	48.	monkey
9.	111111	19.	7777777	29.	princess	39.	123	49.	1234qwer
10.	123123	20.	1q2w3e4r	30.	1qaz2wsx	40.	qwe123	50.	computer

https://securityscorecard.com/blog/worlds-worst-passwords

4. Sistemas operativos de confianza



4. Sistemas operativos de confianza

- Normalmente se habla de SO de confianza por los matices que distinguen la seguridad y la confianza (trust):

La seguridad parecería binaria (seguro/no seguro) y generaría falsas expectativas.

La confianza se puede gestionar por grados y se puede evaluar de manera relativa, teniendo en cuenta el contexto de uso.

4. Sistemas operativos de confianza

- No existe una normativa estándar para los TOS (Trusted Operating Systems).
- Normalmente para decidir si un SO es de confianza o no se utiliza el Common Criteria.
  - Trabajo conjunto de los gobiernos de EEUU, Canadá, Reino Unido, Francia, Alemania y Holanda para armonizar un conjunto de criterios que permitan evaluar la seguridad de productos TIC (ISO/IEC 15408).

4. Sistemas operativos de confianza

- Algunos ejemplos de TOS con diferentes grados de confianza (casi todos EAL 4) son:
  - Microsoft Windows 7 y Microsoft Server 2008 R2.
  - Algunas distribuciones de Linux que han incorporado desde el kernel 2.6 el módulo de la NSA SELinux.
  - Apple Mac OS X 10.6.
  - HP-UX 11i v3.
  - Trusted Solaris
  - AIX 5L con PitBull Foundation.
  - XTS-400.
  - IBM VM (SP, BSE, HPO, XA, ESA, etc.) con RACF.

Para practicar un poco

1. Revisa cómo se asigna propietario a un fichero (por ejemplo, es el recurso más sencillo) en el sistema operativo que usas, cómo se le asignan permisos a otro tipo de usuarios, si existe algún tipo de herencia, si se utilizan roles de algún tipo, etc.
2. Investiga también qué tipo de logs mantiene tu sistema operativo por defecto y cómo se pueden configurar auditorías adicionales. ¿Dónde se guardan todos estos logs? ¿Cómo se pueden consultar? ¿Se pueden consultar, agrupar, exportar, etc. fácilmente?
3. Intenta construir una herramienta sencilla que compruebe la seguridad de una contraseña (de nuevo, no hace falta que programes si no quieres, puedes hacerlo con un Excel). Si no, prueba alguna de las que están disponibles on-line como http://www.passwordmeter.com/ ó http://password-checker.online-domain-tools.com/

Referencias

- Fotografías
  - https://unsplash.com
- Iconos
  - https://www.flaticon.es/