Sicurezza

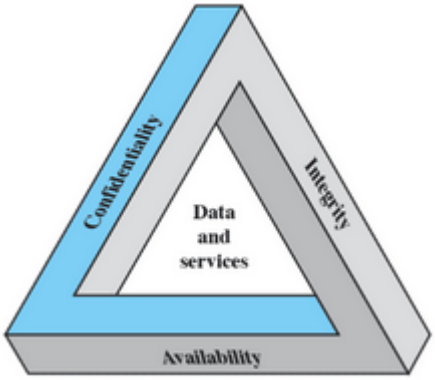
Index

Introduction

Iniziamo riportando la definizione di **sicurezza informatica** del NIST (National Institute of Standards and Technology) ovvero: "è la protezione offerta da un sistema

informativo automatico al fine di conservare integrità, disponibilità e confidenzialità delle risorse del sistema stesso”

La triade



Dunque ci sono tre obiettivi che costituiscono il cuore della sicurezza:

- **integrità**
- **disponibilità**
- **confidenzialità**

Ci sono due ulteriori obiettivi che vengono aggiunti al nucleo della sicurezza informatica:

- autenticità
- tracciabilità

Gli obiettivi nel dettaglio

Analizziamo ora i tre obiettivi più nel dettaglio:

- **Integrità** → riferita tipicamente ai dati, che non devono essere modificati senza le dovute autorizzazioni
- **Confidenzialità** → riferita tipicamente ai dati, che non devono essere letti senza le dovute autorizzazioni
- **Disponibilità** → riferita tipicamente ai servizi, che devono essere disponibili senza interruzioni
- **Autenticità** → riferita tipicamente agli utenti ,che devono essere chi dichiarano di essere (per estensione vale anche per messaggi e dati)

Minacce (threats)

L'**RFC 2828** descrive quattro conseguente delle minacce informatiche

- accesso non autorizzato (*unauthorized disclosure*)
- imbroglio (*deception*)
- interruzione (*disruption*)
- usurpazione (*usurpation*)

Accesso non autorizzato

Si verifica un accesso non autorizzato quando un'entità ottiene l'accesso a dati per i quali non ha autorizzazione. Ciò costituisce una minaccia alla confidenzialità

Tipicamente gli attacchi ad un SO che riescono ad ottenere un accesso non autorizzato sono:

- esposizione (intenzionale o per errore) → ciò che dovrebbe essere privato è invece pubblico
- intercettazione → attaccante che si mette in mezzo ad una comunicazione
- inferenza → riesco a dedurre alcuni dati dai dati pubblici
- intrusione → attaccante riesce ad entrare direttamente in un sistema

Imbroglio

Avviene un imbroglio quando un'entità autorizzata riceve dati falsi e pensa che siano veri. Ciò costituisce una minaccia all'integrità

Questo tipo di minaccia può avvenire per:

- mascheramento → l'attaccante riesce ad entrare in possesso delle credenziali di un utente autorizzato (trojan)
- falsificazione (es. uno studente che modifica i propri voti)
- ripudio → quando un utente nega di aver ricevuto o inviato dati

Interruzione

L'interruzione consiste nell'impedimento al corretto funzionamento dei servizi, e costituisce una minaccia all'integrità del sistema o alla disponibilità

Questo tipo di minaccia può avvenire per:

- incapacitazione → rompendo qualche componente del sistema

- ostruzione → Denial of Service (DoS), per esempio riempiendo il sistema di richieste
- corruzione → alterazione dei servizi

Usurpazione

Si parla di usurpazione quando il sistema viene direttamente controllato da chi non ne ha l'autorizzazione. Ciò costituisce una minaccia all'integrità del sistema

Questo tipo di minaccia può avvenire per:

