#342 – Auditing Security of Oracle Database

Karel Miko, CISA
Consultancy Division, Director
DCIT a.s. (Czech Republic)

Contents

- **[A]** Oracle in a nutshell
- **[B]** Oracle security audit – phases
- **[C]** Auditing – operating system level
- **[D]** Auditing – Oracle RDBMS
- **[E]** Auditing – DB instances
- **[F]** Auditing – related processes
- **[G]** Live demonstration of Oracle DB audit

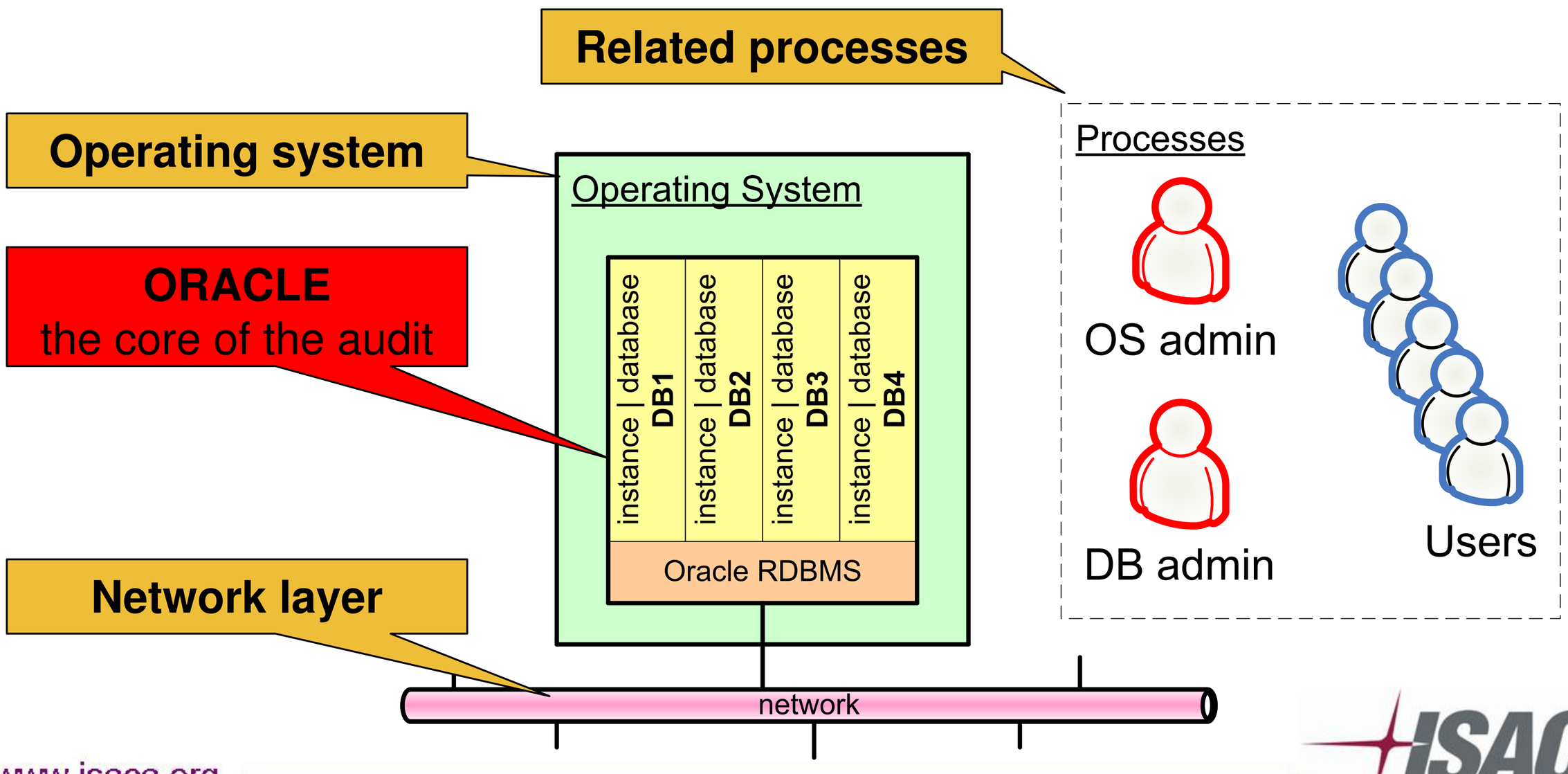
Oracle in a nutshell

[A] Oracle summary

- **Oracle** = one of the leading relational database technologies
- Terminology:
  - **RDBMS** Relational Database management System (installed SW components, no data)
  - **Instance** – logical entities, running processes, memory structures, …
  - **Database** – physical files containing data (datafiles, controlfiles, redologfiles, …)

[A] Oracle Security Audit

- **Oracle security** should be reviewed in a wider context – not only “pure Oracle”:



Oracle security audit phases

[B] Dividing oracle audit into phases

- **Before you start**
  - **If an external audit – contract (NDA – Non Disclosure Agreement)**
  - **Prepare the audit plan (schedule, scope)**
- **There are 4 technical phases**
  - **[C]** Auditing – operating system level
  - **[D]** Auditing – Oracle RDBMS
  - **[E]** Auditing – DB instances
  - **[F]** Auditing – related processes

Audit – phase 1
operating system level

[C] What we need before we start/1

- **Non-technical issues**
  - Define exactly the **subject**: what server(s), what database(s), test/production environment, …
  - Specify the **scope** (what areas and how “deep”)
  - Agree on the assessment **method**: only remote DB access, full access to the server, …
  - Agree on **tools** used for the audit (you might be required to prove confidence of these tools)
  - Line up the **key people**: DB admin, OS admin, Application architect (communicate in advance)

[C] What we need before we start/2

- **Technical issues**
  - **OS access** to the server
    - Usually SSH terminal access (UNIX) or Terminal Services (MS Windows)
    - Without OS access no chance to check OS security
    - Minimum OS access as an “oracle” user, however some security issues you can check only as a “root”
  - **DB access** to the server
    - At least DB account with SELECT ANY DICTIONARY and SELECT ANY TABLE privileges
    - Some issues you can check only as a DBA (e.g. SYS)
    - You can use remote DB access (over the net)

[C] Auditing the operating system of DB server

- **Detailed OS auditing is a separate branch of knowledge**
- Within the Oracle audit we check only a subset of OS security features (somehow related to Oracle)
- Oracle runs (unfortunately) at many OS platforms:
  - AIX, HP-UX, Linux, Solaris, HP Tru64, IBM z/OS, MAC OS X, HP OpenVMS, MS Windows
- An Oracle auditor has to be ready at least for **two major OS platforms**:
  - MS Windows
  - UNIX like systems

[C] Oracle installation ORACLE_HOME

- $ORACLE_HOME directory content:
  - \oracle = Oracle Home
  - \oracle\bin - contains executables
  - \oracle\network\admin - contains listener cfg file
  - \oracle\admin – contains bdump, cdump, udump
  - \oracle\dbs - contains spfiles
  - \oracle\rdbms\audit - contains audit tral (*.aud)
- More or less the same on all platforms

