SECURITATE SI AUDIT

Suport Laborator Saptămâna 10

FEAA Master SIA/SDBIS

Cuprins

Securitate și Audit

Domeniul securităţii e o treabă serioasă şi complexă. Există experţi în securitate naţională, în securitate alimentară, securitatea muncii şi câte şi mai câte. Evident, la acest laborator ne vom concentra doar asupra securităţii datelor şi mai precis a datelor din baza de date Oracle.

Utilizatori administrativi predefiniți

Mai ţineţi minte care este cel mai puternic utilizator în Oracle? L-am folosit la laboratoarele trecute. E vorba de utilizatorul SYS. Conectaţi ca SYS putem face ce vrem cu baza de date, inclusiv să o distrugem. Nerecomandat, evident! Orice bază de date Oracle vine cu acest utilizator predefinit. Fiind un utlizator cu drepturi de administrare depline, va trebui să-l securizaţi cum se cuvine. Pe medii de productie e recomandat să alegeti o parolă cu complexitate mare, eventual să interziceţi connectarea de la distanţă folosind acest utilizator.

Pe langă utilizatorul SYS, Oracle vine şi cu SYSTEM, un utilizator cu drepturi administrative, dar care nu este de tip SYSDBA, deci nu e la fel de puternic ca SYS. Chiar şi asa, utilizatorul SYSTEM poate să facă o mulţime de operaţii în baza de date pe care un utilizator normal, non-administrativ, nu le poate face. Fiind un utilizator administrativ e bine ca şi pentru SYSTEM să alegeţi o parola cu complexitate mare.

Haideţi să vedem cum putem schimba parola utilizatorului SYSTEM:

```
C:\Users\talek>sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Sun May 7 21:55:35 2017

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL> alter user SYSTEM identified by "ComplexPwd123!";

User altered.
```

Cum de a fost posibil sa ne conectăm fără a specifica parola? Vezi comanda: ***sqlplus / as sysdba***

Reţineţi modul în care puteţi schimba parola unui utilizator Oracle. O puteţi folosi pentru a schimba parola şi a unui utilizator non-admin.

Într-o baza de date Oracle pot exista mai multi utilizatori de tip SYSDBA. Pentru a vedea care sunt aceştia se poate interoga view-ul V$PWFILE_USERS:

```
SQL> desc v$pwfile_users
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USERNAME                                           VARCHAR2(30)
 SYSDBA                                             VARCHAR2(5)
 SYSOPER                                            VARCHAR2(5)
 SYSASM                                             VARCHAR2(5)
 SYSBACKUP                                          VARCHAR2(5)
 SYSDG                                              VARCHAR2(5)
 SYSKM                                              VARCHAR2(5)
 CON_ID                                             NUMBER

SQL> select * from v$pwfile_users;

USERNAME                       SYSDB SYSOP SYSAS SYSBA SYSDG SYSKM     CON_ID
------------------------------ ----- ----- ----- ----- ----- ----- ----------
SYS                            TRUE  TRUE  FALSE FALSE FALSE FALSE          0
SYSDG                          FALSE FALSE FALSE FALSE TRUE  FALSE          0
SYSBACKUP                      FALSE FALSE FALSE TRUE  FALSE FALSE          0
SYSKM                          FALSE FALSE FALSE FALSE FALSE TRUE           0
ADMIN                          TRUE  FALSE FALSE FALSE FALSE FALSE          0
```

Instalarea schemei de test

În acest laborator vom lucra cu o schemă simplificată a bine-cunoscutei aplicații de VANZARI, cea pe care am folosit-o și la laboratoarele precedente dedicate optimizării. Avem date despre CLIENTI, FACTURI, INCASARI și așa mai departe. Pentru a instala această schemă va trebui să descărcați de pe portal scriptul *vanzari_schema.sql* și să-l rulați într-o sesiune de Sqlplus, ca mai jos:

```
C:\Users\talek>sqlplus /nolog

SQL*Plus: Release 12.1.0.2.0 Production on Sun May 7 23:26:20 2017

Copyright (c) 1982, 2014, Oracle.  All rights reserved.

SQL> @C:\Users\talek\Dropbox\facultate\abd\lab_05\scripts\vanzari_schema.sql
Connecting as SYS...
Connected.

Cleanup VANZARI environment...

Create VANZARI schema.
The password is "Vanzari123"
Grant rights to VANZARI.

Connecting as VANZARI...
Connected.

Creating JUDETE table...
Creating CODURI_POSTALE table...
Creating CLIENTI table...
Creating PERSOANE table...
Creating PERSCLIENTI table...
Creating PRODUSE table...
Creating FACTURI table...
Creating LINIIFACT table...
Creating INCASARI_CLUSTER cluster...
Creating INCASARI table...
Creating INCASFACT table...

Populating tables...

Done.
```

Va trebui să ajustaţi comanda de rulare a scriptului "vanzari_schema.sql" cu calea unde aţi salvat, local, acest script.

