Sicurezza dei Sistemi e delle Reti

Indice

1 Introduzione **3**
- 1.1 Terminologia . . . 3
- 1.2 Sicurezza . . . 4

2 Standard e Concetti base **5**
- 2.1 Standard . . . 5
- 2.2 Reference Monitor Model - RMM . . . 5
- 2.3 Definizioni . . . 6
- 2.4 Sfide della sicurezza informatica . . . 6
- 2.5 Principi fondamentali di progettazione della sicurezza . . . 7
- 2.6 Superificie di attacco . . . 7
  - 2.6.1 Categorie di attacco . . . 8
- 2.7 Albero di attacco . . . 8
- 2.8 Tipi di attacco . . . 8
- 2.9 Implementazione della sicurezza . . . 8

3 Controllo degli accessi **9**
- 3.1 Requisiti di sicurezza . . . 10
  - 3.1.1 Requisiti di sicurezza di base . . . 10
  - 3.1.2 Requisiti di sicurezza derivati . . . 10
- 3.2 Elementi di Access Control . . . 10
- 3.3 Politiche di controllo degli accessi . . . 11
  - 3.3.1 DAC . . . 11
  - 3.3.2 MAC . . . 12
  - 3.3.3 RBAC . . . 13
- 3.4 Unix security model . . . 14
  - 3.4.1 Processi in Linux . . . 14
  - 3.4.2 Unix file access control . . . 14
- 3.5 Windows security architecture . . . 15
  - 3.5.1 Windows security model . . . 15
  - 3.5.2 Security descriptor . . . 15
- 3.6 Set-UID . . . 16
  - 3.6.1 Superificie di attacco dei programmi Set-UID . . . 16

4 Politiche di sicurezza **18**
4.1 Definizioni 19
4.2 Malware 19
4.2.1 Classificazione 19

Capitolo 1

Introduzione

1.1 Terminologia

Un **sistema** può essere visto come (anche una combinazione di):

- *hw*
- *sw*
- persone che lavorano con *hw* e *sw*
- clienti

Un **attore** può essere:

- una *persona* che *interagisce* con il sistema
- un *dispositivo* che *interagisce* con il sistema
- un *ruolo* (cliente)
- un *ruolo complesso* (Alice che finge di essere Bob)

Una rete è una configurazione di individui interconnessi. Una **rete di computer** può essere vista sotto due punti di vista:

- ***Fisico:*** una infrastruttura *hw* che connette diversi dispositivi
- ***Logico:*** un sistema che facilita lo scambio di informazioni tra applicazioni che non condividono uno spazio di memoria

1.2 Sicurezza

La sicurezza può essere intesa come il **raggiungimento di un obiettivo in presenza di un attacco**; è difficile da assicurare perché l'obiettivo è *negativo*:

- *dimostrare che Alice può accedere ad un file è facile*
- *dimostrare che nessuno oltre ad Alice può accedervi è molto più difficile*

Di norma si raggiunge con un processo **iterativo**:

- si cerca di trovare l'*anello debole* nel sistema
- si adottano delle *contromisure*
- si continua a fare *analisi* in cerca di nuove vulnerabilità

Il concetto di *sicurezza perfetta* non è raggiungibile; per discutere di sicurezza si deve definire:

- **Politica di sicurezza:** definizione di regole di sicurezza che il sistema deve rispettare
- **Modello di minaccia:** assunzioni su cosa possa fare l'avversario per penetrare nel sistema; devo comprendere la potenza dell'avversario
- **Meccanismi:** *sw* o *hw* che cercano di assicuare che la politica sia rispettata, finché l'attaccante segue il modello di minaccia

Le reti di computer sono sistemi insicuri: abbiamo un sistema complesso (*computer*) in un sistema complesso (*rete*) $\rightarrow$ è difficile prevedere da quale punto arriveranno gli attacchi e quali vettori verranno sfruttati.
Ad oggi, le motivazioni dietro agli attacchi sono principalmente:

- economiche
- politiche / militari
- attivismo

Capitolo 2

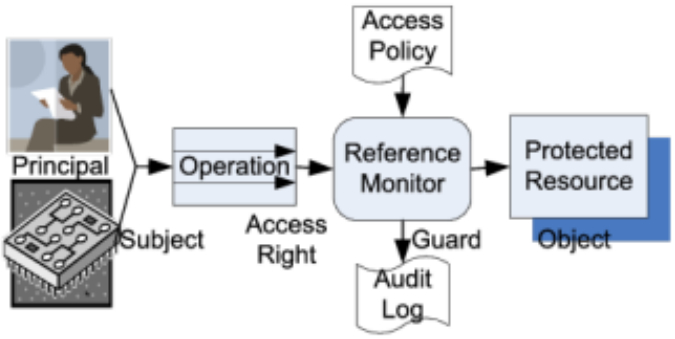
Standard e Concetti base

2.1 Standard

Ci sono diverse organizzazioni che si occupano di standard:

- NIST (*National Institute of Standars and Technology*)
- ISOC (*Internet Society*)
- ITU-T (*International Telecommunicatin Union*)
- ISO (*International Organization for Standardization*)
  - 27001: documento a cui fare riferimento per costruire un sistema di gestione della sicurezza delle informazioni che possa essere certificato da un ente indipendente
  - 27002: non è certificabile, è una raccolta di *best practices* per soddisfarre i requisiti della 27001

2.2 Reference Monitor Model - RMM



Il **reference monitor** è un **sistema dotato di una politica di controllo degli accessi**. Si occupa di:

- **autenticare chi vuole accedere**
- **autorizzare** o meno le operazioni richieste in base ai permessi
- fare **audit** → tenere un log delle azioni compiute

2.3 Definizioni

- La **sicurezza informatica** è l'insieme di strumenti, politiche, linee guida ... che possono essere utilizzate per proteggere l'ambiente e le risorse dell'organizzazione e degli utenti del cyberspazio
- I **beni dell'organizzazione e degli utenti** comprendono i dispositivi informatici connessi, personale, infrastrutture e la totalità delle informazioni trasmesse e/o archiviate nel cyberspazio
- Gli **obiettivi** generali di sicurezza comprendono *disponibilità, integrità e confidenzialità*
- *Sottoinsiemi* della sicurezza informatica:
  - **Sicurezza delle informazioni:** conservazione della CIA delle informazioni
  - **Sicurezza delle reti:** protezione delle reti e del loro servizio da modifiche non autorizzate e garanzia che la rete svolga correttamente le sue funzioni critiche

2.4 Sfide della sicurezza informatica

- Non è semplice; può avere requisiti semplici ma **meccanismi di implementazione complessi**
- Nello sviluppo di un meccanismo di sicurezza, si deve sempre **considerare potenziali attacchi**
- Le procedure utilizzate per fornire particolari servizi possono essere **controintuitive poiché complesse**
- Bisogna decidere **dove utilizzare i meccanismi di sicurezza**, sia a livello logico che a livello fisico
- I meccanismi di sicurezza in genere coinvolgono **più di un algoritmo o protocollo**
- Una battaglia **continua** tra attaccante e difensore

2.5 Principi fondamentali di progettazione della sicurezza

- **Fail-safe default:** nel caso in cui il sistema vada in default, deve rimanere in uno *stato protetto*
- **Economia di meccanismo:** i meccanismi devono essere il più semplice possibile
- **Mediazione completa:** *tutti* gli accessi devono essere controllati per assicurarsi che siano consentiti; solitamente accade che solo la prima interazione è controllata
- **Design aperto:** la sicurezza non deve dipendere dalla segretezza della sua progettazione o implementazione
- **Seperazione dei privilegi:** un sistema non dovrebbe concedere l'autorizzazione in base a *una singola* condizione
- **Minimi privilegi:** devono essere concessi il minor numero possibile di privilegi ad ogni soggetto; eventuali permessi addizionali devono essere concesso per il tempo minimo possibile
- **Accettabilità psicologica:** i meccanismi di sicurezza non dovrebbero rendere l'accesso ad una risorsa più difficile
- **Isolamento**
- **Incapsulamento**
- **Modularità**
- **Stratificazione (*layering*)**
- **Minima sorpresa:** evitare che l'utente si trovi davanti a situazioni inaspettate che potrebbero portarlo a seguire comportamenti scorretti

2.6 Superificie di attacco

Una superficie di attacco è costituita dalle **vulnerabilità raggiungibili e sfruttabili** in un sistema, come ad esempio:

- porte aperte verso l'esterno
- interfacce web
- dipendente con accesso a dati sensibili
- ...