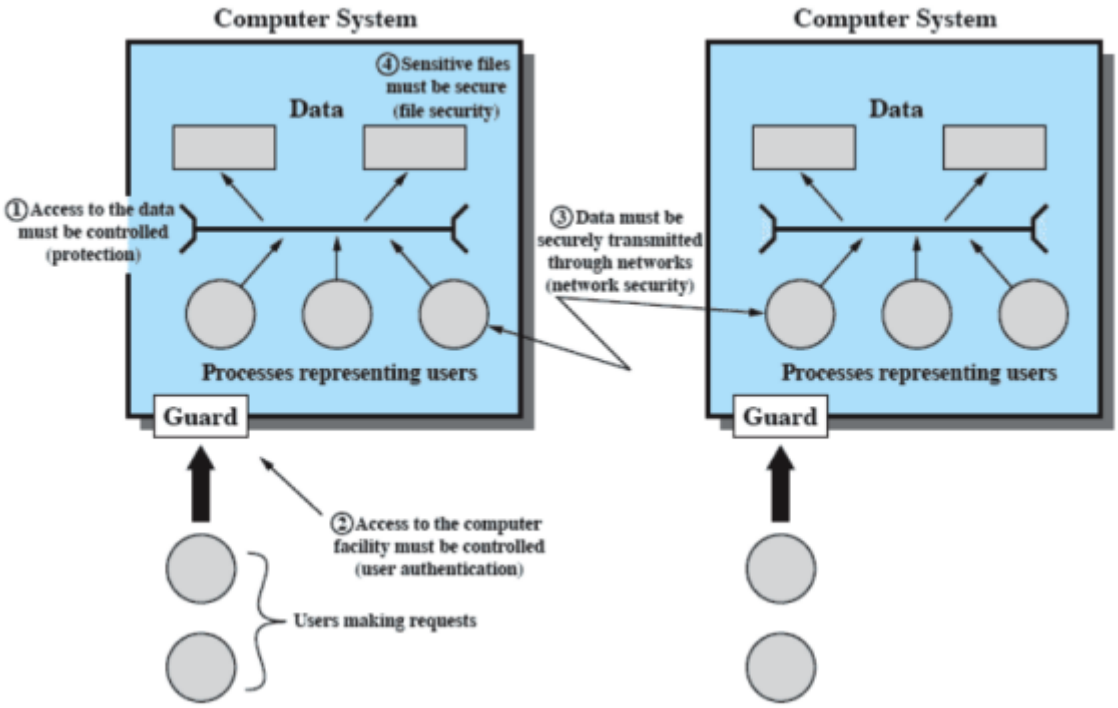
- attacchi → appropriazione indebita (diventare amministratore di una macchina non propria, es. le macchine che compongono le botnet per poter poi fare DoS)
- uso non appropriato → virus che cancella file o fa danni

Asset

Un'altra cosa da considerare quando si parla di sicurezza sono gli asset. Gli asset consistono nelle **risorse da proteggere**
Gli asset sono categorizzati come:

- hardware
- software
- dati
- linee di comunicazione e reti

Ambito della sicurezza informatica



Relazione tra Asset e Triade

	Disponibilità	Confidenzialità	Integrità
Hardware	Workstation rubate o rese inutilizzabili		
Software	Programmi cancellati	Copia non autorizzata dei programmi	Modifica dei programmi (per non farli funzionare o per fargli fare compiti indesiderati)
Dati	File cancellati	File letti senza autorizzazione. Dati inferiti da analisi statistica	Modifica di file esistenti o creazione di file
Comunicazione	Messaggi distrutti. Linee di comunicazione rese inutilizzabili	Lettura dei messaggi o osservazione dei pattern	Modifica, ritardo, riordino o duplicazione di messaggi esistenti, creazione di messaggi falsi

---

Autenticazione

L'autenticazione è alla base per la maggior parte dei tipi di controllo di accesso e tracciabilità. Questa consiste in:

- identificazione
- verifica

L’autenticazione serve a determinare **se un utente è abilitato ad accedere al sistema**, e inoltre determina anche i privilegi dell’utente abilitato
Ciò rende possibile il *discretionary control access* (controllo di accesso discrezionale), che consiste nel fatto che un utente può decidere a quali utenti concedere determinati permessi