[C] Oracle installation ORACLE_HOME

- Basic permissions in $ORACLE_HOME directory:
  - **10g:** the most of the files should have
    - rwxr-x--- (owner oracle)
  - **10g:** files in $ORACLE_HOME/bin/*
    - rwxr-xr-x (owner oracle)
  - **9i:** the most of the files should have
    - rwxr-xr-x (owner oracle)
  - **9i:** no access for "others" or “Everyone” to these dirs:
    - $ORACLE_HOME/rdbms/audit
    - $ORACLE_HOME/rdbms/log
    - $ORACLE_HOME/network/trace

[C] Oracle installation (UNIX)/1

- Get the list of files & permissions

```
[oracle@db]$ ls -lR $ORACLE_HOME > orahome-list.txt
```

- Check non-standard permissions (10gR2) – writable by “group” or “others”

```
[oracle@db]$ find $ORACLE_HOME -perm +022 ! -type l
-exec ls -ld {} \;
(should return an empty list on Oracle 10g R2)
```

- Check world-writable files/directories

```
[oracle@db]$ find $ORACLE_HOME -perm -002 ! -type l
-exec ls -ld {} \;
(should return an empty list)
```

[C] Oracle installation (UNIX)/2

- Check **Set-uid/gid bits** (allows to switch effective user ID to file owner during runtime)
  - Particularly risky – files with s-bit owned by “root”
  - Check all s-bits and keep only where necessary

```
[oracle@db]$ find $ORACLE_HOME -perm +6000 -exec ls -ld
{} \;
(expected result for Oracle 10gR2)
-rwsr-s--x oracle oinstall /oracle/orahome/bin/oracle
-r-sr-s--- root oinstall   /oracle/orahome/bin/oradism
-rwsr-s--x oracle oinstall /oracle/orahome/bin/emtgtctl2
-rwsr-s--- root oinstall   /oracle/orahome/bin/nmb
-rwsr-s--- root oinstall   /oracle/orahome/bin/nmo
-rwsr-x--- root oinstall   /oracle/orahome/bin/extjob
```

[C] Oracle installation (UNIX)/3

- Check “backup” copies “*O” with s-bits

```
[oracle@db]$ find $ORACLE_HOME -name "*O" -perm +6000 -
exec ls -ld {} \;
(should return an empty list)
```

- Check for files not owned by “oracle:oinstall”

```
[oracle@db]$ find $ORACLE_HOME \( ! -group oinstall -o
! -user oracle \) -exec ls -ld {} \;
(expected result for Oracle 10gR2)
-r-sr-s--- root oinstall /oracle/orahome/bin/oradism
-rwsr-s--- root oinstall /oracle/orahome/bin/nmb
-rwsr-s--- root oinstall /oracle/orahome/bin/nmo
-rwsr-x--- root oinstall /oracle/orahome/bin/extjob
-rw-r----- root oinstall /oracle/orahome/rdbms/admin/
                         externaljob.ora
```

[C] Oracle installation (Windows)

- Unfortunately no standard tools in MS Windows for finding files and directories with risky permissions
- You have to either check ORACLE_HOME permissions manually or use a special tool
- **General rules** for ORACLE_HOME
  - Defaults are quite OK on Win2003 & Ora10gR2
  - Avoid any access for Everyone
  - Avoid create/modify rights for “wide” groups like Users, Domain users, Authenticated Users, …

[C] OS account used for running Oracle database/1

- **UNIX environment**
  - Can be chosen during installation – usu. “oracle”
  - Using standard/expected name for this account is not a “good-practice” (arising risks not high)
  - Never run Oracle DB under “root”
  - OS user “oracle” is **very powerful**
    - Owner of the whole installation
    - Has full access to any DB instance running on the server (without password!!!)
    - Taking control over “oracle” = control over ALL DATA

[C] OS account used for running Oracle database/2

- **Microsoft Windows environment**
  - By default Oracle on Windows runs as **LOCAL SYSTEM** – it is **extremely risky**
  - Oracle should run under **non-privileged local account** (not member of Local Administrators) – it is not easy to set up
  - Do not run Oracle under domain account and never install Oracle on a domain controller
  - It is a good idea to use Win Native Authentication
  - Check privs like Network/Local/ServiceLogon

[C] How to check running Oracle

- UNIX – find Oracle processes

```
[oracle@orabox ~]$ ps -ef | grep -i "ora_pmon"
oracle     4189  00:01:12 ora_pmon_firstdb
oracle     7347  00:00:59 ora_pmon_mindb
oracle     7635  00:01:22 ora_pmon_testdb
```

Running under OS account “oracle”

3 DB instances
- firstdb
- mindb
- testdb

- Windows – check system services

Running as SYSTEM

Service	Status	EXE	User
OracleOraDb10g_home1TNSListener	Running	TNSLSNR.EXE	SYSTEM
OracleServiceTESTDB1	Running	oracle.exe	SYSTEM
OracleServiceTESTDB2	ng	oracle.exe	SYSTEM

2 DB instances
- TESTDB1
- TESTDB2

[C] Special OS group/1

- Mapping DB privileges SYSOPER/SYSDBA to special OS groups:
  - SYSOPER (DB startup/shutdown, mount/dismount, …)
  - SYSDBA (full system privileges)
- **UNIX environment**
  - SYSOPER + SYSDBA is mapped to OS group “dba”
  - Any member of “dba” has full access to all databases (by default only “oracle” is member of “dba”)
  - Group name and mapping can be changed – check:

```
[ora]$ grep define.SS $ORACLE_HOME/rdbms/lib/config.c
#define SS_DBA_GRP "dba"
#define SS_OPER_GRP "dba"
```

[C] Special OS group/2

- **MS Windows environment**
  - **OSDBA** → to local MS Windows group ORA_DBA and ORA_<INSTANCENAME>_DBA
  - **OSOPER** → to local MS Windows group ORA_OPER and ORA_<INSTANCENAME>_OPER
  - In MS Win environment the group names cannot be changed. All ORA_* groups are empty by default.
- **The most important risk**
  - Any OS user that is member of “dba” or “ORA_DBA” is a threat to the database (full access without password !!!)
  - Check group membership

[C] OS user “oracle” (UNIX)

- Strong “oracle” password – absolutely crucial
- Check scheduled “crontab” and “at” jobs:

```
[oracle@orabox ~]$ crontab -l
...
[oracle@orabox ~]$ at -l
...
```

  – Scheduled scripts should not be readable by “group” or “others” (may contain passwords)
  – Scheduled scripts of “oracle” user should be owned only by “oracle” or “root”

[C] Filesystem – critical files/dirs could be also outside ORACLE_HOME

- Check permissions to **Oracle Critical files**
  - Directories mapped to Oracle through UTL_FILE parameter or through DIRECTORY objects
  - DataFiles, RedoLogFiles, ControlFiles
  - ArchiveLogs
  - CDUMP, BDUMP, UDUMP directories
  - Directory with audit trail (if set to OS)
  - Dynamic libraries mapped to Oracle as external libraries
  - *… and others discussed later*

[C] OS security generally

- Stop unnecessary network services:

```
[oracle@orabox ~]$ netstat -an | grep LISTEN
(shows all listening TCP ports)
```

- Review OS users / groups:
  - Files: /etc/passwd /etc/group
  - Avoid simple/default passwords
- Review current OS patch level / updates
- Check filesystem rights
  - e.g. unprotected scripts with passwords
- Check network settings (network shares, …)