→ è necessario **ridurre al minimo** la superificie di attacco

2.6.1 Categorie di attacco

- Superificie di attacco di **rete**: sono incluse vulnerabilità del protocollo di rete, che possono portare a DoS, interruzione dei collegamenti di comunicazioni ed altri attacchi intrusivi
- Superficie di attacco **software**: vulnerabilità nel codice delle applicazioni; un focus particolare è il software per server web
- Superificie di attacco **umano**: vulnerabilità create dal personale o da estranei, come *social engeneering*, errore umano o intrusi

2.7 Albero di attacco

Un albero di attacco è un modo ti **rappresentare le possibilità di attacco**, e quindi di progettare le **contromisure**.

2.8 Tipi di attacco

- **Passivi:** non alterano le informazioni in transito; lo scopo è ottenere informazioni sui messaggi trasmessi
- **Attivi:** modificano il flusso delle informazioni
  - *Attacco di replay:* l'attaccante osserva le informazioni e le riutilizza in un secondo momento per creare una nuova sessione di comunicazione
    $\rightarrow$ ci si tutela con *numeri casuali* e *timestamp* per, rispettivamente, controllare che i messaggi non siano già stati scambiati o che siano ancora validi
  - *DoS e DDoS*
  - ...

2.9 Implementazione della sicurezza

Quattro linee d'azione complementari:

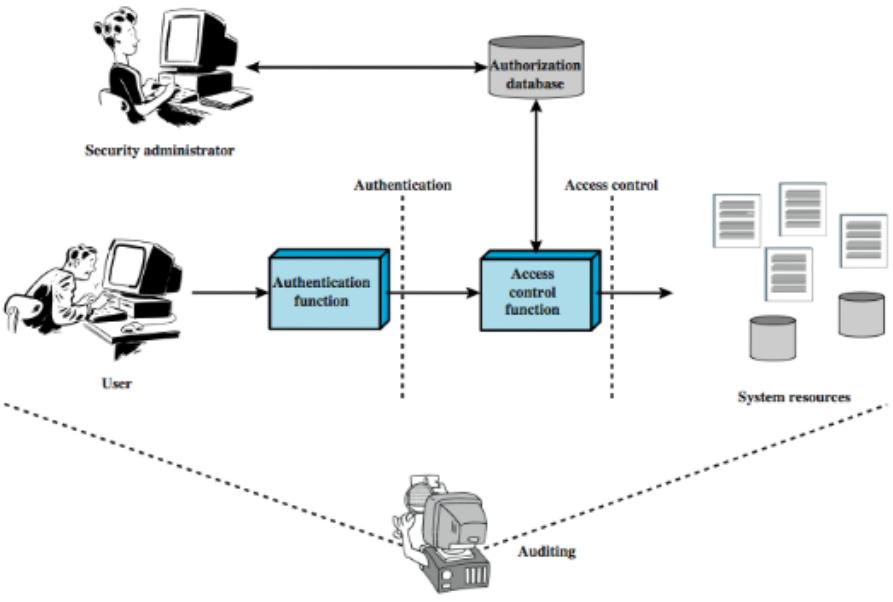
- **Prevenzione**
- **Rilevamento**
- **Risposta** in modo da fermare un attacco e prevenire ulteriori danni
- **Ripristino** con sistemi di backup in caso l'integrità dei dati sia compromessa

Capitolo 3

Controllo degli accessi

Il controllo degli accessi è un elemento centrale nella sicurezza informatica. È il **meccanismo** che definisce una **politica di sicurezza** per la quale si decide quali utenti possano accedere o meno ad una risorsa.
Si basa su tre principi fondamentali:

- **Autenticazione:** verifica che le credenziali fornite siano valide
- **Autorizzazione:** concessione di un permesso ad un'entità affinché possa accedere ad una risorsa del sistema
- **Auditing:** verifica delle attività e dei registri di sistema