Utilizatori/Conturi Oracle

Un utilizator Oracle poate fi privit din perspectiva persoanei sau a entităţii care se conectează la baza de date sau din perspectiva schemei care reuneşte diferite obiecte ale bazei de date. Haideţi să vedem cum creăm noi utilizatori Oracle.

```
CREATE USER FMIHAI IDENTIFIED BY FMIHAI;
CREATE USER PSILVIU IDENTIFIED BY PSILVIU;
CREATE USER GFLAVIU IDENTIFIED BY GFLAVIU;
CREATE USER TMARIN IDENTIFIED BY TMARIN;

GRANT CREATE SESSION TO FMIHAI, PSILVIU, GFLAVIU, TMARIN;
```

Care este semnificaţia privilegiului "CREATE SESSION"?
Ar putea TMARIN să creeze noi tabele în schema sa?

Reţineţi felul în care se specifică parola pentru utilizatorii nou creaţi. Daca parola conţine caractere speciale, va trebui sa o încadraţi între ghilimele duble.

Pentru a vedea toţi utilizatorii Oracle, inclusiv pe cei pe care tocmai i-am creat, putem da următoarea interogare:

```
SQL> desc dba_users
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 USERNAME                                  NOT NULL VARCHAR2(128)
 USER_ID                                   NOT NULL NUMBER
 PASSWORD                                           VARCHAR2(4000)
 ACCOUNT_STATUS                            NOT NULL VARCHAR2(32)
 LOCK_DATE                                          DATE
 EXPIRY_DATE                                        DATE
 DEFAULT_TABLESPACE                        NOT NULL VARCHAR2(30)
 TEMPORARY_TABLESPACE                      NOT NULL VARCHAR2(30)
 CREATED                                   NOT NULL DATE
 PROFILE                                   NOT NULL VARCHAR2(128)
 INITIAL_RSRC_CONSUMER_GROUP                        VARCHAR2(128)
 EXTERNAL_NAME                                      VARCHAR2(4000)
 PASSWORD_VERSIONS                                  VARCHAR2(12)
 EDITIONS_ENABLED                                   VARCHAR2(1)
```

```
 AUTHENTICATION_TYPE                              VARCHAR2(8)
 PROXY_ONLY_CONNECT                               VARCHAR2(1)
 COMMON                                           VARCHAR2(3)
 LAST_LOGIN                                       TIMESTAMP(9) WITH TIME ZONE
 ORACLE_MAINTAINED                                VARCHAR2(1)

SQL> column username format a25
SQL> column account_status format a20
SQL> select username, account_status, created from dba_users;

USERNAME                  ACCOUNT_STATUS       CREATED
------------------------- -------------------- ------------------
RMS                       OPEN                 12-APR-17
SOE                       OPEN                 25-MAR-17
FB                        OPEN                 12-MAR-17
SCOTT                     EXPIRED & LOCKED     11-MAR-17
GFLAVIU                   OPEN                 08-MAY-17
ORACLE_OCM                EXPIRED & LOCKED     11-SEP-14
OJVMSYS                   EXPIRED & LOCKED     11-SEP-14
SYSKM                     EXPIRED & LOCKED     11-SEP-14
XS$NULL                   EXPIRED & LOCKED     11-SEP-14
PSILVIU                   OPEN                 08-MAY-17
TMARIN                    OPEN                 08-MAY-17

USERNAME                  ACCOUNT_STATUS       CREATED
------------------------- -------------------- ------------------
GSMCATUSER                EXPIRED & LOCKED     11-SEP-14
MDDATA                    EXPIRED & LOCKED     11-SEP-14
SYSBACKUP                 EXPIRED & LOCKED     11-SEP-14
DIP                       EXPIRED & LOCKED     11-SEP-14
SYSDG                     EXPIRED & LOCKED     11-SEP-14
VANZARI                   OPEN                 07-MAY-17
APEX_PUBLIC_USER          OPEN                 11-SEP-14
RMS_TEST                  OPEN                 06-MAY-17
SPATIAL_CSW_ADMIN_USR     EXPIRED & LOCKED     11-SEP-14
SPATIAL_WFS_ADMIN_USR     EXPIRED & LOCKED     11-SEP-14
ADMIN                     OPEN                 02-MAY-17

USERNAME                  ACCOUNT_STATUS       CREATED
------------------------- -------------------- ------------------
GSMUSER                   EXPIRED & LOCKED     11-SEP-14
AUDSYS                    EXPIRED & LOCKED     11-SEP-14
FMIHAI                    OPEN                 08-MAY-17
FLOWS_FILES               EXPIRED & LOCKED     11-SEP-14
DVF                       EXPIRED & LOCKED     11-SEP-14
MDSYS                     EXPIRED & LOCKED     11-SEP-14
ORDSYS                    EXPIRED & LOCKED     11-SEP-14
DBSNMP                    EXPIRED & LOCKED     11-SEP-14
WMSYS                     EXPIRED & LOCKED     11-SEP-14
APEX_040200               EXPIRED & LOCKED     11-SEP-14
APPQOSSYS                 EXPIRED & LOCKED     11-SEP-14

USERNAME                  ACCOUNT_STATUS       CREATED
------------------------- -------------------- ------------------
GSMADMIN_INTERNAL         EXPIRED & LOCKED     11-SEP-14
ORDDATA                   EXPIRED & LOCKED     11-SEP-14
CTXSYS                    EXPIRED & LOCKED     11-SEP-14
ANONYMOUS                 EXPIRED & LOCKED     11-SEP-14
XDB                       EXPIRED & LOCKED     11-SEP-14
ORDPLUGINS                EXPIRED & LOCKED     11-SEP-14
DVSYS                     EXPIRED & LOCKED     11-SEP-14
SI_INFORMTN_SCHEMA        EXPIRED & LOCKED     11-SEP-14
OLAPSYS                   EXPIRED & LOCKED     11-SEP-14
LBACSYS                   EXPIRED & LOCKED     11-SEP-14
OUTLN                     EXPIRED & LOCKED     11-SEP-14

USERNAME                  ACCOUNT_STATUS       CREATED
------------------------- -------------------- ------------------
SYSTEM                    OPEN                 11-SEP-14
SYS                       OPEN                 11-SEP-14
```