Audit – phase 2
RDBMS level

[D] Oracle version/1

- **Version / patches / updates**:
  - Base versions 9i R2, 10g R1, 10g R2, 11g
  - Patchsets (10.2.0.2, 10.2.0.3, …)
  - Critical Patch update (CPUOct/07, CPUJan/08 ..)
  - Interim patches (6121242, 5556081, …)
  - Information available at metalink.oracle.com
- **Best practice:**
  - Keep supported patchset (supported = Oracle still releases quarterly CPUs)
  - Install latest Critical Patch Update

[D] Oracle version/2

- **Patchset**
  - Patchset installation = big step - could harm the running applications (the changes are usually quite significant)
  - Upgrade requires app vendor cooperation / heavy testing
- **Critical Patch Updates** (quarterly)
  - Fix mostly security vulnerabilities and other critical issues
  - Should be installed regularly (within routine IT processes)
- **Interim patches**
  - Installed to fix mostly malfunctions, performance issues ..
  - In fact not necessary to audit

[D] Oracle version/3

- **CPU January 2008** was released for:
  - Oracle 11g, version **11.1.0.6**
  - Oracle 10g release 2, versions **10.2.0.2**, **10.2.0.3**
  - Oracle 10g release 1, version **10.1.0.5**
  - Oracle 9i release 2, version **9.2.0.8**
- The above mentioned patchsets can be nowadays considered as “supported”

[D] Info about Oracle version

- Connect by sqlplus to any database and run:

```
SQL> select * from v$version;
BANNER
Oracle Database 10g Enterprise Edition Release
10.2.0.3.0 - Production
PL/SQL Release 10.2.0.3.0 - Production
CORE 10.2.0.3.0 Production
TNS for Linux: Version 10.2.0.3.0 - Production
NLSRTL Version 10.2.0.3.0 - Production
```

```
SQL> select action_time, version, comments from
dba_registry_history;
ACTION_TIME                    VERSION     COMMENTS
20-NOV-07 10.21.53.685858 AM   10.2.0.3.0  CPUOct2007
```

[D] Info about installed patches using opatch utility

```
[oracle@orabox ~]$ $ORACLE_HOME/OPatch/opatch lsinventory
Invoking OPatch 10.2.0.3.3
Oracle interim Patch Installer version 10.2.0.3.3
Copyright (c) 2007, Oracle Corporation.  All rights reserved..

Oracle Home       : /oracle/orahome
Central Inventory : /oracle/orabase/oraInventory
   from           : /etc/oraInst.loc
OPatch version    : 10.2.0.3.3
OUI version       : 10.2.0.3.0
OUI location      : /oracle/orahome/oui
Log file location : /oracle/orahome/cfgtoollogs/opatch/opatch2C                    PM.log

Lsinventory Output file location :
/oracle/orahome/cfgtoollogs/opatch/lsinv/lsinventory2008-01-19_
--------------------------------------------------------------------------------
Installed Top-level Products (2):

  e Database 10g                                                    10.2.0.1.0
  e Database 10g Release 2 Patch Set 2                              10.2.0.3.0
  are 2 products installed in this Oracle Home.

I  rim patches (32) :

Patch  6394981      : applied on Tue Nov 20 10:16:48 CET 2007
   Created on 16 Sep 2007, 11:56:18 hrs PST8PDT
   Bugs fixed:
     6394981
...
--------------------------------------------------------------------------------
OPatch succeeded.
```

Installed patchsets

patch 6394981

[D] Metalink – patch search

Patches - Mozilla Firefox

Soubor Úpravy Zobrazit Historie Záložky Nástroje Nápověda

https://metalink.oracle.com/metalink/plsql/f?p=200:10:12470906715640753487::::

ORACLE METALINK

Bookmarks Admin Profile Feedback Sign Out Help

Headlines | Knowledge | Software Configuration Manager | **Patches & Updates** | Forums | Certify

Patches

Quick Find: Knowledge Base — Go — Secure Enterprise Search — Advanced — Saved Searches

6436680	**Oracle Database Family**: Patch MERGE LABEL REQUEST ON TOP OF 10.2.0.3 FOR BUGS 4733582 AND 6397946	10.2.0.3	16-OCT-2007	170K	
6487758	**Oracle Database Family**: Patch MERGE LABEL REQUEST ON TOP OF 10.2.0.3 FOR BUGS 6397940 4587572	10.2.0.3	16-OCT-2007	118K	
6394981	**Oracle Database Family**: Patch MLR BUG FOR 10.2.0.3 FOR CPUOCT2007	10.2.0.3	16-OCT-2007	4.3M	
23	**Oracle Database Family**: Patch MERGE LABEL REQUEST ON TOP OF 10.2.0.3 FOR BUGS 5870523 AND MOLECULE 6397941	10.2.0.3	16-OCT-2007	189K	
4499629	**RDBMS Server**: Patch 10GR2: WRONG "SESSION UGA MEMORY" IN AWR REPORT	10.2.0.3	11-OCT-2007	443K	
5512159	**RDBMS Server**: Patch APPSST-DOCTYPE IS GETTING APPENDED IN THE XML WHEN PARSED AGAINST DTD IN 10G	10.2.0.3	10-OCT-2007	33K	

6394981

Bookmarks Admin Profile Feedback Sign Out Help

Legal Notices and Terms of Use | Privacy Statement

Hotovo | https://metalink.oracle.com/metalink/plsql... | metalink.oracle.com | Proxy: Nic | 148.87.36.12 | Oracle | Adblock

[D] Metalink – CPU availability info

ShowDoc - Mozilla Firefox

Soubor Úpravy Zobrazit Historie Záložky Nástroje Nápověda

https://metalink.oracle.com/metalink/plsql/f?p=130:14:12470906715640753487::::p14_dat

ShowDoc | Oracle® Database Critical Patch Updat...

Table 8 shows the patches available for Oracle Database, based on release and platform.

Table 8 Patch Availability for Oracle Database

Foot 1 Foot 2

Platform	9.2.0.8	9.2.0.8 DV[Foot 1]	10.1.0.5[Foot 2]	10.2.0.2	10.2.0.3
AIX 5L	6395038	6395078	6395024	6394997	6394981
HP Itanium	6395038	NA	6395024	6394997	6394981
HP-UX	6395038	6395078	6395024	6394997	6394981
IBM Linux S/390	NA	NA	6395024	NA	NA
IBM zLinux	NA	NA	6395024	6394997	6394981
IBM z/OS 390 (Server)	6395038	NA	6395024	6394997	6452276
Linux Itanium	6395038	NA	6395024	6394997	6394981
Linux on Power	NA	NA	6395024	6394997	6394981
Linux x86	6395038	NA	6395024	6394997	6394981
Linux x86-64	6395038	6395078	6395024	6394997	6394981
Mac OS	NA	NA	6395024	NA	NA
Solaris	6395038	6395078	NA	NA	NA
Solaris 64-bit	6395038	6395078	6395024	6394997	6394981
Solaris AMD64	NA	NA	NA	6394997	6394981
Solaris x86	NA	NA	6395024	6394997	NA
Tru64	6395038	NA	6395024	6394997	NA
VMS	6395038	NA	NA	6394997	NA
VMS Itanium	NA	NA	NA	6394997	NA
Windows 32-bit	6417013	NA	6408393	6397028	6430171
Windows 64-bit	6417014	NA	6408394	6397029	6430173
Windows AMD 64	NA	NA	NA	6397030	6430174

