

# Protezione e Sicurezza

## Definizioni e livelli di sicurezza

- **Protezione:** insieme di attività volte a garantire il controllo dell'accesso alle risorse logiche e fisiche da parte degli utenti all'interno di un sistema di calcolo.
- **Livelli concettuali**
  - **Modelli:** definisce i soggetti, gli oggetti e i diritti di accessi
    - Soggetto è la parte attiva, cioè i processi che agiscono per conto di utenti, può essere considerato una coppia (*processo, dominio*), dove il dominio è l'ambiente di protezione nel quale sta eseguendo
    - Dominio di protezione: definisce un insieme di oggetti ed i tipi di operazioni che si possono eseguire su ciascun oggetto (*diritti di accesso*)
      - Es. di associazione dinamica processo-dominio quando chiamo una system call (utente chiede a SO, che la esegue a dominio SO)
      - Unix: dominio associato all'utente e il cambio di dominio corrisponde al cambio temporaneo di identità.
  - **Politiche:** regole con le quali i soggetti possono accedere agli oggetti
    - **Discretionary access control:** il creatore di un oggetto ne controlla i diritti di accesso (UNIX)
    - **Mandatory access control:** i diritti vengono gestiti centralmente
    - **Role Based access control:** ad un ruolo sono assegnati specifici diritti di accesso e gli utenti possono appartenere a diversi ruoli
  - **Meccanismi:** strumenti messi a disposizione dal sistema di protezione per imporre una politica
    - Separazione tra meccanismi e politiche
    - Flessibilità del sistema di protezione

## Modello a matrice di accesso

- Righe = utenti, colonne = risorse, dentro ci stanno i diritti (read, write ecc).
- Il meccanismo associato deve verificare se una richiesta è consentita o no, di modificare il numero di oggetti e soggetti, di cambiare dominio ad un processo, di cambiare la matrice
- Modifica dello stato di protezione (*Graham e Denning*):
  - **Propagazione diritti di accesso:** copy flag è un flag dentro una cella che indica se quel diritto può essere copiato dal suo utente ad un altro, in che modo
    - Propago solo il diritto o anche il copy flag?
    - Propago il diritto o lo trasferisco (cioè io lo perdo)?
  - **Assegnazione di un diritto di accesso:** posso assegnare solo se ho il diritto *owner* su quell'oggetto.
  - **Rimozione di un diritto di accesso:** posso togliere un diritto solo se ho il diritto *owner* o *control* sull'oggetto.
- **Problemi:** matrice molto grande e sparsa. Varianti
  - **Access Control List:** memorizzo per colonne, per ogni oggetto è associata una lista che contiene i soggetti che possono accedere all'oggetto e per questi i diritti (soggetti senza diritti non sono in lista). C'è anche una lista default con i diritti uguali per tutti gli oggetti (es. copy object).
    - Si possono includere anche gruppi, quando un utente accede deve specificare anche il gruppo (ruoli diversi per ogni utente a seconda del

- gruppo scelto, la entry ACL specifica una coppia utente-gruppo). L'utente può anche mantenere diritti indipendentemente dal gruppo (ACL con forma utente-\*)
  - Entry esaminate in sequenza (Uti senza diritti, poi \* per il resto)
  - Revoca semplice
  - Sapere i diritti di un utente è oneroso
- **Capability List:** memorizzato per righe, per ogni soggetto ho una lista con gli oggetti accessibili e i relativi diritti. Protetta da manomissioni degli utenti via HW (architettura etichettata) o SW, ossia CL gestita solo da SO. Revoca più onerosa
- **Soluzione mista:** alla richiesta di accesso, si cerca nella ACL il nome del soggetto e se ha il diritto si fornisce la *capability*, così posso accedere più volte senza ricercare nella ACL.
  - Es. UNIX accedo ad un file, copio della tabella dei file aperti associata al processo il descrittore (indice) e la capability, ritorno poi il file descriptor; le altre operazioni si fanno guardando questa capability.

## Sicurezza multilivello

- **Modello Ball – La Padula**
  - Livelli di sicurezza per documenti militari: non classificato, confidenziale, segreto, top secret
  - Persone assegnate ai livelli
  - *Proprietà di semplice sicurezza:* un processo in esecuzione al livello k può leggere solo al suo livello a livelli inferiori
  - *Proprietà \*:* un processo può scrivere al suo livello o superiori
  - Posso sovrascrivere dati di livello superiore!
- **Modello Biba**
  - *Proprietà di semplice sicurezza:* scrivo solo al mio livello e sotto
  - *Proprietà di integrità \*:* leggo solo al mio livello e sopra
  - In contrasto col modello precedente e non sovrapponibile
- **Reference monitor**
  - Elemento di controllo realizzato via HW e dal SO che regola l'accesso dei soggetti agli oggetti sulla base di parametri di sicurezza di soggetto e oggetto.
  - Ha accesso ad una base di dati fidata (*Trusted Computing Base*), che contiene
    - Privilegi di sicurezza di ogni soggetto
    - Attributi di protezione di ciascun oggetto
  - *Mediazione completa:* le regole di sicurezza sono applicate ad ogni accesso e non solo, ad esempio, all'apertura del file. Per questo deve essere anche HW
  - *Isolamento:* monitor e base di dati protette da modifiche non autorizzate
  - *Verificabilità:* deve essere possibile dimostrare formalmente le regole di sicurezza e che sono verificate mediazione completa e isolamento.
  - *Audit file:* contiene gli eventi importanti per la sicurezza, come violazioni scoperte ecc...

## Difesa dai cavalli di troia

- Piero installa cavallo di troia e un file tasca posteriore sul sistema e dà i diritti di scrittura su questo file a Paolo.
- Piero induce Paolo ad attivare il cavallo di troia

- Il programma, eseguito da Paolo, copia il suo file privato nel file tasca posteriore (permessi da ACL!)
- **Sistema operativo sicuro:** vengono fissati due livelli di sicurezza, riservato e pubblico. Ai processi ed al file dati di Paolo viene dato riservato, a quelli di Piero pubblico. Il reference monitor vieta di scrivere dati privati su un file pubblico. Questa politica ha la precedenza sulle ACL.

## Classificazioni di sicurezza

- **D:** nessun livello di sicurezza
- **C1:** autorizzazioni utenti, protezioni programmi e dati propri di ogni utente, controllo degli accessi a oggetti comuni per gruppi di utenti
- **C2:** C1 + controllo accessi anche per singoli utenti
- **B1:** C2 + introduzione livelli sicurezza del modello Bell – La Padula (almeno due livelli)
- **B2:** B1 + uso di etichette di riservatezza per ogni risorsa, compresi i canali
- **B3:** B2 + creazione di liste di controllo accessi in cui sono identificati utenti o gruppi cui non è consentito l'accesso ad un oggetto specificato
- **A1:** B3 + progetto e realizzato utilizzando metodi formali di definizione e verifica.