Mezzi per l’autenticazione

L’autenticazione generalmente si può fare in tre modi (almeno uno deve essere presente, meglio due contemporaneamente):

- qualcosa che *sai* (password)
- qualcosa che *hai* (chiave, badge RFID)
- qualcosa che *sei* (biometrica)

Per sottolineare le possibili problematiche, Nick Mathewson notò come i mezzi per l’autenticazione possano anche essere:

- qualcosa che *hai dimenticato*
- qualcosa che *avevi*
- qualcosa che *eri*

Autenticazione con password

E’ il tipo di autenticazione più nota e usata (spesso anche l’unica). In questo caso l’importante è che le password siano memorizzate non in chiaro

Autenticazione con Token

Riguarda oggetti fisici posseduti da un utente per l’autenticazione e vengono chiamati **token**

Memory card

Possono essere utilizzate solo per memorizzare dati, ma senza elaborarli (es. bancomat), per questo motivo vengono spesso usati insieme a password o PIN

Smartcard

Hanno un microprocessore, memoria e porte I/O. Ne esistono di diversi tipi, a seconda dei seguenti aspetti:

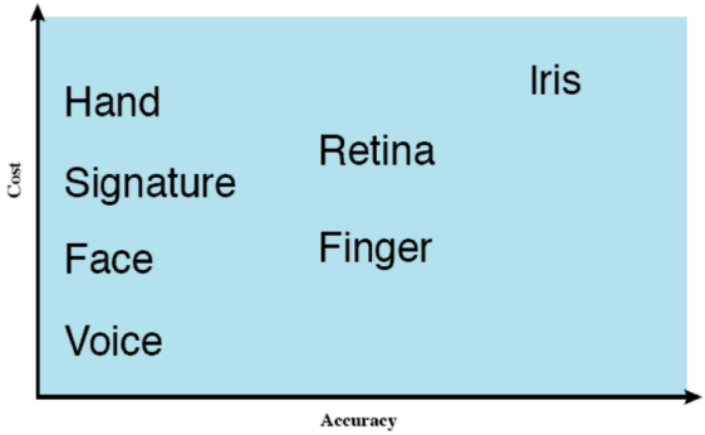
- caratteristiche fisiche → come una carta di credito o una chiavetta USB
- interfaccia → lettore apposito, ma alcune hanno un tastierino

- protocollo di autenticazione → generatore di password statico o dinamico, domanda - risposta

Biometria

Recentemente, la biometria è stata espansa come segue:

- qualcosa che *sei* → biometrica statica: impronta digitale, faccia, … (basata su riconoscimento di pattern, complesso e costoso)
- qualcosa che *fai* → biometria dinamica: scrittura a mano, riconoscimento vocale, ritmo di battitura (i pattern possono cambiare)



Controllo di accesso

Il controllo di accesso serve a determinare quali tipi di accesso sono ammessi, sotto quali circostanze, e da chi
Il controllo di accesso può essere:

- **discrezionale** → un utente può concedere i suoi privilegi ad altri utenti
- **obbligatorio** → un utente non può concedere i suoi stessi privilegi ad altri utenti
- **basato su ruoli**

Le tre modalità possono essere presenti contemporaneamente, ovviamente applicate a diverse classi di risorse

