UNIVERSITÀ
DEGLI STUDI
DI MILANO

Sicurezza delle basi di dati

Politiche e controllo dell’accesso

LA STATALE

Prof. Stefano Montanelli

Obiettivi

- **Segretezza**: protezione delle informazioni da letture non autorizzate
- **Integrità**: protezione dei dati da modifiche o cancellazioni non autorizzate
- **Disponibilità**: garanzia che non si verifichino casi in cui ad utenti legittimi venga negato l'accesso ai dati

Tecniche

- **Autenticazione**: meccanismi per verificare l'identità dell'utente che si connette al sistema
- **Controllo dell'accesso**: meccanismi che, per ogni richiesta di accesso ai dati, verificano che l'utente sia autorizzato a compiere l'accesso
- **Crittografia**: meccanismi che consentono di cifrare i dati in modo che possano essere decifrati solo da utenti autorizzati

Noi ci focalizzeremo sulle tecniche di controllo dell'accesso perchè sono di pertinenza del DBMS

Controllo dell’accesso

- Regola le operazioni che si possono compiere sulle informazioni e le risorse in una base di dati
- Lo scopo è limitare e controllare le operazioni che gli utenti effettuano, prevenendo azioni accidentali o deliberate che potrebbero compromettere l’integrità e la segretezza dei dati
- Le risorse sono costituite dai dati, memorizzati in oggetti a cui si vuole garantire protezione
- I soggetti sono agenti (utenti o programmi in esecuzione) che richiedono di poter esercitare privilegi (come lettura, scrittura o esecuzione) sui dati

Controllo dell’accesso

- **Politiche di sicurezza**: norme e principi che esprimono le scelte di fondo dell’organizzazione relativamente alla sicurezza dei propri dati

- Sono implementate mediante traduzione in un insieme di regole di autorizzazione che stabiliscono le operazioni ed i diritti che gli utenti possono esercitare sui vari oggetti del sistema

- Il *Reference Monitor* è un meccanismo di controllo che ha il compito di stabilire se l’utente è autorizzato (totalmente o parzialmente) a compiere l’accesso

Politiche di sicurezza

- La politica di sicurezza adottata dipende principalmente da fattori organizzativi, quali l'ambiente di installazione, le esigenze degli utenti, i regolamenti dell'organizzazione, o i vincoli di natura legale. Due classi fondamentali:
  - Politiche per l'amministrazione della sicurezza
  - Politiche per il controllo dell'accesso ai dati

Politiche per l'amministrazione della sicurezza

- Stabiliscono chi concede e revoca i diritti di accesso
  - **Centralizzata**: un unico soggetto, detto DBA, controlla l'intera base di dati
  - **Decentralizzata**: più soggetti sono responsabili del controllo di porzioni diverse della base di dati
  - *Ownership*: l'utente che crea un oggetto (il proprietario) gestisce le autorizzazioni sull'oggetto

Politiche per il controllo dell'accesso

- Le politiche per il controllo dell'accesso stabiliscono se e come i soggetti possono accedere a quali dati contenuti nel sistema, e se e come possono venire trasmessi i diritti di accesso

  - **Need-To-Know** (minimo privilegio): molto restrittiva, permette ad ogni utente l'accesso solo ai dati strettamente necessari per eseguire le proprie attività

  - **Maximized Sharing** (massima condivisione): consente agli utenti il massimo accesso alle informazioni nella base di dati, mantenendo comunque informazioni riservate

Politiche per il controllo dell’accesso

- NEED TO KNOW offre ottime garanzie di sicurezza ed è adatta a basi di dati con elevate esigenze di protezione; può portare ad un sistema eccessivamente protetto, negando accessi che non comprometterebbero la sicurezza del sistema

- MAXIMIZED SHARING soddisfa il massimo numero possibile di richieste di accesso; viene utilizzata in ambienti in cui esiste una certa fiducia tra gli utenti ed in cui non è sentita una forte esigenza di protezione

Tipologie di sistema

- **Sistema aperto**. L'accesso è consentito a meno che non sia esplicitamente negato; le regole di autorizzazione indicano per ogni soggetto i diritti che egli non può esercitare sugli oggetti del sistema questi diritti sono i soli che gli saranno negati

