Logica Guru4Pro
June 2\textsuperscript{nd} 2010

Oracle Security
The Right Approach (IMHO)

By
Pete Finnigan

Updated Friday, 21st May 2010
Why Am I Qualified To Speak

• PeteFinnigan.com Ltd, Est 2003.
• http://www.petefinnigan.com
• First “Oracle security” blog.
• Specialists in researching and securing Oracle databases providing consultancy and training Database scanner software authors and vendors.
• Published many papers, regular speaker (UK, USA, Slovenia, Norway, Iceland, Finland and more).
• Member of the Oak Table Network.
Agenda

• Background “glue”
• The correct approach (IMHO) – The message
• Exploit + reaction (a number of levels)
  – downloadable, easy
  – Realistic theft
  – Sophisticated attack
  – Data analysis
  – User Analysis
• Conclusions
Introduction

• You have me for 2 hours (or longer, I am flexible)
  – The focus is “how easy it is to steal” [some examples] and “how easy it is to not secure properly” [examples]
  – But I want to give you some
  – And; we are going to try a lot of demos!
Overview

• What do I want to achieve this evening
  – 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 secure 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 (NOT ALL OF THEM ON PRODUCTION!)
What Is Oracle Security?

• Securely configuring an existing Oracle database?
• Designing a secure Oracle database system before implementation for new databases?
• Understanding what you have – perform an audit?
• 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!
Traditional Security Approach

• Hardening by checklist – good idea?
• A number of them available
  – SANS Step-by-step guide
  – SANS S.C.O.R.E.
  – CIS benchmark
  – DoD Stig
  – IT Governance book
  – Oracle’s own checklist
Problems With Checklists

• Not many checklists exist for Oracle databases
• Most are from same initial source or are very similar
• Some structure there but not good enough
  – “tip based rather than method based”
• Lists don’t focus on securing the data
• Difficult to implement for a large number of databases
• CIS for instance has 158 pages
Solutions are not Simple

• Time based solution
  – Could spend man years on even a single database
  – Finding solutions for each issue is not as simple as applying what it says in the document

• Clever solutions are needed
  – Technical solutions need to be specified
  – Onion based approach is good
  – Basic hardening in parallel
Examples Of Problems

• Two examples:

1) Check 3.0.2 in CIS states “all files in $ORACLE_HOME/bin directory must have privileges of 0755 or less – fine - but the solution states “chmod 0755 $ORACLE_HOME/bin/*” – is it a good idea?

2) Solutions are not as simple as indicated. For instance fixing a weak password should also include, fix the password, management, hard coded passwords, audit, policy….
Checklists And PII Data

Search of the CIS benchmark - There is some mention of data but it is not focused.
<table>
<thead>
<tr>
<th>The Right Method To Secure</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Start with “the data”</td>
</tr>
<tr>
<td>• Understand “data flow” and “access”</td>
</tr>
<tr>
<td>• Understand the problem of securing “your data”</td>
</tr>
<tr>
<td>• Hardening should be part of the solution <strong>BUT</strong> not <strong>THE</strong> solution</td>
</tr>
<tr>
<td>• Checklists do not mention “your” data</td>
</tr>
</tbody>
</table>
Complex But Simple Solutions Needed

• Overarching solutions are needed
• Remove all types of access from the data
• Ensure only those who should see, can see the data
• Unfortunately it’s not that simple as there are:
  – Many paths to the data
  – Many copies of data
  – Data stored or in transit that is accessible
  – Data copied outside of the database
Understand Architecture

Identify each type of person and a sample account for each
Data Access Models

- Privileges
- View
- API

Other Access to Data
- OS files, SQL Text, Redo, Archive
- Flashback, backups, datafiles, etc.