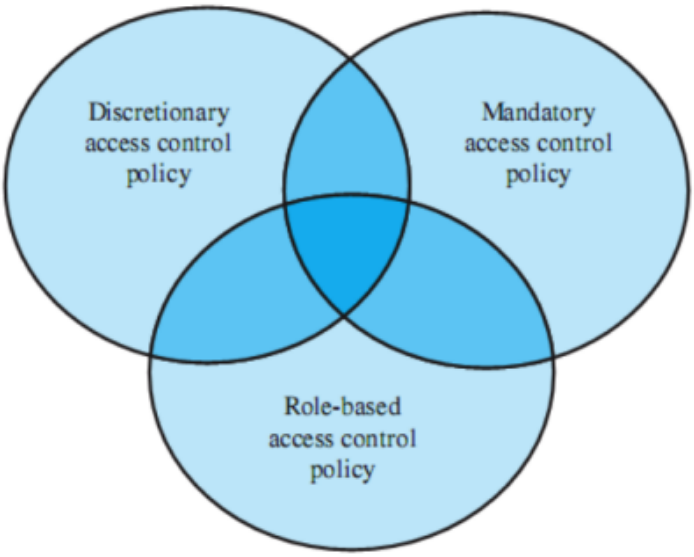


Figure 15.3 Access Control Policies

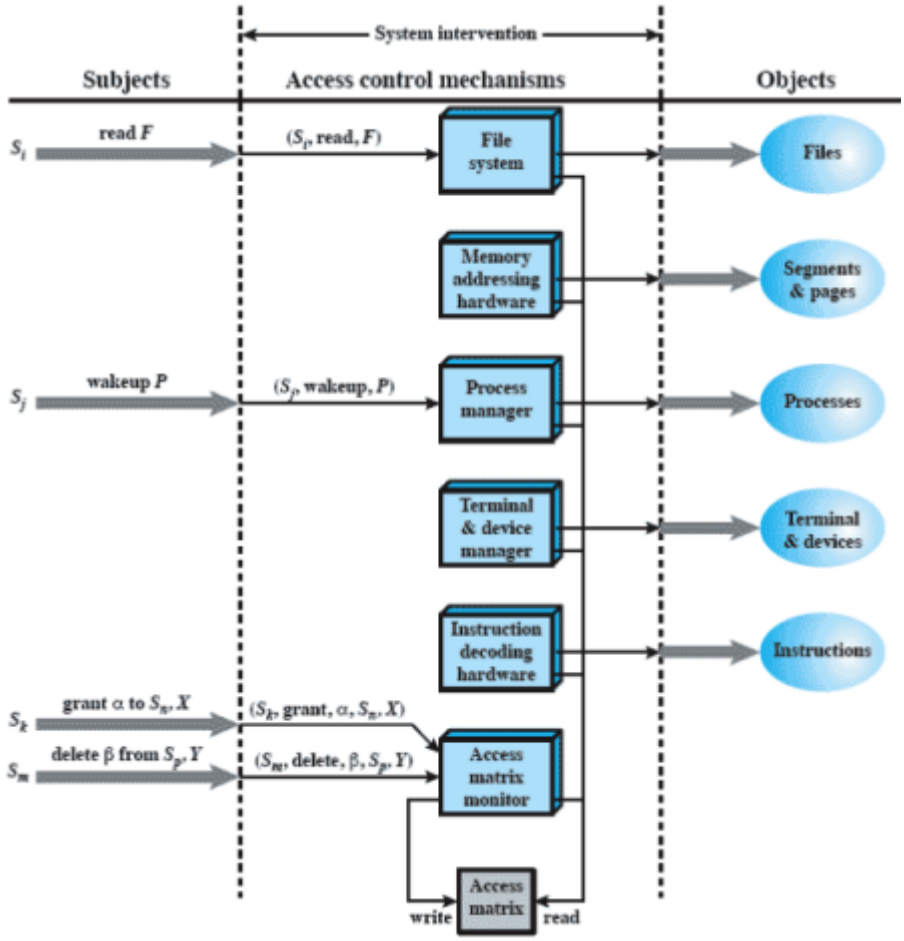
Discrezionale

		OBJECTS								
		subjects			files		processes		disk drives	
		S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
SUBJECTS	S_1	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	S_2		control		write *	execute			owner	seek *
	S_3			control		write	stop			