	*Care este semnificaţia coloanei ACCOUNT_STATUS?*

Roluri Oracle

Un rol Oracle nu este altceva decat o colecţie de privilegii, colecţie căreia i se dă un nume. Putem apoi acorda acest rol utilizatorilor. Totuşi, crearea acestor roluri trebuie să ţină cont de specificul aplicaţiei şi de tipul de utilizatori care urmează să o acceseze.

Pe aplicaţia noastră de vânzări putem imagina următoarele scenarii de utilizare, precum şi drepturile Oracle necesare:

Scenariu de utilizare	**Operaţii în baza de date**
Adăugare clienţi	SELECT FROM JUDETE
	SELECT FROM CODURI_POSTALE
	INSERT INTO CLIENTI
Actualizare clienţi	SELECT FROM JUDETE
	SELECT FROM CODURI_POSTALE
	SELECT FROM CLIENTI
	UPDATE CLIENTI SET DENCL, ADRESA, CODPOST
Introducere facturi	SELECT FROM CLIENTI
	+ Adăugare clienţi
	SELECT FROM PRODUSE
	INSERT INTO FACTURI
	INSERT INTO LINIIFACT
Consultare facturi	SELECT FROM CLIENTI
	SELECT FROM FACTURI
	SELECT FROM LINIIFACT
	SELECT FROM PRODUSE
Corecţii facturi	**+ Consultare facturi**
	UPDATE FACTURI SET CODCL, DATAFACT
	UPDATE LINIIFACT SET CODPS, CANTITATE, PRETUNIT
	INSERT INTO LINIIFACT
Anulare facturi	**+ Consultare facturi**
	DELETE FROM FACTURI

Pornind de la scenariile de utilizare de mai sus, putem crea următoarelor roluri Oracle:

```
CONNECT / AS SYSDBA

CREATE ROLE INTRODUCERE_CLIENTI;
GRANT SELECT ON vanzari.JUDETE TO INTRODUCERE_CLIENTI;
GRANT SELECT ON vanzari.CODURI_POSTALE TO INTRODUCERE_CLIENTI;
GRANT INSERT ON vanzari.CLIENTI TO INTRODUCERE_CLIENTI;

CREATE ROLE ACTUALIZARE_CLIENTI;
GRANT SELECT ON vanzari.JUDETE TO ACTUALIZARE_CLIENTI;
GRANT SELECT ON vanzari.CODURI_POSTALE TO ACTUALIZARE_CLIENTI;
GRANT SELECT ON vanzari.CLIENTI TO ACTUALIZARE_CLIENTI;
GRANT UPDATE(dencl, adresa, codpost) ON vanzari.CLIENTI TO ACTUALIZARE_CLIENTI;

CREATE ROLE INTRODUCERE_FACTURI;
GRANT SELECT ON vanzari.CLIENTI TO INTRODUCERE_FACTURI;
GRANT INTRODUCERE_CLIENTI TO INTRODUCERE_FACTURI;
GRANT SELECT ON vanzari.PRODUSE TO INTRODUCERE_FACTURI;
GRANT INSERT ON vanzari.FACTURI TO INTRODUCERE_FACTURI;
GRANT INSERT ON vanzari.LINIIFACT TO INTRODUCERE_FACTURI;

CREATE ROLE CONSULTARE_FACTURI;
GRANT SELECT ON vanzari.CLIENTI TO CONSULTARE_FACTURI;
GRANT SELECT ON vanzari.FACTURI TO CONSULTARE_FACTURI;
GRANT SELECT ON vanzari.LINIIFACT TO CONSULTARE_FACTURI;
GRANT SELECT ON vanzari.PRODUSE TO CONSULTARE_FACTURI;

CREATE ROLE CORECTII_FACTURI ;
GRANT CONSULTARE_FACTURI TO CORECTII_FACTURI;
GRANT UPDATE(codcl, datafact) ON vanzari.FACTURI TO CORECTII_FACTURI;
GRANT UPDATE(codpr, cantitate, pretunit) ON vanzari.LINIIFACT TO CORECTII_FACTURI;
GRANT INSERT ON vanzari.LINIIFACT TO CORECTII_FACTURI;

CREATE ROLE ANULARE_FACTURI;
GRANT CONSULTARE_FACTURI TO ANULARE_FACTURI;
GRANT DELETE ON vanzari.FACTURI TO ANULARE_FACTURI;
```

Practic, pentru fiecare scenariu de utilizare a fost creat un rol oracle, rol ce conține toate privilegiile asociate respectivului scenariu.

Rețineți comanda GRANT cu care se poate da un privilegiu, fie unui rol, fie unui utilizator.

Pentru a vedea toate rolurile Oracle puteți interoga view-ul DBA_ROLES.

Care este comanda prin care se poate retrage un privilegiu Oracle acordat?

În ce schemă au fost create rolurile aplicației?