- **Sistema chiuso**. L'acceso è permesso solo se esplicitamente autorizzato; le regole di autorizzazione indicano per ogni soggetto i diritti che egli può esercitare sugli oggetti del sistema questi diritti sono i soli che verranno accordati dal meccanismo di controllo

- Un sistema chiuso implementa la politica del minimo privilegio (need to know), un sistema aperto implementa la politica della massima condivisione (maximized sharing). Un sistema chiuso offre maggiori garanzie di sicurezza: una regola inavvertitamente cancellata o non inserita restringe ulteriormente l'accesso, mentre un sistema aperto permette accessi non autorizzati. La maggior parte delle basi di dati oggi esistenti sono sistemi chiusi

Granularità dei permessi

- La granularità dei permessi definisce la tipologia di oggetti a cui il controllo dell'accesso deve essere effettuato

- Requisito minimo: possibilità di specificare regole di autorizzazione sugli oggetti a cui l'utente può accedere → nelle BD relazionali la granularità è rappresentata dalla relazione o dagli attributi di una relazione

Tipologie di controllo

- **Controllo dipendente dal nome**: l'accesso è basato sul nome dell'oggetto
- **Controllo dipendente dal contenuto**: l'accesso è subordinato al valore di uno o più attributi dell'oggetto (es., l'utente X può accedere ai dati degli impiegati il cui stipendio non supera una certa soglia)
- **Controllo dipendente dal contesto**: l'accesso è subordinato al valore di variabili di sistema (es., data, tempo); es., i dati sugli impiegati possono essere acceduti solo in orario di lavoro
- **Controllo dipendente dalla storia degli accessi**: l'accesso è subordinato alla storia degli accessi eseguiti precedentemente (es., un utente può accedere ad un determinato dato solo se il numero di accessi da lui compiuti su quel dato in un determinato intervallo di tempo non supera una certa soglia)

Politiche discrezionali

- Richiedono che vengano specificati i diritti che ogni soggetto possiede sugli oggetti del sistema sottoforma di regole di autorizzazione. Il meccanismo di controllo esamina le richieste di accesso accordando solo quelle che sono autorizzate da una regola. Gli utenti possono a loro discrezione concedere o revocare i diritti di accesso sugli oggetti

- *Vantaggio*: sono estremamente flessibili e adatte a numerosi contesti applicativi

- *Svantaggio*: non impongono restrizioni sull'uso che viene fatto del dato una volta acceduto ovvero non forniscono alcun controllo sul flusso di informazioni nel sistema

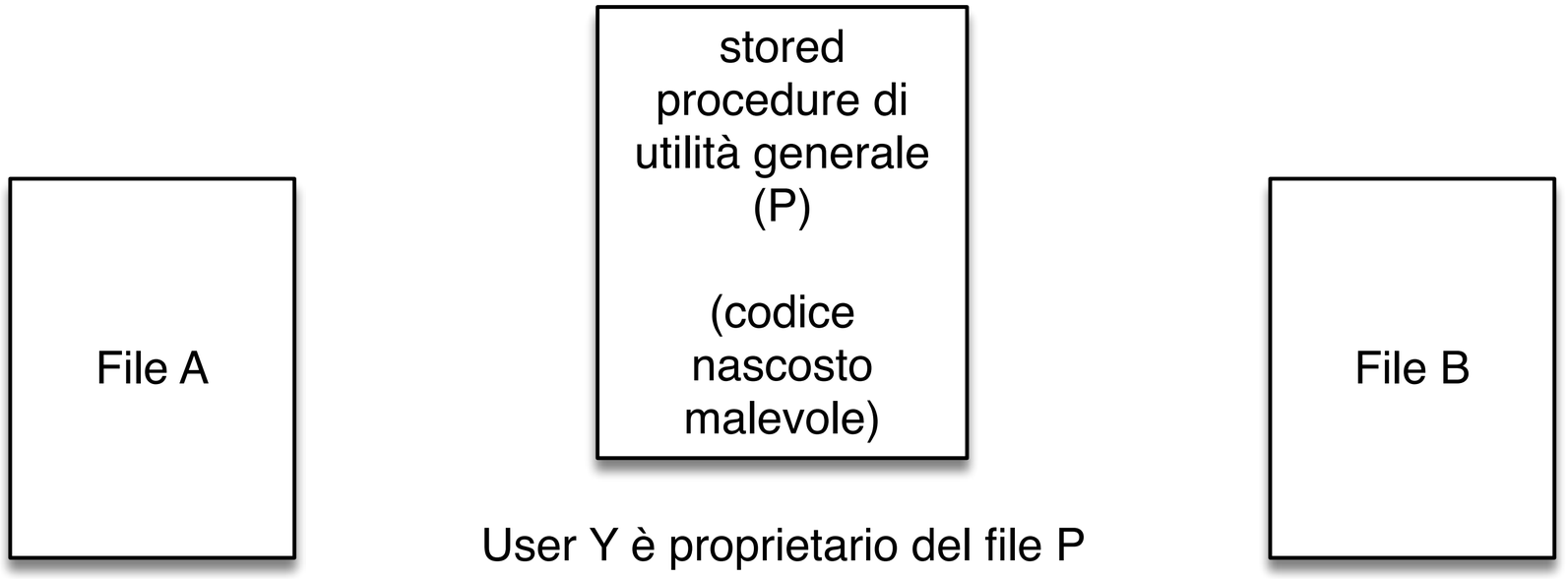
- Si ha un flusso tra un oggetto X e un oggetto Y quando si effettua una lettura del valore di X e una scrittura del valore in Y