6394981

Hotovo | https://metalink.oracle.com/metalink/plsql... | metalink.oracle.com | Proxy: Nic | 141.146.8.193 | Oracle | Adblock

[D] Auditing Oracle version

- Collect version info from all available sources:
  - 1) Use “opatch lsinventory”
  - 2) Query DB system views “v$version” & co.
  - 3) Ask DB administrator
- Especially info about CPUs and interim patches is not always reliable

[D] Oracle Networking

- **NET8** (historically also SQL*Net) protocol
  - Core of the oracle networking
- **Listener** – server component managing network traffic between Oracle clients – Oracle database
  - All clients' network connections with a database go through Oracle Listener
  - Listener = network service by default listening on the port 1521/tcp of DB server
- Other issues related to Net8 protocol
  - Naming services
  - Advanced authentication (LDAP, …)

[D] Oracle Listener/1

- Listener (1)
  - Key files (by default):
    - configuration $ORAHOME/network/admin/listener.ora
    - logfile $ORAHOME/network/log/listener.log
  - Could be running under more aliases (default alias = LISTENER)
  - UNIX – check “running” listener aliases:

2 aliases running

```
[oracle@orabox ~]$ ps -ef | grep tnslsnr
oracle 5353 /oracle/orahome/bin/tnslsnr LSN1 -inherit
oracle 2898 /oracle/orahome/bin/tnslsnr LSN2 -inherit
```

[D] Oracle Listener/2

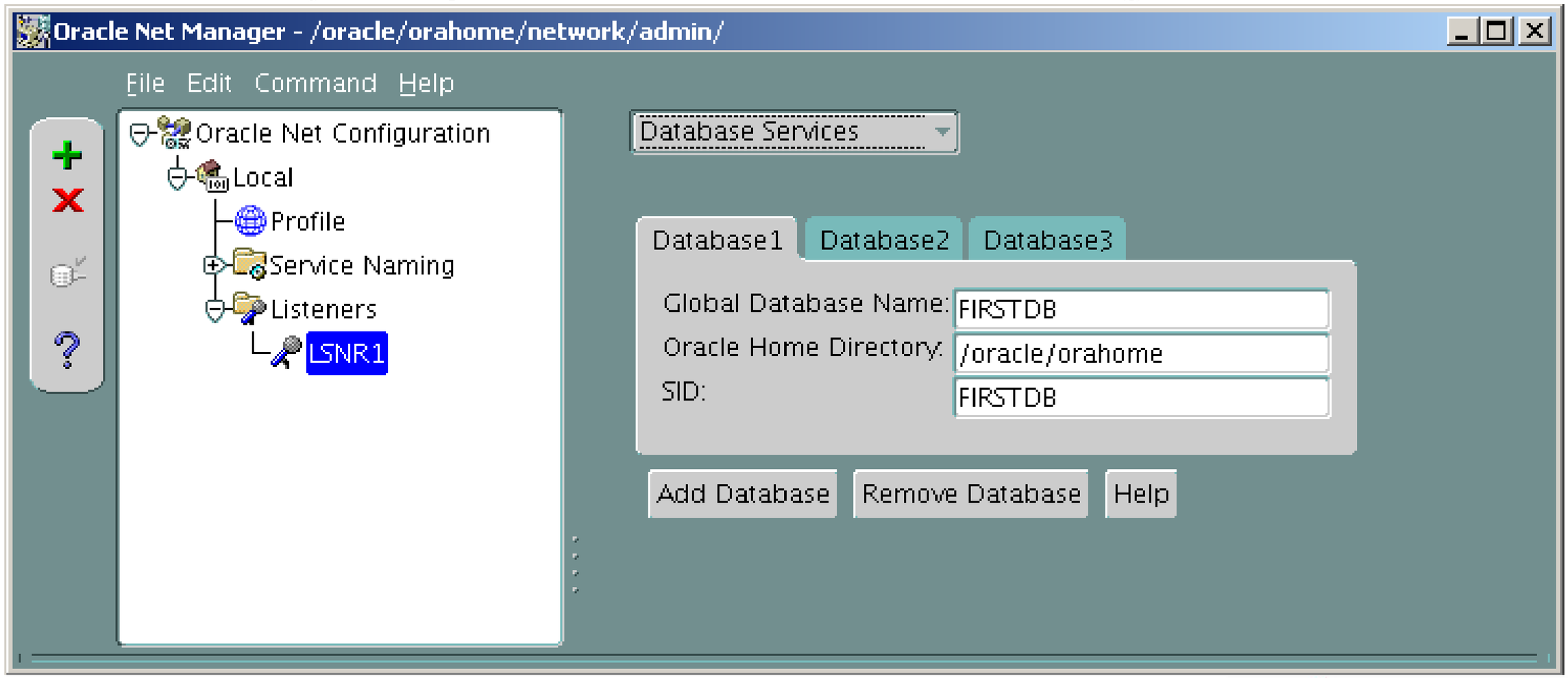
- Listener – getting info (1):
  - “lsnrctl status” and “lsnrctl services” commands

```
[oracle@orabox ~]$ lsnrctl status LSN1

LSNRCTL for Linux: Version 10.2.0.3.0 - Production on 25-JAN-2008 14:25:28
Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost.dom)(PORT=1522)))
STATUS of the LISTENER
------------------------
Alias                     LSN1
Version                   TNSLSNR for Linux: Version 10.2.0.3.0 - Production
Start Date                19-DEC-2007 06:05:13
Uptime                    37 days 8 hr. 20 min. 15 s
Trace Level               off
Security                  ON: Local OS Authenticatio
SNMP                      OFF
Listener Parameter File   /oracle/orahome/network/admin/listener.ora
Listener Log File         /oracle/orahome/network/log/lsnr1.log
...
```

Check config file for more details

[D] Oracle Listener/3

- Listener – getting info (2):
  - Manually inspect Oracle Net Manager



[D] Listener security/1

- Key parameters in “listener.ora”:
  - **PASSWORD_**<listeneralias>
    (password for remote access to listener control functions)
  - **ADMIN_RESTRICTIONS_**<listeneralias>
    (ON = listener remote control denied)
- Recommendations:
  - **9i**: it is necessary either to set PASSWORD or to set ADMIN_RESTRICTION ON
  - **10g**: by default “Local OS Authentication” (setting password not necessary); it is still recommended to set ADMIN_RESTRICTION ON

[D] Listener security/2

- Listener **aliases**
  - It is a “good practice” to avoid default name LISTENER (also avoid aliases identical with DB instance names)
- **TCP port** used for listening
  - Do not use default port 1521/tcp
  - Generally ports within 1520-1530/tcp will be for an attacker the first choice to try

[D] Listener security/3

- Setting **PLSExtProc**
  - **Disable** PLSExtProc if the database does not need to call external libraries (it is “risky by design”)
  - Avoid **ENVS="EXTPROC_DLLS=ANY"**
- Check **file permissions** to
  - Configuration files: **Listener.ora**, **Sqlnet.ora**
  - **Listener.log** (11g: “old-style” & “xml-style” logs)
  - All writable only by owner (oracle)
  - If possible Listener.ora "-rw-r-----" (passwd)