Ce tipuri de privilegii Oracle cunoașteți?

Cum putem vedea toate privilegiile acordate unui rol?

Crearea structurii de autorizare

Am creat deja câţiva utilizatori: FMIHAI, PSILVIU, GFLAVIU şi TMARIN. Mai mult, le-a fost acordat acestor utilizatori dreptul de a se conecta la baza de date (vezi privilegiul de CREATE SESSION).

Ipotetic, haideţi să zicem că:

- *FMIHAI* e şef serviciu contabilitate
- *PSILVIU* este arondat compartimentului desfacere
- *GFLAVIU* şi *TMARIN* sunt la contablitate

Sintetic, ajungem la următoarea structură a rolurilor organizaţionale:

Rol Organizaţional **Grupuri utilizatori**	**Scenarii de utilizare** **Grupuri drepturi**	**Utilizatori**
SEF_SERVICIU_CONTABLITATE	CONSULTARE_FACTURI	FMIHAI
	ANULARE_FACTURI	
SERVICIU_DESFACERE	INTRODUCERE_CLIENTI	PSILVIU
	ACTUALIZARE_CLIENTI	
	INTRODUCERE_FACTURI	
SERVICIU_CONTABILITATE	CORECTII_FACTURI	GFLAVIU
	CONTARE_CLIENTI	TMARIN

Ne amintim că pentru fiecare scenariu de utilizare deja am creat cate un rol Oracle, căruia i-am acordat drepturile corespunzătoare. Nu avem, totuşi, câte un rol Oracle prin care să modelăm rolurile organizaţionale. În Oracle putem acorda un rol altui rol, prin urmare lucrurile devin simple. Rulaţi instrucţiunile de mai jos conectaţi ca SYS:

```
CREATE ROLE SEF_SERVICIU_CONTABILITATE;
CREATE ROLE DESFACERE;
CREATE ROLE CONTABILITATE;

GRANT CONSULTARE_FACTURI, ANULARE_FACTURI TO SEF_SERVICIU_CONTABILITATE;
GRANT INTRODUCERE_CLIENTI, ACTUALIZARE_CLIENTI, INTRODUCERE_FACTURI TO DESFACERE;
GRANT CORECTII_FACTURI TO CONTABILITATE;
```

Bun, avem rolurile organizaţionale create, nu ne mai rămâne decât să le acordăm utilizatorilor, după cum urmează:

```
GRANT SEF_SERVICIU_CONTABILITATE TO FMIHAI;
GRANT DESFACERE TO PSILVIU, GFLAVIU;
GRANT CONTABILITATE TO TMARIN;
```

Activarea auditului standard

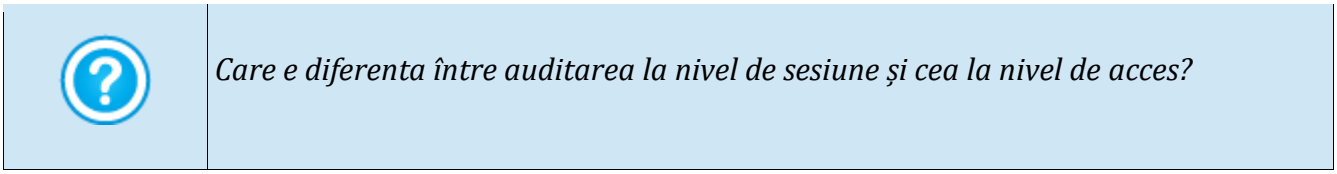
O parte importanta din componenta de securitate o reprezintă auditul. În următoarea parte a laboratorului ne propunem să auditǎm următoarele operaţiuni:

- orice UPDATE, DELETE, INSERT pe care-l face TMARIN, la nivel de acces
- orice incercare nereusita de DELETE pe tabela CLIENTI, indiferent de utilizator, la nivel de acces

- orice incercare de DELETE pe tabela FACTURI, indiferent de utilizator, la nivel de sesiune
- orice incercare nereusita de UPDATE pe tabela LINIIFACT, indiferent de utilizator, la nivel de sesiune

Pentru a implementa auditarea de mai sus, folosim comanda AUDIT:

```
AUDIT UPDATE TABLE, INSERT TABLE, DELETE TABLE BY tmarin BY ACCESS;
AUDIT DELETE ON vanzari.clienti BY ACCESS WHENEVER NOT SUCCESSFUL;
AUDIT DELETE ON vanzari.facturi BY SESSION;
AUDIT UPDATE ON vanzari.liniifact WHENEVER NOT SUCCESSFUL;
```



Care e diferenta între auditarea la nivel de sesiune şi cea la nivel de acces?