Politiche mandatorie

- Per le basi di dati che richiedono elevati livelli di sicurezza (e.g., basi di dati governative) le politiche discrezionali non sono sufficienti
- Informazioni vitali, diversi livelli di sensitività, i controlli sul flusso di dati sono essenziali
- Attacchi sofisticati da parte di utenti (e.g., Cavallo di Troia)

Attacco del cavallo di troia

L'utente Y vuole leggere il contenuto del file A pur non avendone il permesso



- User X esegue la stored procedure P (lo può fare perché Y ha concesso il permesso)
- Il codice malevolo copia il contenuto del file A in B (lo può fare perché Y ha concesso la scrittura)
- **L'utente Y legge il contenuto di A copiato in B pur non avendone il permesso**

Politiche mandatorie

- Regolano l'accesso ai dati mediante la definizione di classi di sicurezza per i soggetti e gli oggetti del sistema
- Le classi di sicurezza sono ordinate parzialmente da una relazione d'ordine
- La classe di sicurezza assegnata ad un oggetto rappresenta il livello di sensitività dell'oggetto: maggiore è la classe assegnata ad un oggetto, più ingente sarà il danno derivante dal rilascio delle informazioni in esso contenute a soggetti non autorizzati
- La classe di sicurezza assegnata ad un soggetto è una misura del grado di fiducia che si ha nel fatto che tale soggetto non commetta violazioni

Controllo dell'accesso

- Il controllo dell'accesso è regolato da una serie di assiomi di sicurezza che stabiliscono le relazioni (in base al modo di accesso considerato) che devono essere verificate fra la classe di un soggetto e quella di un oggetto affinché al primo sia concesso di esercitare un modo di accesso sul secondo

- Queste politiche sono applicate in ambienti, ad esempio quello militare, dove la quantità di informazioni da proteggere è elevata, ci sono forti esigenze di protezione ed è possibile classificare rigidamente gli elementi del sistema

- I sistemi che adottano una politica mandatoria sono spesso indicati come sistemi multilivello

Politiche mandatorie

- Possono essere classificate anche come politiche per il controllo del flusso, poiché evitano che le informazioni una volta accedute vengano trasferite verso oggetti con classificazione inferiore e quindi più accessibili (vedere esempio Cavallo di Troia)
- C’è un controllo completo sul sistema di autorizzazione
- La flessibilità è però ridotta e la circolazione di informazioni tra gli utenti è più difficile
- Le politiche mandatorie e discrezionali non sono mutuamente esclusive, possono cioè essere applicate insieme
  - La politica mandatoria non controlla più le richieste di accesso ma le autorizzazioni che vengono assegnate ad un soggetto
  - Alla politica discrezionale è affidato il compito di controllare le richieste di accesso