[D] Oracle listener – Sqlnet.ora

- **IP restrictions**
  - Net8 protocol allows to set up IP addresses that are allowed/denied to establish connection with DB server (= with listener e.g. port 1521/tcp)
  - Useful e.g. if DB server communicate just with Application server (not directly with clients)

```
[oracle ~]$ cat $ORACLE_HOME/network/admin/sqlnet.ora
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
tcp.validnode_checking = YES
tcp.invited_nodes = ( 10.0.0.1, 10.0.0.2, 10.0.0.3 )
#tcp.excluded_nodes = ( 10.1.0.1, 10.1.0.2 )
#if you specify invited_nodes, all others are excluded
```

Audit – phase 3
DB instances

[E] Oracle DB instances

- Within **one Oracle installation** you could run **more DB instances**
  - All instances share the RDBMS layer (the same ORACLE_HOME, the same listener, runs under the same OS account …)
  - Each instance has it users, parameters, objects, rights, privileges, …
  - **Each DB instance** = **separate "sub-audit"**
  - The audit scope should focus only on selected database instances

[E] Cross-instances risks

- It is **not good to combine Production and Test** DB instances on one Oracle Server
  - **Test DB** usually **not configured so strictly** as Production DB (therefore easier attack target)
  - Developers often have DBA role in Test DB
  - Anyone (developer/hacker) with DBA in Test instance can easily gain OS access as “oracle” and escalate his/her rights to **all other running DB instances**.

[E] Get basic instance info

- Query **sys.v$instance**

```
SQL> select instance_name,version,status,STARTUP_TIME
Startup,trunc(SYSDATE-(STARTUP_TIME),1)||'days'
Uptime from sys.v$instance;
INSTANCE_NAME VERSION      STATUS   Startup     Uptime
FIRSTDB       10.2.0.3.0   OPEN     19-DEC-07   37.6days
```

- Query **sys.v$database**

```
SQL> select NAME,CREATED,LOG_MODE,PLATFORM_NAME from
sys.v$database;
NAME      CREATED     LOG_MODE       PLATFORM_NAME
FIRSTDB   20-NOV-07   NOARCHIVELOG   Linux IA (32-bit)
```

[E] DB parameters/1

- DB parameters are defined in:
  - 1) **init*.ora** configuration files (text files)
  - 2) or **spfile*.ora** configuration files
    - More progressive mechanism
    - Non-text file (not intended for direct editing)
    - Parameters are modified through “ALTER SYSTEM …”
    - This way allows parameter change in real-time (in most cases without DB restart)
- Location of **init*.ora** and **spfile*.ora**
  - Usually: $ORACLE_HOME/dbs/*.ora
  - Recommended permissions: “-rw-r----- (oracle:oinstall)”

[E] DB parameters/2

- You can find all parameters values in a view **sys.v$parameter**

```
SQL> select name, value from sys.v$parameter;
NAME                                     VALUE
processes                                150
sessions                                 170
timed_statistics                         TRUE
timed_os_statistics                      0
resource_limit                           FALSE
license_max_sessions                     0
license_sessions_warning                 0
cpu_count                                1
sga_max_size                             436207616
...
```

[E] DB parameters/3

- Parameters relevant for security audit
  - Important directories:
    - log_archive_dest_*, core_dump_dest, user_dump_dest, background_dump_dest
    - All of them should have permissions ”drwxr-x---”
  - Parameters: ifile, pfile, spfile
    - Values = file path (or NULL string)
    - Check file permissions, recommended “-rw-r-----”
  - Parameter: os_roles
    - **FALSE** is OK = do not retrieve roles from the operating system

[E] DB parameters/4

- Parameter: os_authent_prefix
  - **NULL string** is OK (recommendations for this parameter are ambiguous)
- Parameter: remote_os_roles
  - **FALSE** is OK
- Parameter: remote_os_authent
  - **FALSE** is OK; keep in mind that TRUE is extremely dangerous
- Parameter: o7_dictionary_accessibility
  - **FALSE** is OK

[E] DB parameters/5

- Parameter: remote_login_passwordfile
  - **NONE** (the more secure) = disables remote DB connections of SYS “as sysdba” (e.g. for DB startup/shutdown)
  - EXCLUSIVE = you need to set this if you want to manage DB remotely e.g. using Oracle Enterprise Manager
  - SHARED = not recommended
- Parameter: remote_listener
  - **NULL string** is OK

[E] DB parameters/6

- Parameter: sql92_security
  - **TRUE** is OK = user trying to execute UPDATE or DELETE with WHERE clause also needs SELECT privileged to the modified table

[E] ArchiveLog mode

- Check whether the database is operating in ArchiveLog mode or not:

```
SQL> select NAME,LOG_MODE from sys.v$database;
NAME      LOG_MODE
FIRSTDB   NOARCHIVELOG
```

- Related DB parameters
  - log_archive_dest_*
  - log_archive_start
- It is more about **availability and business continuity**
- OLTP systems in a production environment should always have ArchiveLog enabled

[E] Data files

- **Datafiles** = primary storage for DB data, files holding tables, indexes, …

```
SQL> select name DATAFILE from v$datafile;
DATAFILE
/oracle/orabase/oradata/firstdb/system01.dbf
/oracle/orabase/oradata/firstdb/undotbs01.dbf
/oracle/orabase/oradata/firstdb/sysaux01.dbf
/oracle/orabase/oradata/firstdb/users01.dbf
/oracle/orabase/oradata/firstdb/example01.dbf
```

  – Check file permissions, recommended "-rw-------"
  – Check ownership "oracle:oinstall“

[E] Control files

- **Controlfiles** = files containing information about database structure (technically they do not contain actual data)

```
SQL> select name CONTROLFILE from v$controlfile;
CONTROLFILE
/oracle/orabase/oradata/firstdb/control01.ctl
/oracle/orabase/oradata/firstdb/control02.ctl
/oracle/orabase/oradata/firstdb/control03.ctl
```

  – Check file permissions, not writable by “others” (e.g. "-rw-r-----")
  – Check ownership "oracle:oinstall“

[E] Redolog files

- **Redologs** = contain a history of data changes, they include data that may or may not have been written to the Datafiles yet.

```
SQL> select member "RedoLogFile" from v$logfile;
RedoLogFile
/oracle/orabase/oradata/firstdb/redo03.log
/oracle/orabase/oradata/firstdb/redo02.log
/oracle/orabase/oradata/firstdb/redo01.log
```

  – Check file permissions, recommended "-rw-------"
  – Check ownership "oracle:oinstall“

[E] Utl_File_Dir / Directories

- Two similar concepts
  - **UTL_FILE_DIR** – older method, not recommended
  - **Directory objects** – new since 9i, preferred mechanism, provides finer access control
- DB Users with EXECUTE on UTL_FILE can access directories in OS (with effective rights of “oracle”).

```
SQL> select name, value from v$parameter where
lower(name)='utl_file_dir';
(value should be empty - avoid '/', '.' or '*')
SQL> select * from dba_directories;
(returns list of directory objects)
```

  - Review OS filesystem permissions

[E] Directories

- Check DB privs (READ, WRITE) to directory objects (see “dba_tab_privs”)
- Check owners of DB directory objects (owner has full access)

```
SQL> select D.directory_name, D.directory_path, D.owner,
R.grantee, R.privilege priv from dba_directories D left join
(select distinct table_name, grantee, privilege from
dba_tab_privs) R on R.table_name=D.directory_name;
DIRECTORY_NAME  DIRECTORY_PATH       OWNER  GRANTEE            PRIV
DATA_PUMP_DIR  /orahome/rdbms/log/   SYS    IMP_FULL_DATABASE  READ
DATA_PUMP_DIR  /orahome/rdbms/log/   SYS    IMP_FULL_DATABASE  WRITE
DATA_PUMP_DIR  /orahome/rdbms/log/   SYS    EXP_FULL_DATABASE  WRITE
DATA_PUMP_DIR  /orahome/rdbms/log/   SYS    EXP_FULL_DATABASE  READ
```

[E] Auditing/1

- Since 8i Oracle has quite powerful auditing subsystem.
- If properly configured it allows to log all significant actions and operations in a database e.g.
  - failed login attempts
  - table and column changes
  - privilege grants
  - etc.

