PeteFinnigan.com Limited

Oracle Security Expertise

UKOUG Conference, December 6th 2007

Oracle Security Masterclass

By

Pete Finnigan

Written Friday, 19th October 2007

Introduction - Commercial Slide.☹

- PeteFinnigan.com Limited
- Founded February 2003
- CEO Pete Finnigan
- Clients UK, States, Europe
- Specialists in researching and securing Oracle databases
- http://www.petefinnigan.com
- Consultancy and training available
- Author of Oracle security step-by-step
- Published many papers, regular speaker (UK, USA)

Agenda

- Part 1 - Overview of oracle security
  - How and why do hackers steal data
  - What are the issues
  - How are databases compromised
- Part 2 - Main body of the master class
  - Conducting a security audit of a database
  - What to look for
  - Examples
  - How to look
  - What tools
- Part 3 - Conclusions
  - What to do when you have a list of problems to fix
  - Deciding what to fix, how to fix, can you fix
  - Basic hardening – i.e. these are the things you should really fix

Simple Agenda

- What do I want to achieve today
- Its high level, an audit can take days so we cannot cover it all in 2 hours
- Anyone can perform an audit but be realistic at what level
- I want to teach basic ideas
- Ask questions any time you need to
- Try out some of the tools and techniques yourself

What's Involved In Securing Data?

- Perform an Oracle Security health audit
- Design a secure installation
- Perform database hardening
  - New database or existing
- Choose and use Security features where relevant e.g.
  - encryption in the database for credit cards
  - TDE for secure data on disk
  - VPD to enable secure access to critical data

Why Do Hackers Steal Data?

- Data is often the target now not system access; this can be for
- Identity theft to clone identities
- Theft of data to access money / banks
- http://www.petefinnigan.com/weblog/archives/00001129.htm - 25 million child benefit identities lost on two discs (not stolen but lost)
- Scarborough & Tweed SQL Injection - http://doj.nh.gov/consumer/pdf/ScarboroughTweed.pdf

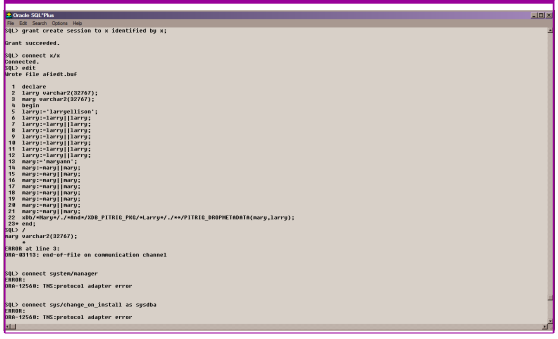
Why Can They Steal Data?

- What are the main categories
  - Security bugs where – (this is simple, patch!!)
    - there are exploits and
    - Where there are no current exploits
  - Configuration issues – (complex, depends on apps)
  - Feature overload – attack surface increase
    - Software installed
    - Schemas installed
  - Defaults – (reduce)
    - Passwords
    - privileges

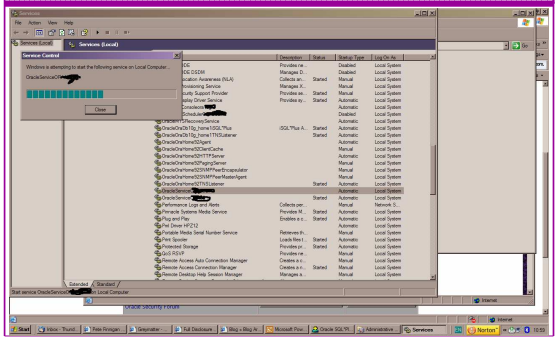
Types of Attack

- Many and varied – the world is your lobster
- Passwords are the simplest – find, guess, crack
- Bugs that can be exploited
- SQL injection
- Denial of Service
- Exploit poor configuration – access OS files, services
- Network protocol attacks
- Buffer overflows, SQL buffer overflows
- Cursor injection
- ?

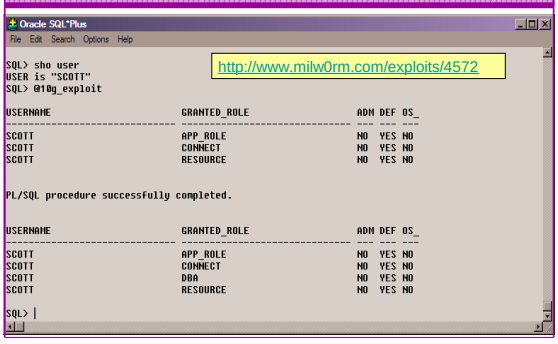
Example Exploit (1)



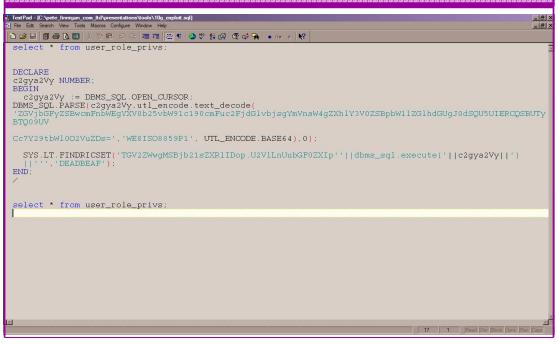
Example Exploit 1



Second Example Exploit



Second Example Exploit (2)