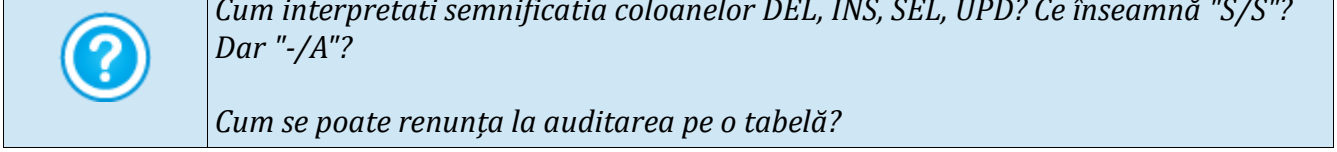
Pentru a vedea ce comenzi SQL sunt auditate pentru utilizatorii aplicaţiei noastre putem folosi interogarea de mai jos:

```
SYS@SQL> column user_name format a20
SYS@SQL> set linesize 120
SYS@SQL> select user_name, audit_option, success, failure from dba_stmt_audit_opts where user_name in ('FMIHAI',
'PSILVIU', 'TMARIN');

USER_NAME            AUDIT_OPTION                             SUCCESS    FAILURE
-------------------- ---------------------------------------- ---------- ----------
TMARIN               INSERT TABLE                             BY ACCESS  BY ACCESS
TMARIN               UPDATE TABLE                             BY ACCESS  BY ACCESS
TMARIN               DELETE TABLE                             BY ACCESS  BY ACCESS
```

Doar TMARIN apare, deoarece ceilalţi utilizatori nu au fost explicit auditaţi.

Pentru a vedea auditul la nivel de obiect în baza de date (in cazul nostru cele câteva tabele din schema VANZARI), folosim următoarea interogare:

```
SYS@SQL> column object_name format a20
SYS@SQL> set linesize 120
SYS@SQL> SELECT object_name, del, ins, sel, upd FROM DBA_OBJ_AUDIT_OPTS WHERE owner='VANZARI' and OBJECT_NAME in
('CLIENTI', 'FACTURI', 'LINIIFACT');

OBJECT_NAME          DEL       INS       SEL       UPD
-------------------- --------- --------- --------- ---------
CLIENTI              -/A       -/-       -/-       -/-
FACTURI              S/S       -/-       -/-       -/-
LINIIFACT            -/-       -/-       -/-       -/A
```



Cum interpretati semnificatia coloanelor DEL, INS, SEL, UPD? Ce înseamnă "S/S"? Dar "-/A"?

Cum se poate renunţa la auditarea pe o tabelă?

Testarea scenariilor de utilizare

Vom începe cu TMARIN care, ne amintim, avea rolul SERVICIU CONTABILITATE. Să testăm câteva scenarii de utilizare şi să vedem daca rolul asociat îşi face treaba cum se cuvine. Ne conectăm cu utilizatorul PSILVIU şi rulăm următoarele comenzi (parola e PSILVIU, cu majuscule):

```
connect PSILVIU/PSILVIU
insert into vanzari.clienti(codcl, dencl, codpost) values('4006', 'XY SRL', 700505);
COMMIT;

insert into vanzari.clienti(codcl, dencl, codpost) values('4007', 'YZ SRL', 701150);

insert into vanzari.facturi(nrfact, datafact, codcl) values(200001,
TO_DATE('01/09/2010', 'dd/MM/yyyy'), 4006);

insert into vanzari.liniifact values(200001, 1, 1, 50, 100, NULL);

insert into vanzari.liniifact values(200001, 2, 2, 10, 250, NULL);

insert into vanzari.liniifact values(200001, 3, 3, 1, 150, NULL);

insert into vanzari.liniifact values(200001, 4, 4, 30, 150, NULL);

COMMIT;

UPDATE vanzari.LINIIFACT SET cantitate = 0, pretunit = 0 WHERE nrfact IN
(SELECT nrfact FROM vanzari.FACTURI WHERE codcl = 4006);

DELETE FROM vanzari.FACTURI WHERE codcl = 4006;
```

Explicaţi rezultatul obţinut după rularea instrucţiunilor de mai sus. E PSILVIU autorizat să execute doar în limita drepturilor acordate?

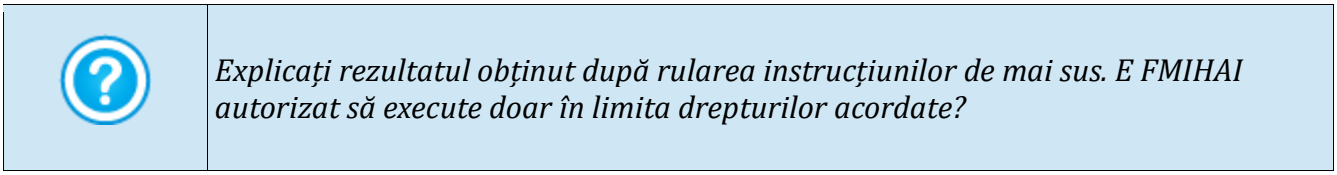
Să încercăm un scenariu de utilizare şi conectaţi ca TMARIN.

```
connect TMARIN/TMARIN
SELECT * FROM vanzari.FACTURI WHERE nrfact = 200001;

SELECT codcl, dencl FROM vanzari.CLIENTI WHERE codcl IN (SELECT codcl FROM vanzari.FACTURI WHERE nrfact =
200001);

SELECT * FROM vanzari.PRODUSE WHERE codpr IN (SELECT codpr FROM vanzari.LINIIFACT WHERE nrfact = 200001);

UPDATE vanzari.liniifact SET cantitate = 20 WHERE nrfact = 200001 AND linie = 2;

COMMIT;

DELETE FROM vanzari.clienti WHERE dencl = 'YZ SRL';
```

Explicaţi rezultatul obţinut după rularea instrucţiunilor de mai sus. E TMARIN autorizat să execute doar în limita drepturilor acordate?