* - copy flag set

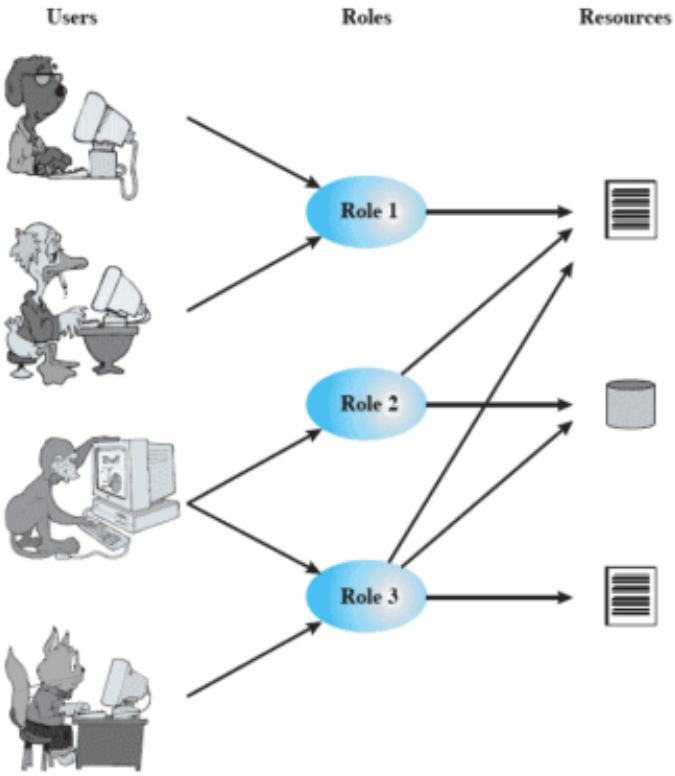
I soggetti riguardano degli utenti o dei processi. Ovviamente ogni utente/processo controlla sé stesso ma solo un utente è proprietario dell'altro. Sui file i soggetti possono leggere scrivere ed eseguire. Sui processi li possono svegliare e fermare

Dunque per ogni soggetto bisogna definire che azioni può eseguire sugli oggetti esistenti (in linux è ogni singolo file che ha le proprietà di chi lo può gestire)

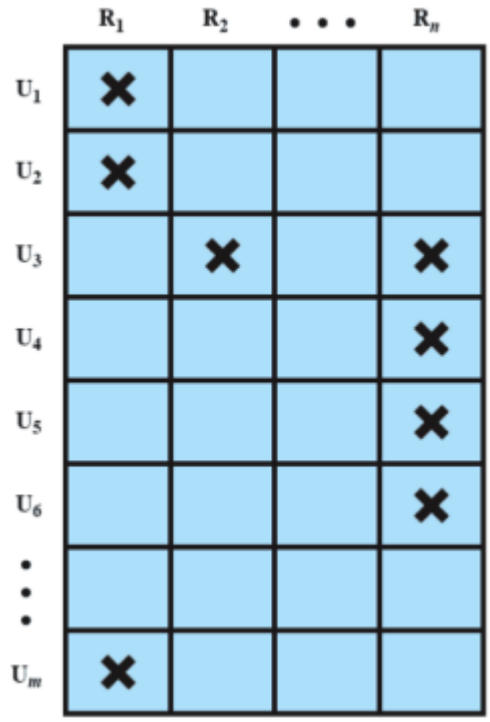


Basato sui ruoli

In questo tipo di controllo dell'accesso ci sta l'implementazione del cosiddetto **principio di minimo privilegio** secondo cui ci sono dei ruoli che definiscono il minimo insieme di diritti che devono avere gli utenti che appartengono a quei ruoli.
Dunque ad ogni utente, alla creazione, viene assegnato un ruolo che lo abilita ad effettuare le operazioni richieste per quel ruolo (ma solo mentre si sta agendo sotto quel ruolo)



Dunque in questo abbiamo bisogno di due tabelle per la gestione dei ruoli e dei permessi. Una per gestire a quali utenti sono assegnati quali ruoli e una per assegnare i permessi a ciascun ruolo