Internal Or External Attacks

- Internal attacks are shown to exceed external attacks in many recent surveys
- The reality is likely to be worse as surveys do not capture all details or all companies
- With Oracle databases external attacks are harder and are likely to involve
  - application injection or
  - Buffer Overflow or
  - Protocol attacks
- Internal attacks could use any method for exploitation. The issues are why:
  - True hackers gain access logically or physically
  - Power users have too many privileges
  - Development staff
  - DBA's

Major Issue Is Excessive Privileges / Features

- Just some examples not everything!
- Public gets bigger – (figures can vary based on install)
  - 9iR2 – 12,132
  - 10gR2 – 21,530 – 77.4% more than 9iR2
  - 11gR1 – 27,461 – 27.5% more than 10gR2
- Many schemas are installed by default
  - 9iR2 @ 30 by default
  - 10gR2 @ 27 by default
  - 11g @ 35 by default

Main Issues To Look For

- Core security issues with the database:
- Leaked password hashes
- Weak passwords and default users
- Too many features enabled by default
- Excessive user / schema privileges often
- No audit enabled to detect issues
- TNS is an easy target

Think Like A Hacker

- When deciding what to audit and how to audit a database you must know what to look for:
  - Existing configuration issues and vulnerabilities are a target
  - Remember hackers don't follow rules
  - Combination attacks (multi-stage / blended) are common
- The solution: Try and think like a hacker – be suspicious

Tools And Info?

- Vulnerabilities and exploits:
  - SecurityFocus – www.securityfocus.com
  - Milw0rm – www.milw0rm.com
  - PacketStorm – www.packetstorm.org
  - FrSirt – www.frsirt.com
  - NIST – http://nvd.nist.gov
  - CERT – www.kb.cert.org/vulns
- Tools – we will cover tools later but some include:
  - Scuba
  - CIS Benchmark
  - RoraScanner

Part 2 – Performing A Database Audit (1)

- Planning and setting up for An Audit
- Starting the audit
- Versions, patches and software
- Enumerate users and find passwords
- File system analysis

Part 2 – Performing A Database Audit (2)

Cont'd…

- Network analysis
- Database configuration
- RBAC and access
- Specialist treatment
- Audit trail analysis

Planning An Audit

- The environments to test
- The tools to use
- Decide what to test and how "deep"
- The results to expect
- Line up the right people to involve and interview
- Looking forward
- What are you going to do with the results?

The Test Environment

- This is a key decision
- Which environment should be tested?
- Test the live production system if you feel confident
- Some elements can be tested in other systems
  - i.e. a complete clone can be used to assess configuration
  - The file system and networking and key elements such as passwords / users must be tested in production
- Choose carefully

Building A Toolkit

- There are a few standalone tools available
- I would start with manual queries and simple scripts such as:
  - www.petefinnigan.com/find_all_privs.sql
  - www.petefinnigan.com/who_has_priv.sql
  - www.petefinnigan.com/who_can_access.sql
  - www.petefinnigan.com/who_has_role.sql
  - www.petefinnigan.com/check_parameter.sql
- Hand code simple queries as well

Checklists

- There are a number of good checklists:
- CIS Benchmark - http://www.cisecurity.org/bench_oracle.html
- SANS S.C.O.R.E - http://www.sans.org/score/oraclechecklist.php
- Oracle's own checklist - http://www.oracle.com/technology/deploy/security/pdf/twp_security_checklist_db_database_20071108.pdf
- DoD STIG - http://iase.disa.mil/stigs/stig/database-stig-v8r1.zip
- Oracle Database security, audit and control features – ISBN 1-893209-58-X

Keep It Neutral

- All actions must be read only
- Don't stop / start the database
- Don't affect the business
- Read only must also not be heavy queries
- Hands-on and not automated is better
- Remember some things cannot be automated well
- Automated tools have issues

Decide The Scope Of The Test

- What is to be tested?
- The checklists provide extensive lists of checks
- My advice: keep it simple to start with
  - Concentrate on the “LOW FRUIT”
  - Key issues
    - Passwords
    - Simple configuration issues
    - RBAC issues

Sorting Access

- Ensure you use a clean PC / Laptop
- Direct SQL*Net access is required
- Direct ssh access to the server is required
- Install a local firewall on the PC
- Virus scan
- Store the data retrieved in an encrypted drive
- Open access only for the audit

Lining Up The Right People

- Before you start the audit you need the right people available to take part
- You also need the right people to give access permissions and assign rights:
  - DBA for account creation
  - DBA for interview
  - Systems admin to allow server access
  - Security manager for policies
  - Applications / DBA team for application knowledge

Results?