[E] Auditing/2

- Since 9i Oracle introduced FGA – Fine Grained Auditing
  - More detailed info in audit trail for SELECT statements
    - **Standard audit:** User – Object – Operation
    - **FGA:** add details about what rows & columns were selected, exact SQL statement, …
- In 10g another major improvement
  - New capability to audit DML statements (INSERT, DELETE, UPDATE)

[E] Auditing/3

- Up to 10gR2 auditing is disabled by default (in 11g enabled by default).
- DB parameter **AUDIT_TRAIL**:
  - **None** = Disables auditing
  - **DB** = records go to table SYS.AUD$
  - **OS** = on UNIX → *.aud files in directory defined in audit_file_dest param, in Windows → Eventlog
- General "**best practice" recommends OS**
  - for some situations DB could be more suitable (beware: SYS.AUD$ is in SYSTEM tablespace)

[E] Auditing/4

- Check audit configuration:

```
SQL> select name,value from v$parameter where
lower(name) like 'audit%';
NAME                          VALUE
audit_sys_operations          FALSE
audit_file_dest               /orabase/admin/firstdb/adump
audit_syslog_level
audit_trail                   NONE
```

  - audit_sys_operations – **TRUE** is OK
  - audit_trail – **OS or DB** is OK
  - audit_file_dest – directory with **write** access **only for "oracle"**

[E] Auditing configuration/1

- Types of auditing available in Oracle DB:
  - **Statement Auditing**: auditing of SQL statements (DML, DDL), but not regarding specifically named objects
  - **Privilege auditing**: audits statements that use a system privilege e.g. SELECT ANY TABLE, ...
  - **Schema Object Auditing**: audit EXECUTE, SELECT, UPDATE, …statements on a given DB objects
  - **Fine-Grained Auditing**: enables to monitor data access based on content

[E] Auditing configuration/1

- Review audit configuration:

```
SQL> select * from dba_obj_audit_opts;
(shows configuration of Schema Object Auditing)
SQL> select * from dba_priv_audit_opts;
(shows configuration of Privilege Auditing)
SQL> select * from dba_stmt_audit_opts;
(shows configuration of Statement Auditing)
```

- If audit trail = DB – check these views:
  - dba_audit_trail (standard audit events)
  - dba_fga_audit_trail (FGA events)

[E] Database links/1

- **DB link** = trusted link between 2 databases
- DB link types available in Oracle:
  - **Public** - Anyone within the database can use these (avoid – **not a good practice**).
  - **Private** - Only the users or subprograms linked to the owner of the private link can use these to access a remote database.
  - **Global/Shared** - All of the users and subprograms in any database can access and use these.

[E] Database links/2

- DB links can be created „with” or „without“ fixed username:password
  - Current-user links
  - Connected-user links
  - Fixed-user links (**not a good practice**)
- The worst case from security point of view:
  - **PUBLIC & Fixed-user link – very risky**
- Up to 10gR1 passwords for fixed-user links were kept in plain text in table **sys.link$**

[E] DB Users/1

- Each DB instance keeps **information about its users** in:
  - System table **SYS.USER$**
  - Info available also in a view **SYS.DBA_USERS**
  - Both (user$, dba_users) include also **users password hashes – very sensitive**
  - **11g – no hashes in SYS.DBA_USERS**

[E] DB Users/2

- Information about users

```
SQL> select username,account_status,created,profile
FROM sys.dba_users ORDER BY username;
USERNAME        ACCOUNT_STATUS      CREATED     PROFILE
ANONYMOUS       EXPIRED & LOCKED    30-JUN-05   DEFAULT
BI              EXPIRED & LOCKED    20-NOV-07   DEFAULT
CTXSYS          EXPIRED & LOCKED    30-JUN-05   DEFAULT
TESTUSER        OPEN                20-JAN-08   DEFAULT
SYS             OPEN                30-JUN-05   DEFAULT
SYSMAN          OPEN                30-JUN-05   DEFAULT
SYSTEM          OPEN                30-JUN-05   DEFAULT
TSMSYS          EXPIRED & LOCKED    30-JUN-05   DEFAULT
WMSYS           EXPIRED & LOCKED    30-JUN-05   DEFAULT
```

[E] Default passwords/1

- During the installation Oracle creates many default accounts with default well-known passwords:
  - SYSTEM/MANAGER, SYS/CHANGE_ON_INSTALL, SCOTT/TIGER, DBSNMP/DBSNMP, XDB/XDB, TRACESVR/TRACE etc.
- Since 10gR2
  - Most of them are locked by default
  - Admin has to choose own password for SYS / SYSTEM
- **Best practice** recommendation
  - Lock all unused account
  - Avoid Default or Simple Passwords

[E] Default passwords/2

- Oracle passwords – **10gR2 and earlier**:
  - Case-insensitive, max. length 30
  - DES algorithm for computing password hash
- Oracle passwords – since **11g**:
  - New hashing algorithm (based on SHA1)
  - Case-sensitive passwords
  - Oracle keeps unfortunately both “old” and “new” hashes

[E] Default passwords/3

- Testing default/simple passwords
  - **Off-line** password testing
    - Get hashes from dba_users or sys.user$
    - Run some of the **brute-force tools** (quite fast: approx. 0.5-1.0 millions tests/sec)
    - Or use **simple script** (see demo in the end)
  - **On-line** password testing
    - Guess password by trying to connect (slow)
    - Beware: you can lock tested DB accounts (depends on user profile settings)

[E] Default passwords/4

- Password testing as a part of security audit
  - Has to be very carefully communicated
  - Taking away password hashes is a great risk – **for the auditor as well as for the auditee**

[E] External - OS Authentication

- Oracle DB relies on authentication at OS level (at client)
- Mapping between OS and DB users depends on parameter „os_authent_prefix“
- Two ways of external authentication:
  - **LOCAL** – database believes local OS of DB server (there are some security issues)
  - **REMOTE** – **extremely dangerous**, **never use it**
- Avoid external auth. if not necessary

[E] DB User profiles/1

- Each user has assigned his/her profile
- Profile definition - DBA_PROFILES view
- Profile defines:
  - **Password policy** - password_life_time, failed_login_attempts, password_verify_ function
  - **Other settings**
- Auditing security:
  - Check settings in dba_profiles
  - Inspect password_verify_function (dba_source)
  - Compare to corporate password policy