În sfârşit, ca să nu se supere FMIHAI (mai ales ca e şef), să vedem un scenariu de utilizare şi cu acest utilizator:

```
connect FMIHAI/FMIHAI
SELECT * FROM vanzari.FACTURI WHERE datafact = TO_DATE('01/08/2010', 'DD/MM/YYYY');

SELECT codcl, dencl FROM vanzari.CLIENTI WHERE codcl IN (SELECT codcl FROM vanzari.FACTURI WHERE datafact =
TO_DATE('01/08/2010' ,'DD/MM/YYYY'));

SELECT * FROM vanzari.PRODUSE WHERE codpr IN (SELECT codpr FROM vanzari.LINIIFACT WHERE nrfact IN (SELECT nrfact
FROM vanzari.FACTURI WHERE datafact  = TO_DATE('01/08/2010', 'DD/MM/YYYY')));

DELETE FROM vanzari.liniifact WHERE nrfact IN ((SELECT nrfact FROM vanzari.LINIIFACT WHERE nrfact IN (SELECT
nrfact FROM vanzari.FACTURI WHERE datafact  = TO_DATE('01/08/2010', 'DD/MM/YYYY'))));
```



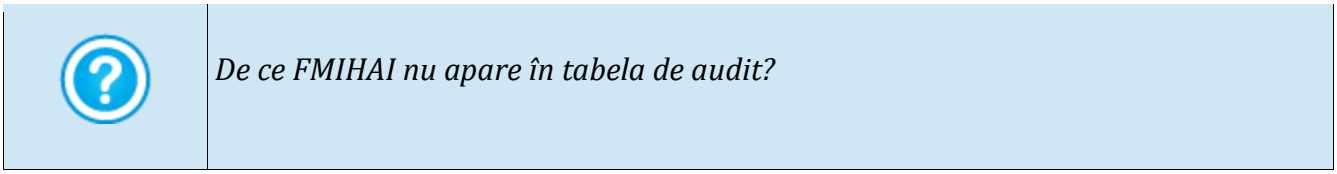
Explicaţi rezultatul obţinut după rularea instrucţiunilor de mai sus. E FMIHAI autorizat să execute doar în limita drepturilor acordate?

Accesarea datelor de audit

Datele de audit le găsim în view-ul sistem DBA_AUDIT_TRAIL. Ne amintim că am activat deja auditul pentru cateva tipuri de operatiuni in baza de date, prin urmare, in urma scenariilor de utilizare pe care le-am testat, ar trebui să avem ceva date în audit. Să vedem:

```
SQL> connect / as sysdba
Connected.
SQL> desc DBA_AUDIT_TRAIL
 Name                                                                    Null?    Type
 ----------------------------------------------------------------------- -------- --------------------
 OS_USERNAME                                                                      VARCHAR2(255)
 USERNAME                                                                         VARCHAR2(128)
 USERHOST                                                                         VARCHAR2(128)
 TERMINAL                                                                         VARCHAR2(255)
 TIMESTAMP                                                                        DATE
 OWNER                                                                            VARCHAR2(128)
 OBJ_NAME                                                                         VARCHAR2(128)
 ACTION                                                                  NOT NULL NUMBER
 ACTION_NAME                                                                      VARCHAR2(28)
 NEW_OWNER                                                                        VARCHAR2(128)
 NEW_NAME                                                                         VARCHAR2(128)
 OBJ_PRIVILEGE                                                                    VARCHAR2(16)
 SYS_PRIVILEGE                                                                    VARCHAR2(40)
 ADMIN_OPTION                                                                     VARCHAR2(1)
 GRANTEE                                                                          VARCHAR2(128)
 AUDIT_OPTION                                                                     VARCHAR2(40)
 SES_ACTIONS                                                                      VARCHAR2(19)
 LOGOFF_TIME                                                                      DATE
 LOGOFF_LREAD                                                                     NUMBER
 LOGOFF_PREAD                                                                     NUMBER
 LOGOFF_LWRITE                                                                    NUMBER
 LOGOFF_DLOCK                                                                     VARCHAR2(40)
 COMMENT_TEXT                                                                     VARCHAR2(4000)
 SESSIONID                                                               NOT NULL NUMBER
 ENTRYID                                                                 NOT NULL NUMBER
 STATEMENTID                                                             NOT NULL NUMBER
 RETURNCODE                                                              NOT NULL NUMBER
 PRIV_USED                                                                        VARCHAR2(40)
 CLIENT_ID                                                                        VARCHAR2(128)
```

```
 ECONTEXT_ID                                                       VARCHAR2(64)
 SESSION_CPU                                                       NUMBER
 EXTENDED_TIMESTAMP                                                TIMESTAMP(6) WITH TIME ZONE
 PROXY_SESSIONID                                                   NUMBER
 GLOBAL_UID                                                        VARCHAR2(32)
 INSTANCE_NUMBER                                                   NUMBER
 OS_PROCESS                                                        VARCHAR2(16)
 TRANSACTIONID                                                     RAW(8)
 SCN                                                               NUMBER
 SQL_BIND                                                          NVARCHAR2(2000)
 SQL_TEXT                                                          NVARCHAR2(2000)
 OBJ_EDITION_NAME                                                  VARCHAR2(128)
 DBID                                                              NUMBER

SQL> column owner format a10
SQL> column obj_name format a15
SQL> SELECT USERNAME, TIMESTAMP, OWNER, OBJ_NAME, ACTION_NAME FROM DBA_AUDIT_TRAIL WHERE username IN ('FMIHAI',
'PSILVIU', 'TMARIN');

USERNAME             TIMESTAMP          OWNER      OBJ_NAME        ACTION_NAME
-------------------- ------------------ ---------- --------------- ----------------------------
PSILVIU              08-MAY-17          VANZARI    FACTURI         SESSION REC
PSILVIU              08-MAY-17          VANZARI    CLIENTI         DELETE
TMARIN               08-MAY-17          VANZARI    CLIENTI         DELETE
PSILVIU              08-MAY-17          VANZARI    LINIIFACT       UPDATE
TMARIN               08-MAY-17          VANZARI    LINIIFACT       UPDATE
```



De ce FMIHAI nu apare în tabela de audit?