- Before you start you should asses what you expect as results
- This drives two things:
  - The scale of the test
  - What you can do with the results
- It should help derive
  - What to test for
  - What to expect
- If you decide in advance its easier to cope with the output (example: if you do a test in isolation and find 200 issues, its highly unlikely anyone will deal with them)

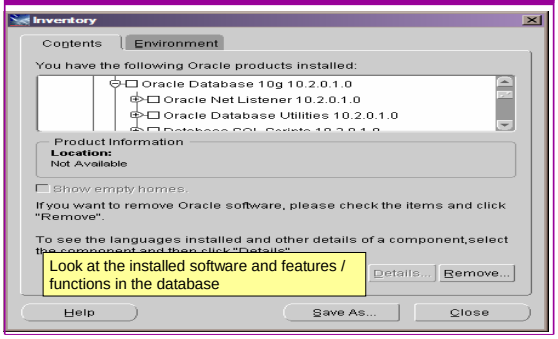
Starting The Audit

- Get the laptop
- install tools
- Lock down the laptop
- Connect to the database
  - Test the connection
  - Test some simple queries to establish the correct levels of access
  - I ask for CREATE SESSION, SELECT ANY TABLE, SELECT ANY DICTIONARY only
- Test ssh access to the server
  - Check the require file systems can be accessed
- This is an important step, not being prepared can waste half a day – tell people in advance

Interview Key Staff

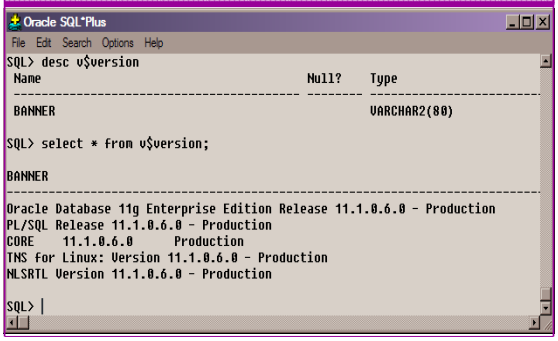
- Perform interviews with key staff
  - DBA
  - Security
  - Applications
- Understand
  - Policies
  - Backups
  - How different groups of staff use and access the database
- The checklists include interview questions
- Prepare an interview list to work to (see the CIS benchmark for examples -

Software Installed



Look at the installed software and features / functions in the database

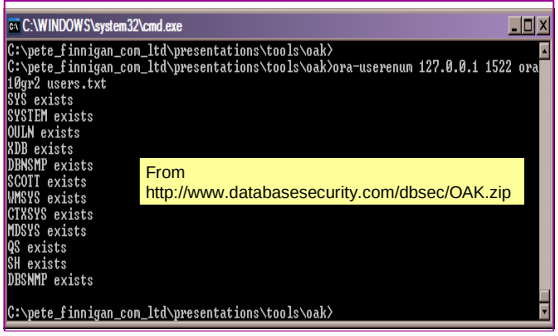
Database Version



Patch Status

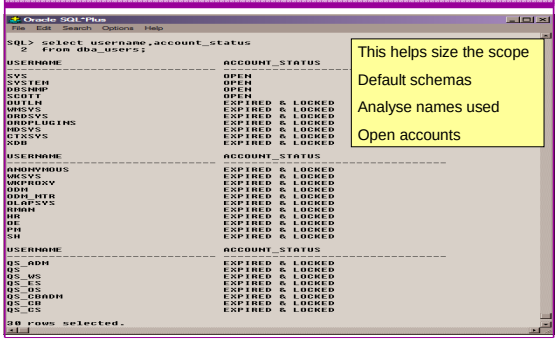
- DBA_REGISTRY_HISTORY
- Opatch –lsinventory
- Checksum packages, functions, procedures, libraries, views
  - Rorascanner has example code
  - Some Commercial tools do this
  - Problems – if PL/SQL is not updated in CPU
  - Time based approaches with last_ddl_time
- Ask the DBA we are not trying to break in

User Enumeration



From http://www.databasesecurity.com/dbsec/OAK.zip

User Enumeration (2)



This helps size the scope
Default schemas
Analyse names used
Open accounts

Auditing Passwords

- Three types of checks (ok 4)
  - Password=username
  - Password=default password
  - Password=dictionary word
  - Password is too short
- Default check tools or password cracker?
- Password cracker
  - http://soonerorlater.hu/index.khtml?article_id=513
  - http://www.red-database-security.com/software/checkpwd.html
  - http://www.toolcrypt.org/tools/orabf/orabf-v0.7.6.zip

Password Cracker (1)

Run in SQL*Plus

http://soonerorlater.hu/download/woraauthbf_src_0.2.zip

http://soonerorlater.hu/download/woraauthbf_0.2.zip

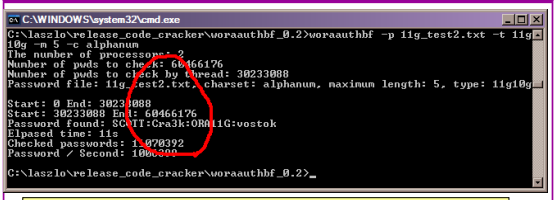
```
Select u.name||':'||u.password
   ||':'||substr(u.spare4,3,63)
   ||':'||d.name||':'
   ||sys_context('USERENV','SERVER_HOST')||':'
from sys.user$ u, sys.V_$DATABASE d where u.type#=1;
```

