Dostopna varnost v SUPB

- Ena od pomembnih nalog SUPB je zagotoviti varnost dostopa do podatkovne baze.
- Večina današnjih SUPB omogoča eno ali obe od naslednjih možnosti:
  - Subjektivno določen nadzor dostopa (Discretionary access control)
  - Obvezen nadzor dostopa (Mandatory access control)

Nadzor dostopa

- Obvezen nadzor dostopa:
  - vsak objekt podatkovne baze ima določeno stopnjo zaupnosti (npr. zaupno, strogo zaupno,...),
  - vsak subjekt (uporabnik, program) potrebuje za delo z objektom določeno raven zaupanja (clearance level).
  - Za različne operacije (branje, pisanje, kreiranje,...) nad objekti podatkovne baze lahko subjekti potrebujejo različne nivoje zaupanja
  - Ravni zaupanja so strogo urejene
  - Značilno za varovana okolja, npr. vojska
  - Znana modela takega nadzora sta Biba in Bell-LaPadula (BLP)
    - BLP - v obliki končnega avtomata, pogoste nadgradnje

Obvezen nadzor dostopa

- Relativno redko implementiran
- Oracle: Virtual Private Database (BLP)
- IBM DB2: BLP
- Microsoft SQL Server: približek Row-Level Security (RLS), Dynamic Data Masking.
- PostgreSQL: SE-PostgreSQL; tudi RLS

Nadzor dostopa...

- Subjektivno določen nadzor dostopa:
  - Vsak uporabnik ima določene dostopne pravice (privilegije) nad dostopom do objektov podatkovne baze.
  - Tipično uporabnik pravice dobi v povezavi z lastništvom, ko kreira objekt.
  - Pravice lahko posreduje drugim uporabnikom na osnovi lastne presoje.
  - Tak način nadzora je lahko tvegan.

Nadzor dostopa in SQL...

- Vsak uporabnik podatkovne baze ima dodeljeno določeno pooblastilo - avtorizacijo (authorisation), ki mu ga dodeli skrbnik podatkovne baze (DBA).
- Pooblastilo je obenem tudi identifikator uporabnika.
- Navadno se za pooblastilo uporablja uporabniško ime ter geslo.
- SQL omogoča preverjanje pooblastila, s čimer identificira uporabnika.

Nadzor dostopa in SQL...

- Vsak SQL stavek, ki ga SUPB izvede, se izvede na zahtevo določenega uporabnika.

- Preden SUPB SQL stavek izvede, preveri dostopne pravice uporabnika nad objekti, na katere se SQL nanaša.

Nadzor dostopa in SQL...

- Vsak objekt, ki ga z SQL-om kreiramo, mora imeti lastnika.
- Vsak objekt se kreira v določeni shemi.
- Lastnika identificiramo na osnovi pooblastila, ki je določeno v shemi, kateri objekt pripada
  - Oracle: ime sheme je enako uporabniškemu imenu
  - MySQL in PostgreSQL: isti uporabnik je lahko lastnik več shem

Nadzor dostopa in SQL...

- Dostopne pravice ali privilegiji določajo, kakšne operacije so uporabniku dovoljene nad določenim objektom podatkovne baze.
- SQL standard pozna naslednje osnovne pravice:
  - SELECT – pravica branja podatkov
  - INSERT – pravica dodajanja podatkov
  - UPDATE – pravica spreminjanja podatkov (ne pa tudi brisanja)
  - DELETE – pravica brisanja podatkov
  - REFERENCES – pravica sklicevanja na stolpce določene tabela v omejitvah (npr. tuji ključi)
  - USAGE – pravica uporabe domen, sinonimov, znakovnih nizov in drugih posebnih objektov podatkovne baze
- Oracle – več ko 200 pravic v 25 skupinah:

  http://psoug.org/definition/GRANT.htm

Nadzor dostopa in SQL...

- Pravice v zvezi z dodajanjem (INSERT) in spreminjanjem (UPDATE) tabel ali pogledov so lahko določene na ravni stolpcev tabele/pogleda.

- Enako velja za pravice sklicevanja (REFERENCES)

Nadzor dostopa in SQL...

- Ko uporabnik kreira tabelo s CREATE TABLE avtomatsko postane lastnik tabele z vsemi pravicami.

- Ostalim uporabnikom dodeli pravice z ukazom GRANT.

Nadzor dostopa in SQL...

- Ko uporabnik kreira pogled s CREATE VIEW avtomatsko postane njegov lastnik, ne dobi pa nujno vseh pravic nad njim.
- Za kreiranje pogleda potrebuje SELECT pravice nad tabelami, iz katerih sestavlja pogled, ter REFERENCES pravice nad tabelami, katerih stolpce uporablja v definiciji omejitev.
- Ob kreiranju pogleda dobi pravice INSERT, UPDATE in DELETE, če te pravice ima nad vsemi tabelami, ki jih pogled zajema.

Nadzor dostopa in SQL...

- Uporaba ukaza GRANT

  GRANT {PrivilegeList | ALL PRIVILEGES}
  ON ObjectName
  TO {AuthorizationIdList | PUBLIC}
  [WITH GRANT OPTION]

- PrivilegeList – je sestavljen iz ene ali več pravic, ločenih z vejico (INSERT, UPDATE,...)
- ALL PRIVILEGES – dodeli vse pravice.

Nadzor dostopa in SQL...