[E] DB User profiles/2

- Profiles - example (default 10gR2)

```
SQL> select profile,resource_name,limit FROM dba_profiles WHERE
RESOURCE_TYPE='PASSWORD' ORDER BY profile, resource_name;
PROFILE                RESOURCE_NAME              LIMIT
DEFAULT                FAILED_LOGIN_ATTEMPTS      10
DEFAULT                PASSWORD_GRACE_TIME        UNLIMITED
DEFAULT                PASSWORD_LIFE_TIME         UNLIMITED
DEFAULT                PASSWORD_LOCK_TIME         UNLIMITED
DEFAULT                PASSWORD_REUSE_MAX         UNLIMITED
DEFAULT                PASSWORD_REUSE_TIME        UNLIMITED
DEFAULT                PASSWORD_VERIFY_FUNCTION   NULL
MONITORING_PROFILE     FAILED_LOGIN_ATTEMPTS      UNLIMITED
MONITORING_PROFILE     PASSWORD_GRACE_TIME        DEFAULT
MONITORING_PROFILE     PASSWORD_LIFE_TIME         DEFAULT
MONITORING_PROFILE     PASSWORD_LOCK_TIME         DEFAULT
MONITORING_PROFILE     PASSWORD_REUSE_MAX         DEFAULT
MONITORING_PROFILE     PASSWORD_REUSE_TIME        DEFAULT
MONITORING_PROFILE     PASSWORD_VERIFY_FUNCTION   DEFAULT
```

[E] DB User profiles/2

- Other profile settings with impact more on performance than security:
  - SESSIONS_PER_USER
  - CPU_PER_SESSION
  - CPU_PER_CALL
  - IDLE_TIME
  - CONNECT_TIME

[E] DB User roles/1

- **ROLE** = named group of privileges that could be granted to DB Users ot to others DB Roles
- Granted roles – DBA_ROLE_PRIVS:

```
SQL> select * from sys.dba_role_privs order by
granted_role, grantee;
(complete listing of all granted roles)
SQL> select * from sys.dba_role_privs where
granted_role='DBA';
(shows users with granted DBA role)
```

- Look for roles granted **"admin_option=yes"** grantee can grant such a role to other users/roles – **mostly a bad idea**

```
SQL> select * from sys.dba_role_privs where grantee
not in ('DBA','SYS','SYSTEM') and admin_option='YES'
order by grantee, granted_role;
```

[E] DB User roles/2

- Roles could be non-default
  - User has to call „SET ROLE“ to activate it

```
SQL> select * from dba_role_privs where
default_role='NO';
```

- Roles can have password
  - „SET ROLE“ command requires password
  - Used in some application to prevent users from direct DB access (passwd hidden from user)

```
SQL> select * from dba_roles where
password_required='YES';
```

[E] Standard (default) roles

- Highly privileged roles – carefully review:
  - **DBA** – virtually unlimited privileges
  - **SELECT_CATALOG_ROLE**, **EXECUTE_CATALOG_ROLE**, **DELETE_CATALOG_ROLE** – access to sensitive catalog data (dba_* views etc.)
- Other roles
  - Prior 10gR2 not recommended to grant standard roles CONNECT, RESOURCE
  - In 10gR2 privs of these roles reduced

[E] Application roles

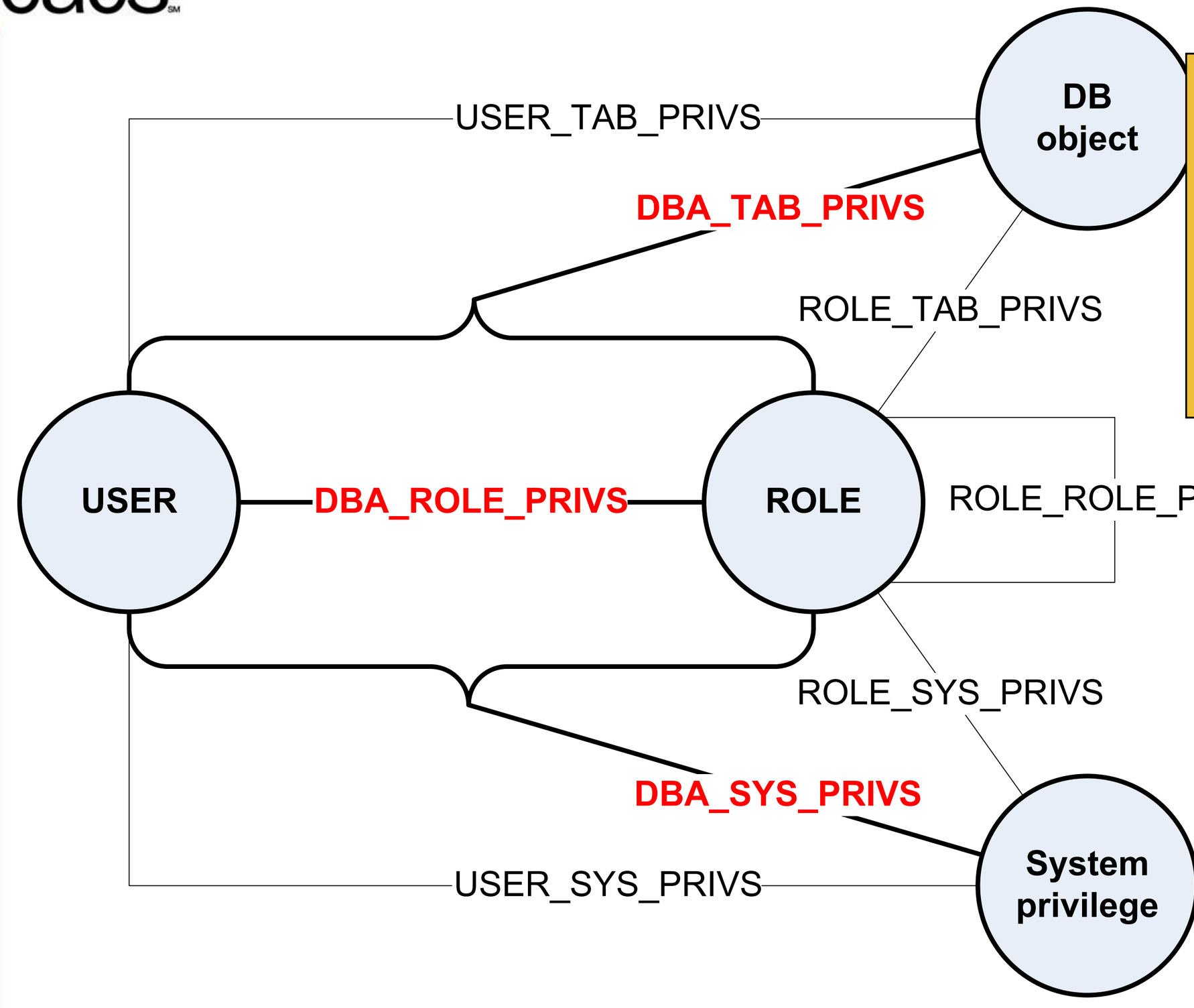
- Auditing application roles require the knowledge about the application
  - You will need at least some interview with App Admin (or App Architect)
- **General "best practice" rule**
  - Avoid granting any privileges (sysprivs, objprivs) directly to the application users
  - Always use application roles

[E] Privileges in DB Oracle

- Database privileges give users the right for specific action, operations.

- In DB Oracle:
  - **Object privileges** – allows user to access to specific object (SELECT, UPDATE, EXECUTE..)
  - **System privileges** – allows for example connect to DB, create DB objects etc.