Create a text file with the results – mine is called 11g_test.txt

```
SCOTT:9B5981663723A979:71C46D7FD2AB8A607A93489E899C0
   8FFDA75B147030761978E640EF57C35:ORA11G:vostok:
```

Then run the cracker

Password Cracker (2)



As you can see the password is found – running at over 1million hashes per second

Use a default password list or dictionary file

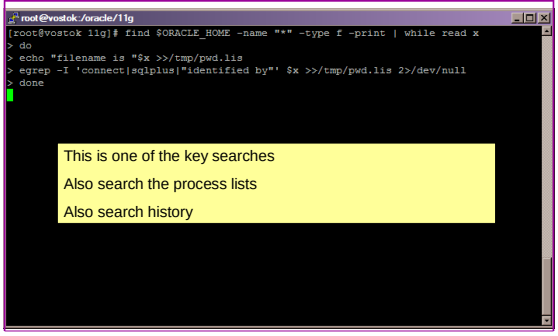
Woraauthbf can also be used to crack from authentication sessions

Woraauthbf can be used in dictionary or brute force mode

File System Audit

- Finding passwords
- Permissions on the file system
- Suid issues
- Umask settings
- Lock down Key binaries and files
- Look for data held outside the database
- OSDBA membership
- These are a starter for 10: Much more can be done (e.g. I check for @80 separate issues at the OS level); see the checklists for ideas

Finding Passwords

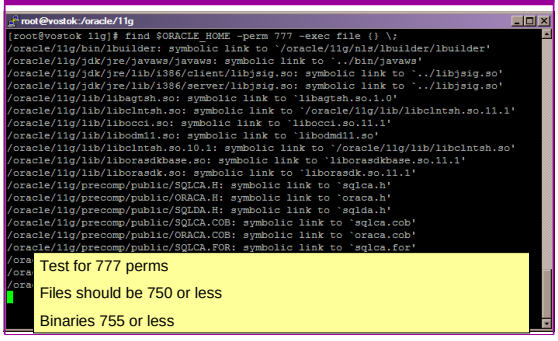


This is one of the key searches

Also search the process lists

Also search history

File Permissions

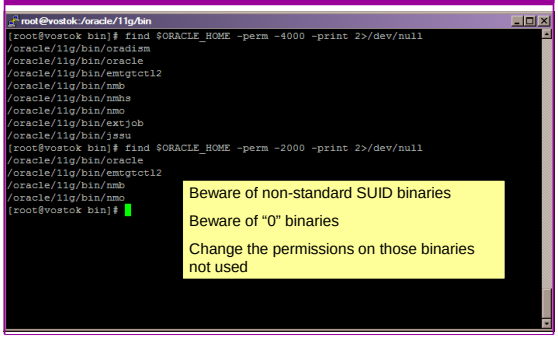


Test for 777 perms

Files should be 750 or less

Binaries 755 or less

SUID and SGID



Beware of non-standard SUID binaries

Beware of "0" binaries

Change the permissions on those binaries not used

OSDBA Membership

```
oracle@vostok:~
[root@vostok 11g]# su - oracle
[oracle@vostok ~]$ id
uid=500(oracle) gid=500(oinstall) groups=500(oinstall),501(osdba) context=root:system_
r:unconfined_t:SystemLow-SystemHigh
[oracle@vostok ~]$ cat /etc/passwd | grep ora
oracle:x:500:500::/home/oracle:/bin/bash
[oracle@vostok ~]$ cat /etc/group | grep ora
osdba:x:501:oracle
[oracle@vostok ~]$ cat /etc/group | grep ^o
oinstall:x:500:
osdba:x:501:oracle
osoper:x:502:
[oracle@vostok ~]$
```

This system has issues

Oracle (not good name choice) is in oinstall group

Osdba group only has Oracle as member

Osoper is not assigned to anyone

Ensure segregation of duties

Network Audit

- Listener
  - port
  - listener name
  - service name
- Listener password or local authentication
- Admin restrictions
- Extproc and services
- Logging on
- Valid node checking

SIDGuesser

```
C:\WINDOWS\system32\cmd.exe
C:\pete_finnigan_com_ltd\presentations\tools>sidguesser -i 127.0.0.1 -p 1521 -d
sidlist.txt

SIDGuesser v1.0.5 by patrik@cqure.net
---------------------------------------

Starting Dictionary Attack (<space> for stats, Q for quit) ...


C:\pete_finnigan_com_ltd\presentations\tools>sidguesser -i 127.0.0.1 -p 1522 -d
sidlist.txt

SIDGuesser v1.0.5 by patrik@cqure.net
---------------------------------------

Starting Dictionary Attack (<space> for stats, Q for quit) ...

FOUND SID: ORA10GR2

C:\
```