Il System R

- Il modello implementa una politica di tipo discrezionale e supporta il controllo dell'accesso in base sia al nome che al contenuto
- Si tratta di un sistema chiuso: un accesso è concesso solo se esiste una esplicita regola che lo autorizza
- L'amministrazione dei privilegi è decentralizzata mediante ownership: quando un utente crea una relazione, riceve automaticamente tutti i diritti di accesso su di essa ed anche la possibilità di delegare ad altri tali privilegi

Il comando GRANT

Tramite il comando GRANT di SQL è possibile concedere privilegi ad altri utenti su una struttura della base di dati, ad esempio tabelle e attributi

```
GRANT Lista Privilegi | ALL [PRIVILEGES] ON Lista
Relazioni | Lista Viste TO Lista Utenti | PUBLIC [WITH
GRANT OPTION]
```

Il comando REVOKE

Tramite il comando REVOKE di SQL è possibile revocare permessi precedentemente concessi

```
REVOKE Lista Privilegi | ALL [PRIVILEGES] ON Lista
Relazioni | Lista Viste FROM Lista Utenti | PUBLIC
```

Un utente può revocare solo privilegi che lui stesso ha concesso

Quando si esegue una operazione di revoca, l'utente a cui i privilegi vengono revocati perde tali privilegi, a meno che essi non gli provengano anche da altre sorgenti indipendenti da quella che effettua la revoca

Delega dei privilegi

- La delega dei privilegi avviene mediante la grant option: se un privilegio è concesso con grant option l'utente che lo riceve può non solo esercitare il privilegio, ma anche concederlo ad altri
- Un utente può concedere un privilegio su una determinata relazione solo se è il proprietario della relazione, o se ha ricevuto tale privilegio da altri con grant option
- Se la clausola WITH GRANT OPTION non è specificata l'utente che riceve i privilegi non può concederli ad altri utenti
- Se ne deduce che i privilegi posseduti da un utente sono divisi in:
  - *delegabili*: privilegi concessi con grant option
  - *non delegabili*: concessi senza grant option

I cataloghi SYSAUTH e SYSCOLAUTH

- Le regole di autorizzazione specificate dagli utenti sono memorizzate in due cataloghi di sistema di nome **sysauth** e syscolauth, implementati come relazioni

id_utente	nome	creator	T	D	I	S	U	GO
bianchi	impiegato	bianchi	R	Y	Y	Y	all	Y
verdi	impiegato	bianchi	R	N	Y	Y	N	Y
rossi	impiegato	bianchi	R	N	N	Y	N	Y
rossi	impiegato	bianchi	R	N	Y	Y	N	N

T: type (Relation o View)
D: privilegio DELETE
I: privilegio INSERT
S: privilegio SELECT
U: privilegio di update sulle colonne
GO: l'utente possiede la Grant Option?

I cataloghi SYSAUTH e SYSCOLAUTH

- Le regole di autorizzazione specificate dagli utenti sono memorizzate in due cataloghi di sistema di nome sysauth e **syscolauth**, implementati come relazioni

id_utente	nome	colonna	GO
bianchi	impiegato	imp	Y
bianchi	impiegato	mansione	Y
...	...	...	...

Il catalogo SYSCOLAUTH è relativo al solo privilegio UPDATE

Uso del catalogo relazionale

- Quando un utente *u* esegue un comando di GRANT, il meccanismo di controllo accede ai cataloghi SYSAUTH e SYSCOLAUTH per determinare se *u* ha il diritto di delegare i privilegi specificati nel comando
- L'insieme dei privilegi delegabili che l'utente *u* possiede è intersecato con l'insieme dei privilegi specificati nel comando di GRANT
- Se l'intersezione è vuota, il comando non viene eseguito
- Se l'intersezione coincide con i privilegi specificati nel comando, vengono concessi tutti i privilegi specificati
- Altrimenti il comando viene eseguito parzialmente, cioè solo i privilegi contenuti dell'intersezione vengono accordati

Revoca ricorsiva

- L'operazione di revoca di un privilegio è ricorsiva: è revocato il privilegio oggetto del comando di revoca e tutti i privilegi che non avrebbero potuto essere concessi se l'utente specificato nel comando di revoca non avesse ricevuto il privilegio revocato

- Un'operazione di revoca del privilegio **m** sulla relazione **R** all'utente **u1** da parte dell'utente **u2** ha l'effetto di far perdere a **u1** il privilegio **m** sulla relazione **R** (se **u1** non ha ottenuto tale privilegio da altri utenti)