	OBJECTS								
ROLES	R_1	R_2	R_n	F_1	F_1	P_1	P_2	D_1	D_2
R_1	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
R_2		control		write *	execute			owner	seek *
. . .									
R_n			control		write	stop			

Unix: meccanismi di protezione

Tipicamente in Unix la sicurezza è basata sull'**autenticazione dell'utente** (*User-Oriented Access Control*) e il modello di controllo degli accessi si concentra sui dati stessi come punto centrale per decidere chi può fare cosa (*Data-Oriented Access Control*)

Nonostante ciò ci potrebbero essere altri meccanismi:

- NIS
- NDAP
- Kerberos

Utenze e gruppi

In Unix dunque per ogni utente ci sta uno *`username`* (alfanumerico) e un *`uid`* (numero intero).
Lo uid è usato ogni volta che occorre dare un proprietario ad una risorsa (file, processi, …)
Inoltre ogni utente appartiene ad un **gruppo** (analogamente identificato da *`groupname`* e *`gip`*)

Esistono inoltre dei file di sistema che permettono di associare i nomi con i corrispettivi codici numerici che sono `/etc/group` e `/etc/passwd` (talvolta in combinazione con `/etc/shadow`)

Una tipica entry del file `/etc/passwd` è formata così
`sabinar:x:6335:283:Sabina Rossi:/home/sabinar:/bin/csh`
In cui `sabrinar` indica l'username `6335` indica lo uid e `283` il gip, `x` password (oscurata), `/home/sabrinar` current working directory, `/bin/csh` shell da eseguire

Invece una tipica entry del file `/etc/group` è formata così
`aan:x:283`
`aan` groupname e `283` il gip

Login

Il login può essere fatto su un terminale della macchina (processo `getty`) o tramite rete (`telnet`, `ssh`). Questi processi richiedono una coppia `username+password`. Se corrisponde ad une entry di `/etc/passwd`, viene eseguita la shell indicata, a partire dalla directory di home indicata
Quando la shell esegue `exit`, o si ritorna al `getty` o si chiude la connessione di rete.
All'interno di una shell si può cambiare identità con il comando `su`

Accesso ai file

Per ogni file ci sono tre terne di permessi: lettura, scrittura, esecuzione.
La prima terna è il proprietario del file, la seconda per il gruppo cui il proprietario del file appartiene, la terza per tutti gli altri utenti.

Le terne di diritti sono usate ogni volta che un processo richiede l'accesso ad un file.
Se il proprietario del file e del processo coincidono, si guarda la prima terna, altrimenti la seconda terna se almeno appartengono allo stesso gruppo, altrimenti la terza terna.
Si prende poi l'elemento della terna corrispondente all'accesso richiesto

Il proprietario è lo stesso del processo che ha creato il file, ma si può cambiare con `chown`, mentre si possono cambiare diritti del file con `chmod`

```
-rwxr-xr-x 1 federico em 5120 Nov 7 11:03 a.out
-rw-r--r-- 1 federico em 233 Nov 7 11:03 test.c
```

SETUID e STGID

Ci sono dei casi in cui un utente normale deve essere messo nelle condizioni di poter accedere a dei file di sistema casomai solo in alcune situazioni particolari
Per questo motivo comandi come `passwrd` hanno il permesso speciale `SETUID` e/o `SETGID`. Tale permesso può essere accordato solo da un utente amministratore con `chmod u+s nomefile` e/o `chmod g+s nomefile`

In questo modo dunque l'uid o il gid del processo non sono quelli dell'utente che lo ha lanciato, ma **del proprietario del file eseguibile**

```
-rwxr-xr-x 1 federico em 5120 Nov 7 11:03 a.out
-rw-r--r-- 1 root root 1715 Oct 12 2014 /etc/passwd
-r-sr-sr-x 1 root sys 21964 Apr 7 2002 /bin/passwd

ci sta s al posto si x
```

⚠ **Meccanismo da usare con estrema cautela, facile fare attacchi `rootkit`**