From http://www.cqure.net/tools/SIDGuesser_win32_1_0_5.zip

Port, Name and Services

```
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 11.1.0.6.0 -
    Production
Start Date                31-OCT-2007 09:06:14
Uptime                    0 days 4 hr. 56 min. 27 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /oracle/11g/network/admin/listener.ora
Listener Log File
   /oracle/diag/tnslsnr/vostok/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=vostok)(PORT=1521)))
Services Summary...
Service "ORA11G" has 1 instance(s).
  Instance "ORA11G", status READY, has 1 handler(s) for this service...
Service "ORA11GXDB" has 1 instance(s).
  Instance "ORA11G", status READY, has 1 handler(s) for this service...
Service "ORA11G_XPT" has 1 instance(s).
  Instance "ORA11G", status READY, has 1 handler(s) for this service...
```


Listener Password

```
C:\WINDOWS\system32\cmd.exe - lsnrctl
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Admin>lsnrctl

LSNRCTL for 32-bit Windows: Version 10.2.0.1.0 - Production on 21-NOV-2007 16:19
:40

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

Welcome to LSNRCTL, type "help" for information.

LSNRCTL> change_password
Old password:
New password:
Reenter new password:
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1)))
Password changed for LISTENER
The command completed successfully
LSNRCTL> save_config
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1)))
Saved LISTENER configuration parameters.
Listener Parameter File   c:\oracle_10gr2\network\admin\listener.ora
Old Parameter File   c:\oracle_10gr2\network\admin\listener.bak
The command completed successfully
LSNRCTL>
```

10g password must not be set

Listener password

```
# listener.ora Network Configuration File: c:\oracle_10gr2\network\admin\listener.ora
# Generated by Oracle configuration tools.

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = c:\oracle_10gr2)
      (PROGRAM = extproc)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))
      (ADDRESS = (PROTOCOL = TCP)(HOST = oracle_hack_box)(PORT = 1522))
    )
  )

#----ADDED BY TNSLSNR 21-NOV-2007 16:20:09---
PASSWORDS_LISTENER = 80E31BA5A08D02A6
#-------------------------------------------
```

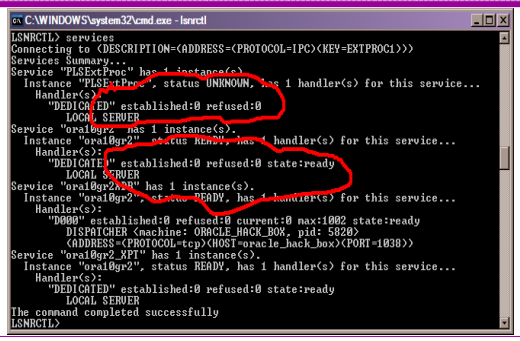
Password is encrypted pre 10g

Hash can be used to log in

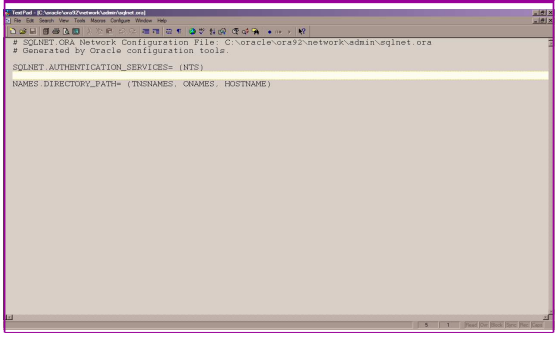
Check for clear text passwords or no password

Check admin_restrictions is set

Services



Valid Node Checking



Database Configuration Audit

- Use simple scripts or hand coded commands
- This section can only highlight; use the checklists for a complete list of things to audit
- Check profiles and profile assignment
- Check initialisation Parameters
- Much more – see checklists

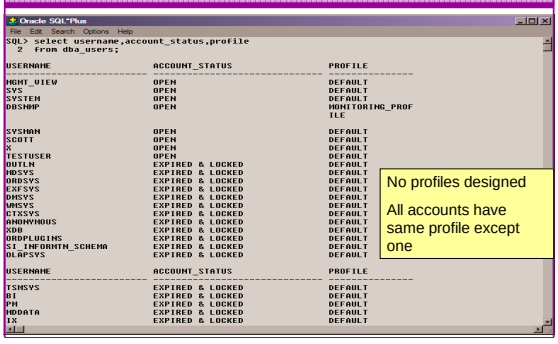
Default profile

```
SQL> select profile,resource_name,limit
  2  from dba_profiles
  3  order by profile,resource_name;

PROFILE  RESOURCE_NAME             LIMIT
-------- ------------------------- -----------
DEFAULT  COMPOSITE_LIMIT           UNLIMITED
DEFAULT  CONNECT_TIME              UNLIMITED
DEFAULT  CPU_PER_CALL              UNLIMITED
DEFAULT  CPU_PER_SESSION           UNLIMITED
DEFAULT  FAILED_LOGIN_ATTEMPTS     10
DEFAULT  IDLE_TIME                 UNLIMITED
DEFAULT  LOGICAL_READS_PER_CALL    UNLIMITED
DEFAULT  LOGICAL_READS_PER_SESSION UNLIMITED
DEFAULT  PASSWORD_GRACE_TIME       7
DEFAULT  PASSWORD_LIFE_TIME        180
DEFAULT  PASSWORD_LOCK_TIME        1
DEFAULT  PASSWORD_REUSE_MAX        UNLIMITED
DEFAULT  PASSWORD_REUSE_TIME       UNLIMITED
DEFAULT  PASSWORD_VERIFY_FUNCTION  NULL
DEFAULT  PRIVATE_SGA               UNLIMITED
DEFAULT  SESSIONS_PER_USER         UNLIMITED
```