- PUBLIC – omogoča podelitev pravic vsem trenutnim in bodočim uporabnikom.
- ObjectName – se nanaša na osnovno tabelo, pogled, domeno, znakovni niz, dodelitve in prevedbe.
- WITH GRANT OPTION – dovoljuje, da uporabnik naprej podeljuje pravice.

Nadzor dostopa in SQL...

- Vloga (ROLE): definiranje skupine pravic
- Nekatere vloge so definirane vnaprej (npr. dba)
- Uporabniško definirane vloge

```
-- Skupina pravic
CREATE ROLE Student;
GRANT priv1, priv2, … TO Student;
-- Podeljevanje skupine pravic uporabniku
GRANT Student TO PBB123456;
```

Primer dodeljevanja pravic...

- Uporabniku Janezu dodaj vse pravice nad tabelo rezervacija, tako da jih lahko podeljuje naprej.

```
GRANT ALL PRIVILEGES
ON rezervacija
TO Janez WITH GRANT OPTION;
```

Primer dodeljevanja pravic

- Uporabnikoma Petru in Pavlu dodeli SELECT in UPDATE pravice nad stolpcem cid v tabeli rezervacija.

```
GRANT SELECT, UPDATE (cid)
ON rezervacija
TO Peter, Pavel;
```

Nadzor dostopa in SQL...

- Z ukazom REVOKE pravice odvzamemo

```
REVOKE [GRANT OPTION FOR]
{PrivilegeList | ALL PRIVILEGES}
ON ObjectName
FROM {AuthorizationIdList | PUBLIC}
[RESTRICT | CASCADE]
```

Nadzor dostopa in SQL...

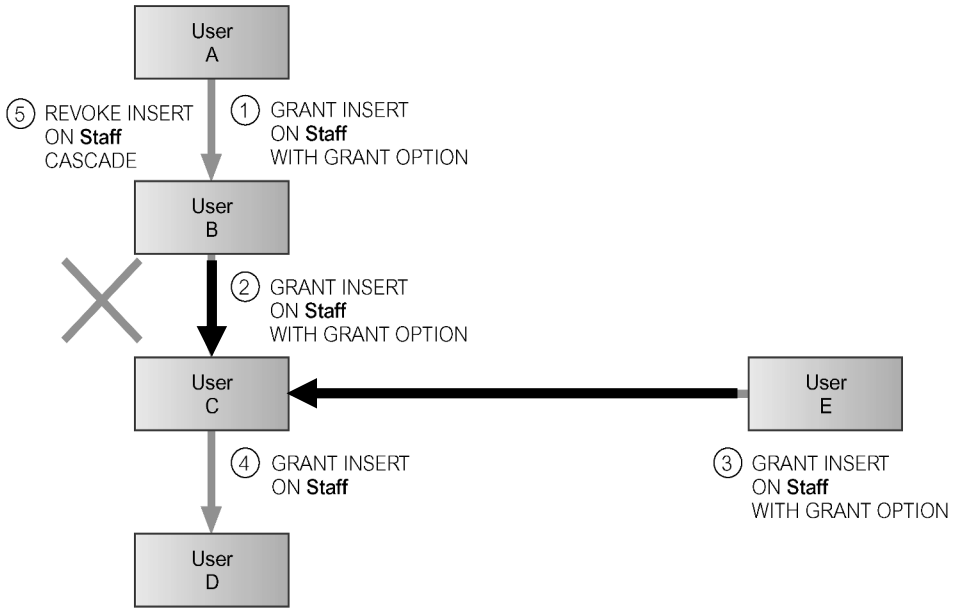
- ALL PRIVILEGES določa vse pravice, ki jih je uporabnik, ki REVOKE uporabi, podelil uporabniku ali uporabnikom, na katere se REVOKE nanaša.
- GRANT OPTION FOR – omogoča, da se pravice, ki so bile podeljene prek opcije WITH GRANT OPTION ukaza GRANT, odvzema posebej in ne kaskadno.
- RESTRICT, CASCADE – podobno kot pri ukazu DROP ali DELETE

Nadzor dostopa in SQL...

- Ukaz REVOKE ne uspe, kadar SUPB ugotovi, da bi njegova izvedba povzročila zapuščenost objektov:
  - Za kreiranje določenih objektov so potrebne pravice. Če te pravice odstranimo, lahko dobimo zapuščene objekte (odvisno od implementacije SUPB).
    - Uporabnik je imel pravico, da je ustvaril objekt v drugi shemi. Ko mu to pravico vzamemo, je ta objekt lahko zapuščen.
    - Uporabnik sebi ne more vzeti pravic nad lastnimi objekti (lastnina ≠ pravica)
  - Če uporabimo opcijo CASCADE, bo REVOKE ukaz uspel tudi v primeru, da privede do zapuščenih objektov. Kot posledica bodo ti ukinjeni.
- Poseben primer (odvisno od SUPB): ukaz REVOKE ne uspe, če ne uporabimo CASCADE, ko prekličemo pravico uporabnika, ki mu je bilo podeljena z GRANT OPTION.

Nadzor dostopa in SQL...

- Če uporabnik U_a odvzema pravice uporabniku U_b potem pravice, ki so bile uporabniku U_b dodeljene s strani drugih uporabnikov, ne bodo odvzete.



Primer odvzemanja pravic...

- Odvzemi DELETE pravice nad tabelo rezervacija vsem uporabnikom.

```
REVOKE DELETE
ON rezervacija
FROM PUBLIC;
```

Primer odvzemanja pravic

- Uporabniku Tinetu odvzemi vse pravice na tabelo rezervacija.

```
REVOKE ALL PRIVILEGES
ON rezervacija
FROM Tine;
```