Activarea auditului de tip FGA

FGA vine de la Fine Grained Auditing şi e o optiune disponibilă doar în varianta Oracle Enterprise Edition. Cu ajutorul acestei facilităţi putem audita într-o manieră foarte flexibilă şi granulară. Spre exemplu, putem audita doar când o anumită coloană este accesată şi doar pentru anumite înregistrări, dacă e cazul. Mai întâi va trebui să pregătim puţin terenul. Avem nevoie de câteva privilegii acordate schemei VANZARI:

```
connect / as sysdba
GRANT EXECUTE ON dbms_fga TO vanzari;
GRANT select ON dba_audit_policies TO vanzari;
GRANT select ON dba_fga_audit_trail TO vanzari;
```

Apoi vom crea o tabelă în care dorim să colectăm datele de audit şi o procedură stocată care va fi invocata automat de mecanismul FGA şi în care vom controla cum şi unde dorim să facem scrierea auditului.

```
connect vanzari/Vanzari123

CREATE TABLE fga_vanzari_trail (
  owner       VARCHAR2(30),
  table_name  VARCHAR2(30),
  policy_name VARCHAR2(30),
  sql_text VARCHAR(1000),
```

```
  data_tranzact DATE,
  usr VARCHAR(100)
);

CREATE OR REPLACE PROCEDURE fga_vanzari_handler (
sname VARCHAR2, tname VARCHAR2, pname VARCHAR2) IS
PRAGMA AUTONOMOUS_TRANSACTION;
BEGIN
  INSERT INTO fga_vanzari_trail
    (owner, table_name, policy_name, sql_text, data_tranzact, usr)
  VALUES
    (sname, tname, pname,
    SYS_CONTEXT('USERENV','CURRENT_SQL'),
    SYSDATE,
    SYS_CONTEXT('USERENV','CURRENT_USER'));
  COMMIT;
END;
/
```

Care este rolul sintagmei PRAGMA AUTONOMOUS_TRANSACTION?

În sfârşit, va trebui să creăm o politică de audit de tip FGA. Să presupunem că dorim auditarea tabelei FACTURI pentru toate operaţiile de tip SELECT şi INSERT care implică folosirea coloanei VALFACT şi dacă, şi numai dacă, VALFACT > 2000. Cu alte cuvinte, nu ne batem capul cu mizilicuri, ci doar cu valori suspect de mari ale facturii. Creăm politica de audit după cum urmează:

```
connect vanzari/Vanzari123

BEGIN
dbms_fga.add_policy(
  object_schema=>'VANZARI',
  object_name=> 'FACTURI',
  policy_name=> 'Facturi_AUDIT',
  audit_condition=>  'valfact > 2000',
  audit_column=> 'VALFACT',
  handler_schema=> 'VANZARI',
  handler_module=> 'FGA_VANZARI_HANDLER',
  enable => TRUE,
  statement_types => 'SELECT, INSERT',
  audit_trail => DBMS_FGA.DB + DBMS_FGA.EXTENDED,
  audit_column_opts => dbms_fga.all_columns);
END;
/
```

Pentru a vedea toate politicile de audit puteţi interoga view-ul DBA_AUDIT_POLICIES.

Testarea politicii de audit FGA

Suspectăm că şefu', PSILVIU, face "golănii" prin baza de date, mai ales că are drepturi de INSERT pe tabela FACTURI. Să presupunem că, intr-adevăr, PSILVIU rulează următorul INSERT:

```
connect PSILVIU/PSILVIU
insert into vanzari.facturi(nrfact, datafact, codcl, valfact) values(200002, TO_DATE('01/09/2010',
'dd/MM/yyyy'), 4006, 3500);
```

PSILVIU îşi dă seama de gravitatea faptei şi decide să facă ROLLBACK:

```
rollback;
```

Să vedem acum dacă politica noastră de audit a funcţionat:

```
SQL> connect vanzari/Vanzari123
Connected.
SQL> select * from fga_vanzari_trail;

OWNER      TABLE_NAME                     POLICY_NAME
---------- ------------------------------ ------------------------------
SQL_TEXT
--------------------------------------------------------------------------------------------------------------------
--------
DATA_TRANZACT      USR
------------------ ---------------------------------------------------------------------------------------------
-------
VANZARI    FACTURI                        FACTURI_AUDIT
insert into vanzari.facturi(nrfact, datafact, codcl, valfact) values(200002, TO_DATE('01/09/2010', 'dd/MM/yyyy'),
4006,
3500)
08-MAY-17          VANZARI
```

Testaţi ce se întâmplă daca PSILVIU ar face un insert cu o valoarea a facturii de 1000.