Privileges

View

API

Data Table (Copy)

Sweeping Privileges
Data Access Is Not “Flat”

- Data model is not flat – remove the “blinkers”
- Access rights are also not flat
- Data is often replicated
  - In other tables – in interfaces – flexfields …
  - Indexes
  - Shared memory
  - Data files
  - Operating system
  - Many more…
How / Who

• The data must be identified (found?)
• The access paths must be found
• The “people” – real people identified
• Map to these to database user accounts
• Assess who can access data and how
• Only now can we hope to secure data
Database Security Focus

• If you are a hacker what is the focus?
  – Lots of bugs to study
  – Lots of exploits for download
  – Lots of info on hacking Oracle to use

• If you are a defender what is the focus?
  – In my experience not much has been done
  – People rely on Oracle doing the work BUT they don’t!
More for the Attacker

• Lots of databases have these issues:
  – Weak and guessable passwords
  – No password management (fixed from 11gR1 and 10.2.0.2)
  – Weak controls on the data and functions
  – No audit in the database (fixed from 11gR1 and 10.2.0.2)
  – Weak privilege design for users, solutions (batch, feeds etc) and DBA’s
  – Usually no processes to manage any breach or potential breach
Simple Exploit

- Escalation of Privileges
- 5 minutes demonstration

Live Demo 1
What are the issues?

• For you:
  – Easy to download
  – Easy to run
  – No skill needed
  – Everyone learn about and download
  – Only real solution is patch (for most bugs / exploits)
  – BUT.....
Payloads, Targets

• The focus of researchers is “grant DBA to public”
• This is wrong, the possible payloads are infinite
• The “real” target is
  – Data
  – Job satisfaction
  – Revenge
  – More?
• Factor in IDS evasion
• Factor in downloadable exploits benefit those who “know”...
Stealing Data - Realistic

• 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 - Slide 2

• Hacking an Oracle database to “steal”
• 15 minutes demonstration
Reaction

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

- OK, easy and realistic
- There are still issues, for someone to steal they still need Oracle knowledge and business knowledge
- The issue is that because “WE” (the Oracle customers) do not fix databases we make it easy to steal – the target audience for these “ADVANTAGES” is likely employees – DBA, Power users, Dev....
Data Theft

• Data theft is more likely possible due to:
  – Application abuse
  – Data not in the database
  – Data given to users
  – More....

• Oracle will not fix these issues for you, they are your responsibility!
The Defenders View

- Did our realistic attack leave evidence
- Does the DBA review these evidences?
- Audit trail
- Listener log
- redo
- More...

Live Demo 3
What if the Hacker Was Clever

• If he was clever he may take a number of different approaches
  – Stealth
    • in finding an account
    • Escalate first
    • Check identity
    • Steal the data from somewhere else
Some Thoughts

• A data security solution must be comprehensive
• All copies of the data must be located and protected to the same level
• Theft will always occur taking the easiest approach!
The True Access To The Data

Live Demo 5
## 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.frtsirt.com](http://www.frtsirt.com)
  - CERT – [www.kb.cert.org/vulns](http://www.kb.cert.org/vulns)
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
• There are a number of good checklists to define what to check:
• CIS Benchmark -
  http://www.cisecurity.org/bench_oracle.html
• SANS S.C.O.R.E -
  http://www.sans.org/score/oraclechecklist.php
• Oracle’s own checklist -
• DoD STIG - http://iase.disa.mil/stigs/stig/database-stig-v8r1.zip
Analysis Of Users

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
Analysing Users

Live Demo 7
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

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

In general this should be set to null as it is system wide.
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
Check all directories in the same manner
Assess who can access them and why
Start with the dangerous directories
Access To The Server - 5

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
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]
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 packages.
• Java – find file access permissions
• Locate all packages that use the privileges, check dependencies, access to those packages…
Access To The Server - 9

Check who can create or drop directories

Check who can change utl_file_dir

Check who could grant these privileges

Check who can change, create.. Procedures and libraries
• 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
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?
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
• Oracle security is complex though because we must consider “where” the “data” is and “who” can access it and “how”
• Often there are “layers” and “duplication”
• Careful detailed work is often needed
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