3.1 Requisiti di sicurezza

3.1.1 Requisiti di sicurezza di base

- Limitare l'accesso al sistema informativo agli **utenti aurotizzati**, ai processi che agiscono per conto degli utenti autorizzati o ai dispositivi
- Limitare l'accesso al sistema informativo alle tipologie di funzioni che gli utenti autorizzati possono eseguire

3.1.2 Requisiti di sicurezza derivati

- **separare i doveri** dei singoli individui per ridurre il rischio di attività malevole
- utilizzare **account non privilegiati** quando si accede a funzioni non di sicurezza
- **impedire agli utenti non privilegiati** di eseguire funzioni privilegiate e controllare l'esecuzione di tali funzioni
- limitare i **tentativi di accesso** non riusciti
- utilizzare il **blocco della sessione** per nascondere l'accesso ai dati dopo un periodo di inattività
- fornire **avvisi di privacy** e sicurezza secondo le norme vigenti
- **terminare automaticamente** la sessione dopo una determinata azione
- **monitorare e controllare** le sessioni di **accesso remoto**
- usare sistemi **crittografici** per garantire la riservatezza delle sessioni per accessi da remoto
- **instradare l'accesso remoto** tramite punti di controllo degli accessi
- **autorizzare** l'esecuzione remota di **comandi privilegiati**
- **autorizzare l'accesso wireless** prima di consentire tali connessioni

3.2 Elementi di Access Control

- **Soggetto:** entità che può accedere agli oggetti (ad esempio, un processo che rappresenta l'utente)
- **Oggetto:** risorsa ad accesso controllato (file, directory, ...)
- **Diritto di accesso:** modo in cui un soggetto accede ad un oggetto (lettura, scrittura, ...)

3.3 Politiche di controllo degli accessi

- **DAC:** controlla l'accesso in base all'identità del soggetto e alle autorizzazioni che indicano che è/non è consentito fare ai richiedenti
- **MAC:** controlla l'accesso in base al confronto tra delle specifiche etichette di sicurezza applicate agli oggetti e le autorizzazioni di sicurezza
- **RBAC:** controlla l'accesso in base al ruolo che l'utente ha nel sistema e alle regole che stabiliscono quali accessi sono consentiti a quali ruoli
- **ABAC:** controlla l'accesso in base agli attributi dell'utente

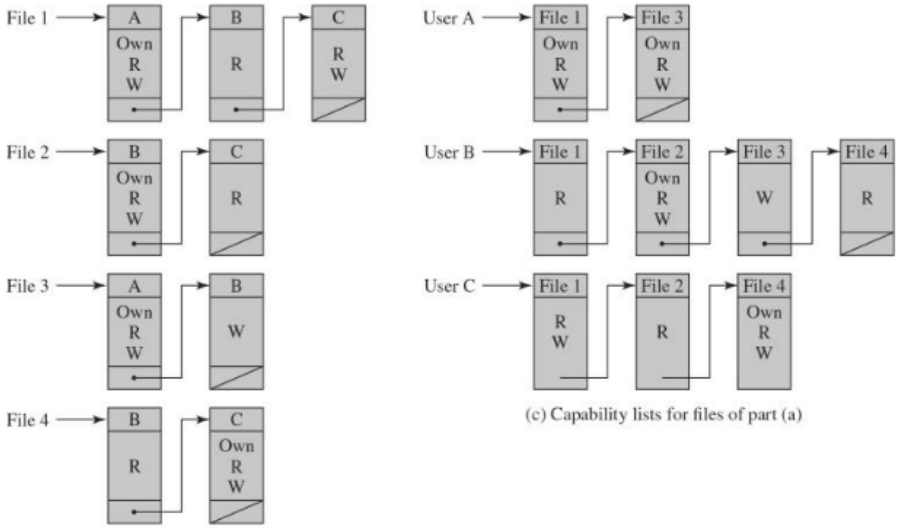
3.3.1 DAC

Il controllo dell'accesso viene fatto sull'**identità del soggetto richiedente** e delle **regole di accesso**. Definito *discrezionale* perché un'entità potrebbe avere i privilegi di accessi che le permettono, a sua volta, di concedere l'accesso ad un'altra entità.
Si può rappresentare mediante una matrice di accesso, dove:

- elenca i soggetti in una dimensione (colonne)
- elenca gli oggetti nell'altra dimensione (righe)
- ogni cella specifica i diritti di accesso di quel soggetto a quel determinato oggetto

	OBJECTS			
	File 1	File 2	File 3	File 4
User A	Own Read Write		Own Read Write	
User B	Read	Own Read Write	Write	Read
User C	Read Write	Read		Own Read Write

Questa matrice ha il problema di essere *sparsa*, ovvero molto grande e con molte celle vuote
→ viene trasformata in una serie di liste per risorse ed utenti



(b) Access control lists for files of part (a)

(c) Capability lists for files of part (a)

3.3.2 MAC

Controlla l'accesso in base al confronto tra:

- **Etichette di sicurezza** che indicano quanto sono sensibili le risorse
- **Autorizzazioni di sicurezza** che indicano quali entità del sistema sono idonee ad accedere a quali risorse

Questa politica è definita obbligatoria (*mandatory*) perché un'entità che ha accesso ad una risorsa non può estendere il permesso ad un'altra; può farlo solo l'amministratore di sistema.
I sistemi MAC si dividono in:

- ***Multilevel security systems:*** consiste in una struttura verticale di livelli di sicurezza; agli utenti viene assegnato un livello e possono accedere solo a risorse con livello uguale o inferiore
- ***Multilateral security systems:*** l'accesso viene assegnato in base a segmenti che formano gruppi costituiti da livelli di sicurezza e parole in codice

  → si ottiene una struttura orizzontale, che contiene livelli di sicurezza verticali aggiuntivi

Vantaggi e svantaggi

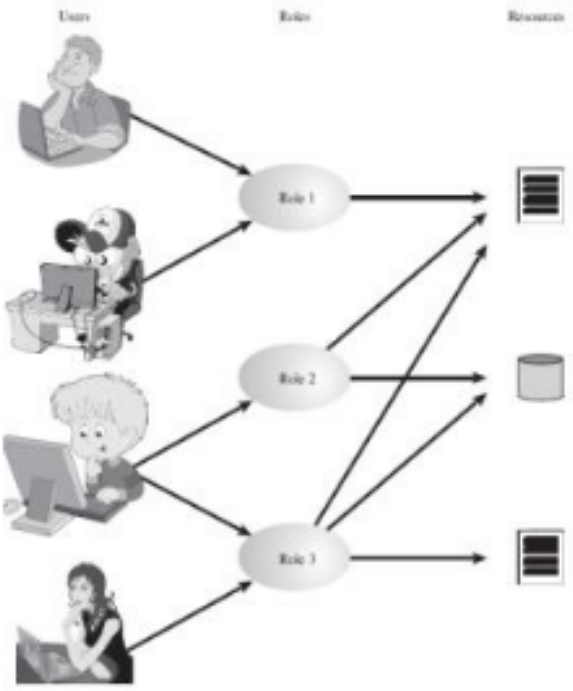
- Il MAC è uno dei sistemi di accesso più sicuri, poiché è praticamente a prova di manomissione

  - gli utenti non possono fare modifiche
  - il controllo è automatizzato
  - i dati non possono essere modificati senza apposita autorizzazione
- ...tuttavia
  - richiede una pianificazione dettagliata e lavoro amministrativo
  - controllare e aggiornare i diritti di accesso
  - manutenzione per aggiunta di nuovi dati o utenti e relative modifiche
  - → elevato carico di lavoro per l'amministratore

3.3.3 RBAC

Ci sono quattro tipi di entità:

- **Utente:** una persona che ha accesso al sistema; ogni individuo ha un ID associato
- **Ruolo:** inteso come una funziona lavorativa all'interno dell'organizzazione
- **Autorizzazione:** approvazione di una modalità di accesso ad uno o più oggetti
- **Sessione:** mappatura tra un utente e un sottoinsieme dei ruoli a cui è assegnato