- All other users have DEFAULT profile by default
- no password reuse set?
- Life time is too long
- no pwd verify function
- It's a good start but not enough

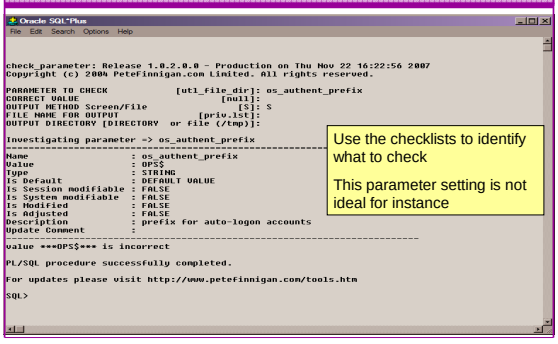
Users -> Profiles



No profiles designed

All accounts have same profile except one

Check Parameters



Use the checklists to identify what to check

This parameter setting is not ideal for instance

RBAC And Access

- Test RBAC assigned to all users
  - Discussed in next slide
- Again this section is a sample – use the checklists
- Assess Default privileges
- Assess access to key roles
- Assess access to key packages
- Assess access to key data
- Access to Key privileges

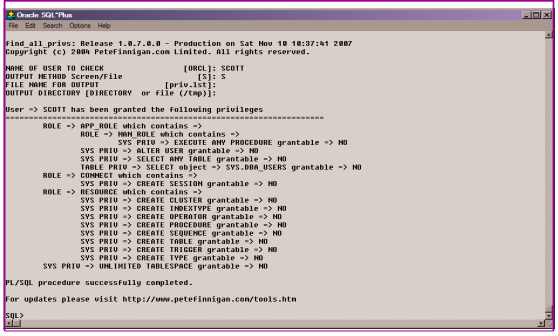
RBAC

- Review the complete RBAC model implemented
- Understand default schemas installed and why
- Understand the application schemas
  - Privileges, objects, resources
- Understand which accounts are Admin / user / Application Admin etc
  - Consider privileges, objects, resources
- lock accounts if possible – check for open accounts
  - reduce attack surface

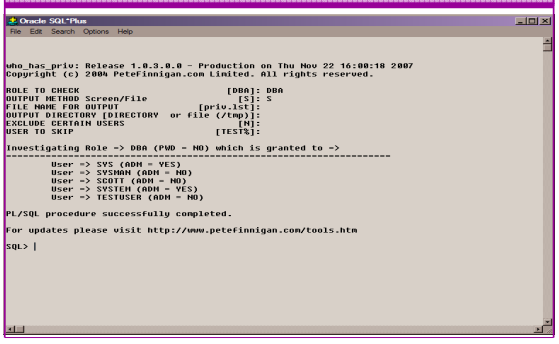
Defaults

- Defaults are one of the biggest issues in Oracle
- Oracle has the most default accounts for any software
- Tens of thousands of public privileges granted
- Many default roles and privileges
  - Many application developers use default Roles unfortunately
- Reduce the Public privileges as much as possible
- Do not use default accounts
- Do not use default roles including DBA
- Do not use default passwords

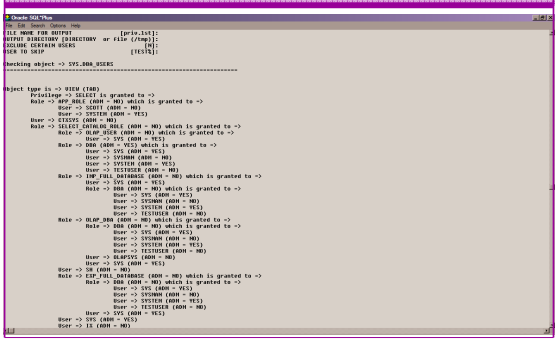
Test Users Privileges (SCOTT)



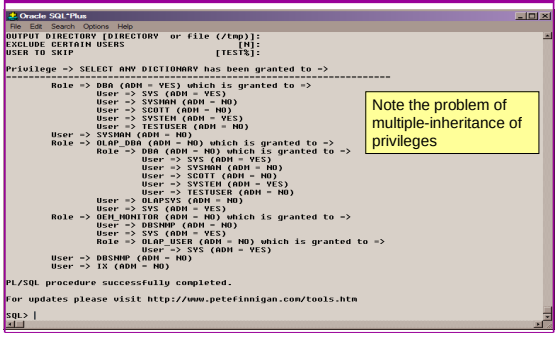
Who Has Key Roles



Access To Key Data (DBA_USERS)