Asocierea unui profil utilizatorilor Oracle

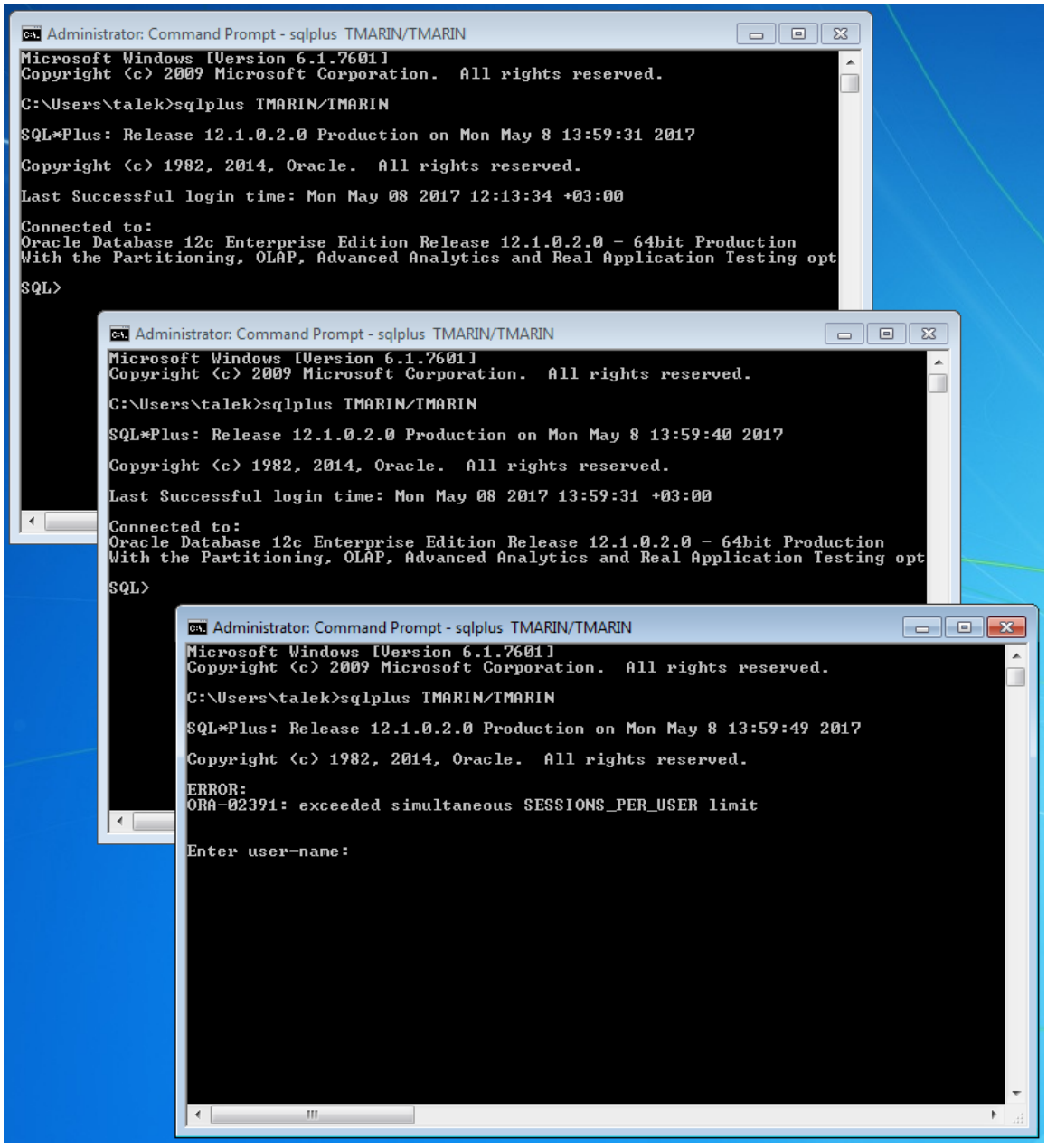
Un profil Oracle reprezintă o colecţie de proprietăţi prin care se pot controla aspecte ce tin de:

- termenul de valabilitate a parolei
- cat timp o conexiune a unui anumit utilizator poate sta inactiva
- numarul maxim de conexiuni ale aceluiaşi utilizator
- limite la nivelul resurselor server pe care un utilizator le poate folosi
- şi multe altele

Cu titlu de exemplu, să presupunem că simpaticii de la "Contabilitate" deschid foarte multe conexiuni la baza de date şi "cocoşează" serverul. Cică au ei de făcut multe rapoarte simultan. Nu ne place asta, prin urmare vom crea un profil prin care vom limita numarul de conexiuni ale lui GFLAVIU şi ale lui TMARIN, utilizatorii arondaţi la departamentul CONTA, la maximum 2. Mai întai, cu SYS, creăm un nou profil:

```
SQL> conn / as sysdba
Connected.
SQL> create profile conta_profile limit sessions_per_user 2;

Profile created.

SQL> alter user tmarin profile conta_profile;

User altered.

SQL> alter user gflaviu profile conta_profile;

User altered.
```

Pentru a testa, deschideți trei conexiuni noi cu utilizatorul TMARIN:



Ar trebui ca la a treia conexiune Oracle să arunce un mesaj de eroare, mai mult sau mai puțin explicit.