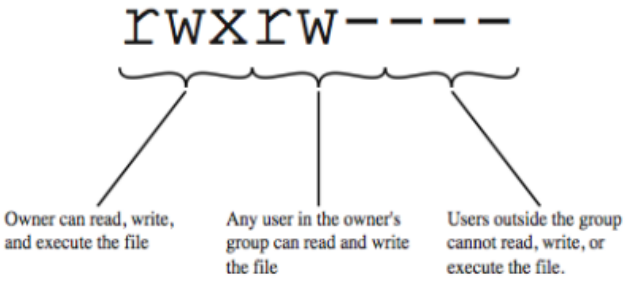


3.4 Unix security model

In Linux ci sono tre entità da considerare:

- **Soggetto:** può essere un utente o un processo
- **Oggetto:** file, cartelle, ...
- **Operazioni consentite:** lettura, scrittura, esecuzione

In Unix, ogni utente ha associato un id univoco, detto **UID**; può appartenere a gruppi di utenti, anch'essi identificati da un id univoco detto **GID**. Tutti gli utenti appartenenti ad un gruppo possono condividere tra loro oggetti.
Ad ogni file è assegnato un unico utente proprietario e un unico gruppo proprietario. L'autorizzazione viene concessa mediante una ACL che identifica le operazioni che i soggetti possono fare.



3.4.1 Processi in Linux

Ogni processo è isolato dagli altri e non possono accedere alla memoria altrui. Ogni processo viene eseguito con le autorizzazione dell'UID dell'utente che lo sta eseguendo.
Nel momento della creazione, ad ogni processo sono assegnati tre ID (inizialmente tutti uguali all'UID):

- **Effective UID:** determina le autorizzazioni per il processo
- **Real UID:** determina l'utente che ha avviato il processo
- **Saved UID:** EUID prima di eventuali modifiche

L'utente *root* può cambiare EUID/RUID/SUID a valori arbitrari; utenti non privilegiati possono cambiare EUID solo a RUID o SUID

3.4.2 Unix file access control

Le modifiche agli ID sono apportate mediante i comandi *setUID* e *setGID*; questa modifica permette ai programmi non privilegiati di accedere a risorse generalmente non accessibili.

Le directory possono aver impostato uno ***sticky bit***: specifica che solo il proprietario di un file nella cartella può apportare una modifica a quel file
Il ***superuser*** è esente dalle consuete restrizioni di controllo degli accessi, ha accesso a tutto il sistema.

3.5 Windows security architecture

L'architettura di sicurezza di Windows è basata su più entità:

- ***Security Reference Model (SRM):*** componente che in modalità kernel esegue controlli delle autorizzazioni e manipola i privilegi degli utenti
- ***Local Security Authority (LSA):*** risiede in un processo utente, è responsabile dell'applicazione della politica di sicurezza locale, tra cui:
  - criteri per le password, come complessità e tempi di scadenza
  - politica di controllo → specifica quali operazioni su quali oggetti vadano controllate
  - impostazioni dei privilegi
- ***Security Account Manager (SAM):*** è un database che archivia i dati degli account e le informazioni di sicurezza su entità locali e gruppi
- ***Active Directory (AD):*** implementa il protocollo LDAP (*Lightweight Directory Access Protocol*)

3.5.1 Windows security model

Windows ha un complesso sistema di controllo dell'accesso; ogni oggetto ha ACL per permettere autorizzazioni granulari ad utenti e/o gruppi di utenti.

3.5.2 Security descriptor

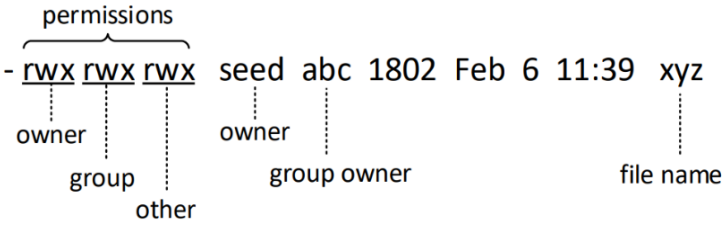
Ogni oggetto ha un *security descriptor* che contiene:

- ***security identifier* (SID)** per il possessore e il gruppo primario dell'oggetto (SID è associato univocamente ad ogni utente)
- ***discretionary ACL* (DACL):** diritti di accesso per gli utenti e i gruppi
- ***system ACL* (SACL):** tipi di accesso che generano log

Ad ogni processo viene inoltre associato un insieme di **token**, che prende il nome di ***security context***. Nel momento in cui un processo vuole accedere ad un oggetto, presenta il suo insieme di token e il sistema controlla se il security context abbia o meno accesso a tale risorsa in base al *security descriptor* dell'oggetto.