- Ha inoltre l'effetto di modificare il sistema portandolo in uno stato equivalente a quello in cui si sarebbe trovato se **u2** non avesse mai concesso a **u1** il privilegio di accesso m sulla relazione **R**

Revoca ricorsiva

- Per attuare la revoca ricorsiva è necessario determinare se un privilegio proviene da fonti indipendenti rispetto a quella specificata nel comando di revoke
- Sysauth e Syscolauth sono modificati per mantenere, per ogni privilegio, anche l'utente che ha concesso il privilegio, denominato grantor e il timestamp che denota il tempo in cui il privilegio è stato concesso
  - Il valore 0 indica che l'utente non ha quel privilegio
  - Un valore t <> 0 indica che privilegio è stato garantito all'utente al tempo t
  - Privilegi garantiti con lo stesso comando di GRANT hanno lo stesso timestamp

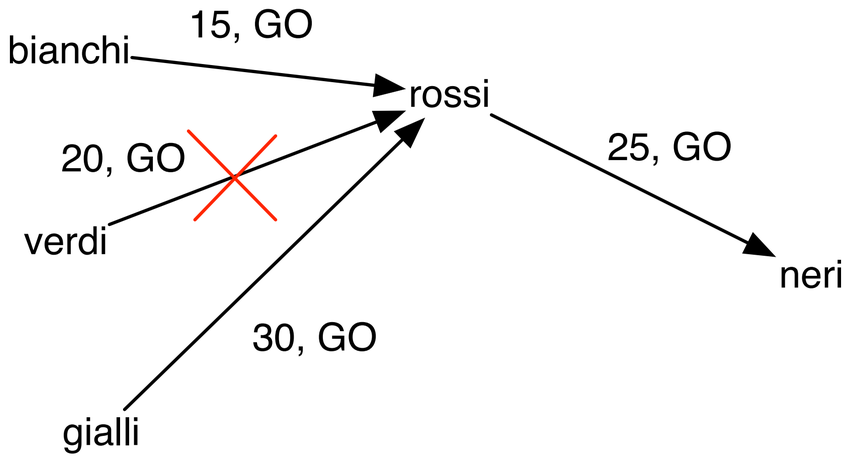
Revoca ricorsiva - esempio

id_utente	nome	grantor	T	I	S	D	GO
rossi	impiegato	bianchi	R	15	15	0	Y
rossi	impiegato	verdi	R	0	20	20	Y
neri	impiegato	rossi	R	25	25	25	Y
rossi	impiegato	gialli	R	0	30	30	Y

Al tempo 35, verdi: REVOKE ALL on impiegato from rossi;

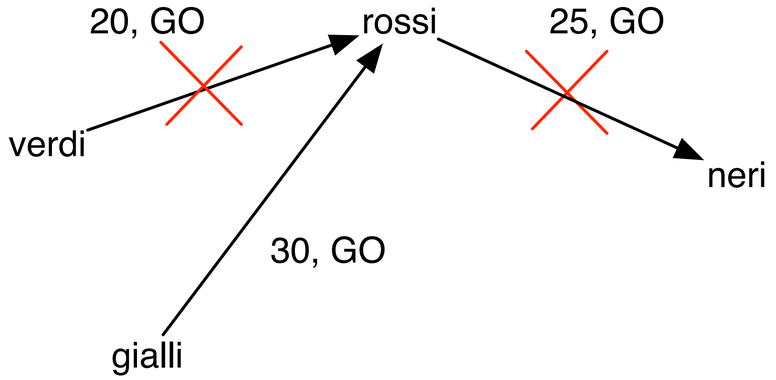
Revoca ricorsiva - esempio

SELECT - impiegato



Attenzione! Se rossi avesse ricevuto il privilegio da bianchi al tempo 15 senza GO, neri avrebbe perso il privilegio perchè rossi non avrebbe potuto concederlo al tempo 25 (lo riceverà a 30 da gialli)

DELETE - impiegato



Il privilegio delete su impiegato viene revocato in cascata a neri perché rossi non avrebbe potuto delegare il privilegio a neri al tempo 25

INSERT - impiegato