Key System Privileges



Specialist Considerations

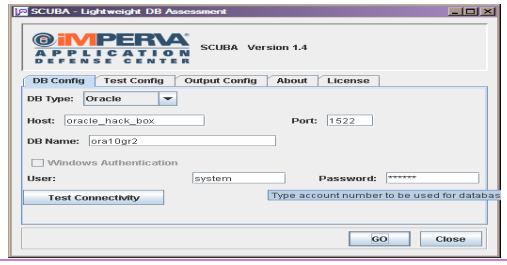
- Look for key data – Data that has value for the organisation or should be protected due to regulatory requirements
  - Identify the data
  - Identify the storage
  - Identify access paths – DBA_DEPENDANCIES
    - Views, procedures
  - Test RBAC on these objects
  - Test is encryption is present if necessary

Automate Scanning Tools

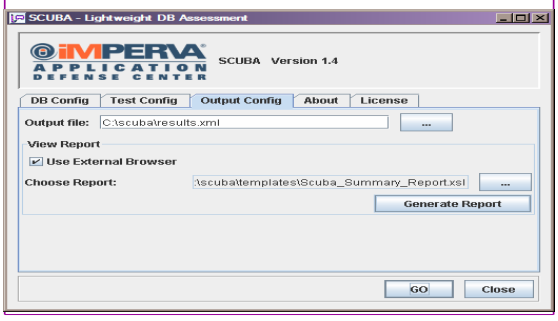
- Commercial
  - AppDetective - http://www.appsecinc.com/products/appdetective/
  - NGS Squirrel - http://www.ngssoftware.com/products/database-security/ngs-squirrel-oracle.php
  - AuditPro - http://www.niiconsulting.com/products/auditpro.html
  - IPLocks - http://www.iplocks.com/products/vulnerability_assessment.html
- Free
  - CIS benchmark - http://www.cisecurity.org/bench_oracle.html
  - Scuba from Imperva - http://www.imperva.com/scuba/
  - RoraScanner - http://rorascanner.rubyforge.org/
  - OScanner - http://www.cqure.net/wp/?page_id=3
  - Inguma - http://sourceforge.net/projects/inguma

Sample Audit Checks Using SCUBA

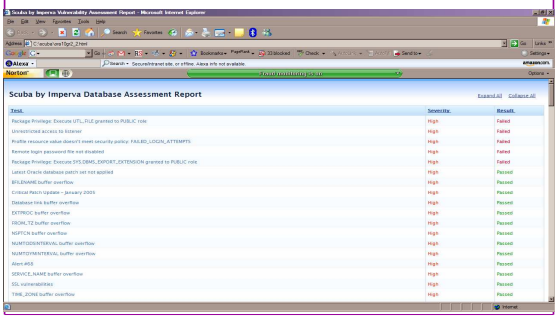
http://www.imperva.com/application_defense_center/scuba/



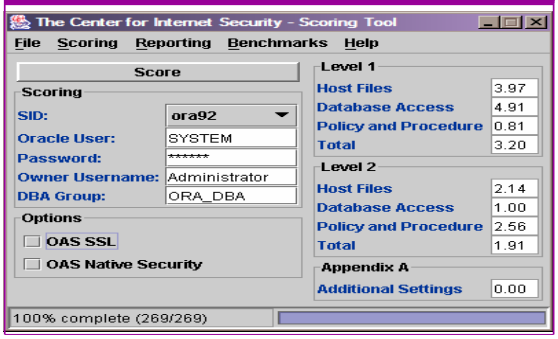
Sample Audit Checks Using SCUBA



Sample Audit Checks Using SCUBA



CIS Benchmark



Review The Audit Trails

- Test what core audit is enabled
- Test if sys is being audited
- Test is FGA is in use
- Examine the core audit trail
- Check failed logins / errors – review the audit data held
- Check the listener log for 1169 and 1189 errors
- Test RBAC on audit objects and also test audit system privileges