3.6 Set-UID

In Linux, ad ogni utente viene assegnato un ID univoco, memorizzato in `/etc/password`
Tramite i gruppi si rappresentano i gruppi di utenti che hanno le stesse autorizzazioni.
Le autorizzazioni su file vengono espresse nel seguente modo:



Per **set-UID** si intende **consentire all'utente di eseguire un programma con i privilegi del proprietario del programma**. Quando viene eseguito un normale programma, $RUID = EUID$, con entrambi che equivalgono all'id dell'utente che esegue il programma.
Quando viene eseguito un set-uid, RUID è uguale all'id dell'utente, ma EUID è uguale all'id del proprietario del programma

→ se il programma è di proprietà di *root*, viene eseguito con i *privilegi di root*

3.6.1 Superificie di attacco dei programmi Set-UID

- **Attacchi tramite input utente:** potrebbe essere inserito del codice malevolo, ad esempio tramite:
  - buffer overflow
  - inserimento di stringhe formattate appositamente
  - comando `chsh` → permette di cambiare sheòò tra quelli presenti in `/etc/password`

  → problema: è impossibile disinfettare gli input dell'utente
- **Attacchi tramite input di sistema:** cambiare i riferimenti simbolici a file privilegiati
- **Attacchi tramite variabili d'ambiente:** le variabili d'ambiente possono essere impostate prima dell'esecuzione di un programma e influenzarne il comportamento
- **Perdita di capacità:** i programmi privilegiati potrebbero declassarsi durante l'esecuzione

- **Invocazione di programmi:** invocazione di comandi esterni dall'interno di un programma
  - agli utenti viene chiesto di fornire dati in input al comando; se non vengono fatti gli opportuni controlli, i dati di input dell'utente potrebbero essere trasformati nel nome di un comando
- **Attacchi tramite dynamic linker:** se il programma prevede linking dinamico di librerie, anche il contenuto diventa rilevante; se manomesse, potrebbero portare a comportamenti anomali del codice

Capitolo 4

Politiche di sicurezza

La *gestione della sicurezza* è un *processo formale* per rispondere alle domande:

- quali sono i beni da proteggere
- quali sono le possibili minacce
- come si possono contrastare le minacce

Questo processo ha natura iterativa, ed è contenuto nella ISO 31000; in questa norma viene descritto un *modello per la gestione della sicurezza delle informazioni* che comprende le seguenti fasi:

- **Plan:**
  - stabilire politiche, processi e procedure di sicurezza
  - eseguire la valutazione del rischio
  - sviluppare un piano di trattamento del rischio
- **Do:**
  - implementare il piano di trattamento del rischio
- **Check:**
  - monitorare e mantenere il piano di trattamento del rischio
- **Act:**
  - mantenere e migliorare la gestione dei rischi
  - risposta ad incidenti, analisi di vulnerabilità e riprendere il ciclo iterativamente

4.1 Definizioni

- **Rischio:** esprime la possibilità che un attacco causi danni ad una organizzazione
- **Risorsa:** tutto ciò che necessita di essere protetto
  - *hw*
  - *sw*
  - *reputazione*

  La valutazione di una risorsa viene fatta in base:
  - ai costi da sostenere per sostituire la risorsa nel caso non sia più disponibile
  - perdita di incassi in caso di attacco
- **Vulnerabilità:** punti deboli che possono essere sfruttati per causare danni al sistema; possono essere classificate come:
  - critico
  - moderato
  - basso

4.2 Malware

Una definizione *informale* può essere quella di programma **malevolo**, solitamente inserito di nascosto in un sistema, che ha lo scopo di compremettere la riservatezza, l'integrità o la disponibilità del sistema stesso.

4.2.1 Classificazione

I malware sono classificati in base a:

- **Propagazione** (sw, rete, social engeneering, ...)
- **Azioni sui dati** colpiti (corruzione, furto, crittografia, ...)
- **Attack kit** (strumenti già pronti per attaccare)
- **Attori e/o motivazioni** dell'attacco