[E] Roles and privileges



Auditor is interested in:

- dba_tab_privs
- dba_sys_privs
- dba_role_privs

[E] System privileges/1

- System privileges could be granted to users or roles. See – DBA_SYS_PRIVS.
- Some of the **powerful** (thus dangerous) **sysprivs**:
  - ALTER SYSTEM / DATABASE
  - EXEMPT ACCESS POLICY / IDENTITY POLICY
  - GRANT ANY PRIVILEGE / ROLE / OBJECT PRIVILEGE
  - BECOME USER
  - CREATE (ANY) LIBRARY
  - SELECT ANY DICTIONARY
  - EXECUTE ANY PROCEDURE
  - SELECT ANY TABLE etc.

[E] System privileges/2

- **Granted sysprivs** %ANY%, ALTER%, GRANT% ADMINISTER% and EXEMPT% should be **carefully reviewed**

```
SQL> select * FROM dba_sys_privs WHERE (privilege like '%ANY%' or
privilege like 'ALTER%' or privilege like 'GRANT%' or privilege
like 'ADMINISTER%' or privilege like 'EXEMPT%') and grantee not in
('SYS','DBA') ORDER BY privilege,grantee;
GRANTEE                PRIVILEGE                        ADM
EXFSYS                 ADMINISTER DATABASE TRIGGER      NO
IMP_FULL_DATABASE      ADMINISTER DATABASE TRIGGER      NO
EXP_FULL_DATABASE      ADMINISTER RESOURCE MANAGER      NO
IMP_FULL_DATABASE      ADMINISTER RESOURCE MANAGER      NO
OEM_ADVISOR            ADMINISTER SQL TUNING SET        NO
OLAP_DBA               ALTER ANY DIMENSION              NO
IMP_FULL_DATABASE      ALTER ANY PROCEDURE              NO
IMP_FULL_DATABASE      ALTER ANY TABLE                  NO
OLAP_DBA               ALTER ANY TABLE                  NO
...
```

[E] System privileges/3

- **General "best practice" rules**
  - Grant sysprivs "WITH ADM OPTION" only when really necessary
  - Do not grant sysprivs directly to users (use roles)
  - Never grant any sysprivs to PUBLIC

```
SQL> select * FROM dba_sys_privs WHERE grantee='PUBLIC'
OR grantee IN (SELECT USERNAME from DBA_USERS) ORDER BY
grantee, privilege;
(shows sysprivs granted directly to DB users or PUBLIC)
SQL> select * from dba_sys_privs where grantee not in
('DBA','SYS','SYSTEM') and admin_option='YES' order by
grantee, privilege;
(shows sysprivs granted WITH ADM - except SYS*,DBA)
```

[E] Object privileges/1

- The most common objprivs:
  - SELECT - Query the data in a table or view.
  - EXECUTE - Run stored procedures and functions.
  - INSERT - Add records to a table or view.
  - UPDATE - Modify the data in a table or view.
  - DELETE - Delete records from a table or view.
  - ALTER - Change the definition of a table.
  - INDEX - Create an index on a table.
  - READ - Allow the user to read from a directory object.
  - WRITE - Allow the user to write to a directory object.
  - REFERENCE - Create a reference to a table.

[E] Object privileges/2

- Granted objprivs – see DBA_TAB_PRIVS
- To get the basic summary:

```
SQL> SELECT grantee, privilege, count(*) privcount FROM
sys.dba_tab_privs GROUP BY grantee, privilege ORDER BY grantee;
GRANTEE                        PRIVILEGE      PRIVCOUNT
AQ_ADMINISTRATOR_ROLE          EXECUTE        8
AQ_ADMINISTRATOR_ROLE          SELECT         13
AQ_USER_ROLE                   EXECUTE        4
BI                             SELECT         23
CTXAPP                         EXECUTE        5
CTXAPP                         INSERT         4
...
```

- For precise objpriv audit you will need details about the application using audited database

[E] Object privileges/3

- **General "best practice" rules**
  - Do not grant PUBLIC any objprivs to any application objects
  - Grant objprivs through roles not directly to users
  - If possible avoid granting objprivs as "GRANTABLE"

```
SQL> select grantee, privilege, grantable, count(*)
from DBA_TAB_PRIVS where GRANTABLE='YES' AND GRANTEE
not in ('PUBLIC','SYS','SYSTEM') group BY grantee,
privilege, grantable;
(objprivs granted as GRANTABLE; except SYS*,DBA,PUBLIC)
```

[E] Privileges and nested roles/1

- Situation:
  - Nested roles
    - userX → role1
    - role1 → role2
    - role2 → ...
    - ... → roleY
  - If “roleY has PrivP“ then “userX has also PrivP“
- Nested roles (if recursion is too deep) can complicate DB privilege audit.

[E] Privileges and nested roles/1

– Dealing with nested roles

```
SQL> select r1.grantee, r1.granted_role from dba_role_privs r1;
(shows directly granted roles)

SQL> select r1.grantee, r2.granted_role from dba_role_privs r1,
dba_role_privs r2 where r2.grantee = r1.granted_role;
(roles through 1 recursion)

SQL> select r1.grantee, r3.granted_role from dba_role_privs r1,
dba_role_privs r2, dba_role_privs r3 where r2.grantee =
r1.granted_role and r3.grantee = r2.granted_role;
(roles through 3 recursion)
```

– DB privileges auditing usually involves manual ad-hoc analysis (using CAAT or similar tools)

[E] Object owners

- Object **owners have full access** to their own objects – regardless the granted objprivs
  - It is a good practice to use special technical accounts (normally locked) as object (schema) owners
  - Be suspicious about DB users that are owners of a very few objects (often forgotten tables etc.)

```
SQL> select owner,count(*) count from dba_objects
group by owner;
(show basic statistics of DB object owners)
```

[E] Special objects – external libraries

- External libraries
  - Oracle libraries implemented through external dynamic libraries (binary *.dll or *.so files)
  - Can be **very dangerous** if not properly set up (it is connected with Listener ExtProc)
  - Avoid external libraries at all (if possible)

```
SQL> select owner, library_name,file_spec from
dba_libraries where file_spec is not NULL;
OWNER   LIBRARY_NAME     FILE_SPEC
SYS     DBMS_SUMADV_LIB /oracle/orahome/lib/libqsmashr.so
ORDSYS ORDIMLIBS        /oracle/orahome/lib/libordim10.so
```

[E] Special objects privileges

- **Revoke EXECUTE for PUBLIC** on:
  - DBMS_JOB, DBMS_LOB
  - UTL_* (specially UTL_FILE, UTL_TCP, UTL_HTTP, UTL_SMTP, UTL_INADDR)
- Pay special attention to **sensitive objects**:
  - DBA_USERS (user password hashes)
  - SYS.USER$ (user/role password hashes)
  - SYS.LINK$ (link passwords)
  - SYS.AUD$ (audit trail – if set to DB)

Audit – phase 4
related processes

[F] Processes

- Oracle Security audit is mostly technically oriented; however do not forget at basic processes
  - Backup/restore procedures
  - Segregation of duties (DB admin/OS admin)
  - Patch management
  - Monitoring
  - Change / configuration management
  - App development and deployment
  - Access of external companies (e.g. App support)

[G] Live demo

Live demo

For More Information:

Karel Miko, CISA
DCIT, a.s.
http://www.dcit.cz
e-mail: miko@dcit.cz

Scripts used in this session are available at:
http://www.dcit.cz/files/ECACS-OracleAudit.zip

Thank you!