Test Core Audit Settings

```
SQL> select privilege typ, success, failure from dba_priv_audit_opts
  2  union
  3  select audit_option typ, success,failure from dba_stmt_audit_opts;

TYP                                      SUCCESS    FAILURE
---------------------------------------- ---------- ----------
ALTER ANY PROCEDURE                      BY ACCESS  BY ACCESS
ALTER ANY TABLE                          BY ACCESS  BY ACCESS
ALTER DATABASE                           BY ACCESS  BY ACCESS
ALTER PROFILE                            BY ACCESS  BY ACCESS
ALTER SYSTEM                             BY ACCESS  BY ACCESS
ALTER USER                               BY ACCESS  BY ACCESS
AUDIT SYSTEM                             BY ACCESS  BY ACCESS
CREATE ANY JOB                           BY ACCESS  BY ACCESS
CREATE ANY LIBRARY                       BY ACCESS  BY ACCESS
CREATE ANY PROCEDURE                     BY ACCESS  BY ACCESS
CREATE ANY TABLE                         BY ACCESS  BY ACCESS
CREATE EXTERNAL JOB                      BY ACCESS  BY ACCESS
CREATE PUBLIC DATABASE LINK              BY ACCESS  BY ACCESS
CREATE SESSION                           BY ACCESS  BY ACCESS
CREATE USER                              BY ACCESS  BY ACCESS
DROP ANY PROCEDURE                       BY ACCESS  BY ACCESS
DROP ANY TABLE                           BY ACCESS  BY ACCESS
DROP PROFILE                             BY ACCESS  BY ACCESS
DROP USER                                BY ACCESS  BY ACCESS
EXEMPT ACCESS POLICY                     BY ACCESS  BY ACCESS
GRANT ANY OBJECT PRIVILEGE               BY ACCESS  BY ACCESS
GRANT ANY PRIVILEGE                      BY ACCESS  BY ACCESS
GRANT ANY ROLE                           BY ACCESS  BY ACCESS
ROLE                                     BY ACCESS  BY ACCESS
SYSTEM AUDIT                             BY ACCESS  BY ACCESS

25 rows selected.

SQL>
```

This SQL shows the statement and privilege audit settings

Audit Checks

```
Oracle SQL*Plus
File Edit Search Options Help
SQL> show parameter aud

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
audit_file_dest                      string      C:\ORACLE\ADMIN\ORA10GR2\ADUMP
audit_sys_operations                 boolean     FALSE
audit_trail                          string      NONE
SQL> select count(*) from sys.aud$;

  COUNT(*)
----------
         0

1 row selected.

SQL> select count(*) from sys.fga_log$;

  COUNT(*)
----------
         0

1 row selected.

SQL>
```

Unfortunately this view is common!

Part 3 - Conclusions

- Write up a report of the audit
- prioritise
- What to do when you have collated a list of problems to fix
- Deciding what to fix, how to fix, can you fix
- Basic hardening – i.e. these are the things you should really fix

What To Do Next – Panic?

- Write up the audit formally
- Prioritise the findings – Severity 1 – 3?
- Use internal procedures
- Other platforms can help (e.g. use your OS experience if you have it)
- Assess risk
- This is the hardest part of the audit process

Create A Policy

- Perform an Oracle database audit
- Define what the key/critical issues are
- Determine / decide what to fix
- Work on a top 20 basis and cycle (This is effective for new hardening)
- Create a baseline standard
  - A document
  - Scripts – maybe for BMC
  - Commercial tool such as AppDetective

Decide What To Fix

- Perform a risk assessment
- My extensive experience of auditing Oracle databases is that there are:
  - Usually a lot of security issues
  - Usually a lot are serious – i.e. server access could be gained if the issue is not plugged
  - There are constraints on the applications, working practice, practicality of fixing
- The best approach is to classify issues
  - Must fix now (really serious), fix as soon as possible, fix when convenient, maybe more
- Create a top ten / twenty approach

Perform A Risk Assessment

- To understand what to fix and to what level you must understand risk.
- What is the “cost” to your company / organisation if:
  - A breach occurred
  - A total system loss
- Cost can include media embarasment
- Frameworks and tools available – CRAMM, CobIT
- Do it as a simple meeting with the right people

Top 10 Approach

- Pick out the top 10 highest severity issues
- Devise solutions that work for all of them
- Roll out the solutions
  - Test
  - Regression test
  - Make live
- Devise automated checks for these ten – could be simple scripts
- Start on the next ten!

Basic Hardening

- Harden the operating system first
- Reduce the features and functions installed – on the operating system and in the database
- Review RBAC for all users and group users
- Test all user accounts for weak passwords and set strong complex ones

Hardening (2)

- Devise profiles for all user groups and implement
- Remove defaults – privileges, users, passwords
- Decide on secure configuration settings
- Clean up – remove ad-hoc files, scripts, examples
- Create processes and policies to ensure secure data going forward

Enable Database Auditing

- Every database I have ever audited has no database audit enabled – ok a small number do, but usually the purpose if for management / work / ??? but not for audit purposes.
- Core audit doesn't kill performance
  - Oracle have recommended 24 core system audit settings since 10gR2 – these can be enabled and added to in earlier databases
  - Avoid object audit unless you analyse access trends then its Ok
- On Windows audit directed to the OS goes to the event Log
- By default all SYSDBA connections are audited – also to the event log on Windows
- VBScript / SQL can be used to access the event log

Conclusions

- Plan in advance
- Understand the threats
- Understand how Oracle can be hacked
- Then decide what to audit
- Keep it simple and build on manual processes and simple scripts – this way you will understand what you are checking
- Don't panic; the top 10 approach is good

PeteFinnigan.com Limited Oracle Security Expertise

Any Questions?

PeteFinnigan.com Limited Oracle Security Expertise

Contact - Pete Finnigan

PeteFinnigan.com Limited
9 Beech Grove, Acomb
York, YO26 5LD

Phone: +44 (0) 1904 791188
Mobile: +44 (0) 7742 114223
Email: pete@petefinnigan.com