Global Information Assurance Certification Paper

CONDUCTING A SECURITY AUDIT OF AN ORACLE DATABASE

By Egil Andresen
Practical assignment GSEC Ver. 1.3
8 March 2002

0. Summary

This paper has been written from the perspective of an external, independent auditor with the task of conducting a security audit on a system based around an Oracle database. The methodology presented in the Federal Information System Controls Audit Manual is described as a foundation for conducting the audit. Specific security issues related to Oracle databases are discussed based on the methodology.

The focus of the paper is on auditing access controls to Oracle databases. What should the auditor evaluate and test to enable him to give an informed opinion about the security of an information system based on an Oracle database? A number of issues that the auditor should evaluate are discussed in the paper, with indications of how these issues should be dealt with by the entity being audited.

1. Introduction

Databases are a key element in most business-related information systems. The transactions are recorded in the database. The accounting system of any company relies on a database. But how do we know that the information stored in the database and the reports made from these, can be relied upon? We have to evaluate the internal controls that the system is based upon. The topic of this paper is the conducting of a security audit of an Oracle database. The paper is written from the perspective of an external auditor independent from the entity being audited.

The Federal Information System Controls Audit Manual (FISCAM), provided by The United States General Accounting Office (GAO), describes a methodology that can be used when evaluating internal controls related to computer-based information systems. FISCAM is used as the basis for the themes covered in this paper. While FISCAM provides a more general overview of the methodology, I have here tried to relate this basis to issues specific to an environment based around an Oracle database.

When conducting a security audit of a system, there might be a lot of ground to cover. This paper cannot cover every subject in any detail. I have chosen to give a short introduction to the audit process itself. The focus is on access controls in relation to an Oracle system. I have chosen not to make a checklist for this category of controls, but rather to discuss issues that the auditor should look into. This approach gives more room to explain why certain issues should be evaluated and consequently gives a possibility to gain more knowledge. Other areas of internal controls that should be

evaluated by the auditor are mentioned in order to gain an overall understanding, but are not covered in any depth.

2. The phases of an audit

2.1 The planning phase

2.1.1. Gathering information

To be able to evaluate the security of a system the auditor first of all has to gain adequate knowledge of the system being evaluated. Some information needed in this connection could be e.g.:

- The purpose of the system.

  Why is this system run? Is the processes it support of vital importance to the company? The importance of the system or parts of it is essential input to a risk assessment.

- The structure of the system.

  Network diagrams specifying hardware and software might be a useful starting point. As we are concentrating on databases, it is also important to start gathering information about this subject. What kind of database is used? What operating system runs on the servers where the database system software resides? What applications access data in the database? Are there several application-specific databases or do several applications access the same database? How do the applications interface with the database?

- Security policies and plans

  These will be the foundation for the internal controls system in the entity, and also a possible starting point for the audit.

- Information about the organization and it's procedures

  Organization charts could be a starting point here. To dig a little further procedural manuals or similar descriptions could be useful. This is important as towards an initial evaluation of whether there exist procedures for vital activities security-wise (e.g. procedures for administration of user-id's, procedures for backup and recovery).

2. 1. 2. Risk assessment

After gaining an understanding of the entity's operations, the auditor must assess the risks associated with this entity, as well as the audit of it. The risk assessment will determine the extent of the audit of this entity, as well as which areas the audit should focus on.

Above we have taken for granted that a database system is the area of focus for the audit. In real life this would be determined from a risk assessment. If we chose to concentrate on a database system, the risk assessment might also help guide us as to how we approach this audit. One approach might be to start the audit by focusing on the applications used in connection with the database. Another approach could be to start by focusing on the database itself and evaluate controls independent from each individual application.

2.2. Evaluating and testing controls

During this phase the auditor should obtain detailed information on policies and procedures of importance to the internal controls in the areas being evaluated. The auditor needs to perform tests of these control activities to determine if they are operating effectively.

FISCAM identifies six categories of controls that the auditor should consider. These are briefly:

- Entitywide security program

  FISCAM emphasizes the importance of an entitywide program for security planning and management as the foundation of a security structure for a company. The program should establish the framework and a continuing cycle for risk assessment, the development and implementation of effective security procedures, and the monitoring of the effectiveness of these procedures.

- Access controls

  These controls limit or detect access to computer resources. As the focus of this paper, the audit of this category of controls is described in more detail below.

- Application software development and change controls

  An entity should have procedures and policies that prevent unauthorized programs, or modifications to an existing program, from being implemented.

- System software controls

  The auditor need to evaluate controls that limit and monitor access to the powerful programs and sensitive files that control the computer hardware and secure applications supported by the system.

- Segregation of duties

  FISCAM define these controls as the policies, procedures, and an organizational structure established so that one individual cannot control key aspects of computer-related operations and thereby conduct unauthorized actions or gain unauthorized access to assets or records.

- Service continuity

  These controls help to ensure that when unexpected events occur, critical operations continue without interruption or are promptly resumed and critical and sensitive data are protected.

2.3. Reporting phase

During this phase the auditor will draw his conclusions on the controls that have been evaluated, and the effects on the entity as a whole. The auditor would make a report to present his conclusion. The manner and recipient of the report would be dependent on the situation and the auditor's role in relation to the auditee.

3. Evaluating and testing access controls in an Oracle-based system

3.1. Introduction

The remainder of this paper will focus on evaluating and testing access controls in an Oracle-based system.

It is important to have in mind why access controls are established. Generally these controls are established to help the entity in trying to reach the following objectives:

- Confidentiality: controls should provide reasonable assurance that unauthorized users cannot access information held in the database
- Integrity: controls should help provide reasonable assurance that information are protected against unauthorized modification or impairment
- Availability: controls should provide reasonable assurance that information are not lost as a result of unauthorized access

In FISCAM this category of controls have been broken down into four critical elements. Below each one of these are commented on and related to risks associated to databases, specifically Oracle databases.

3.2. Classify information resources according to their criticality and sensitivity

What kind of information is stored in the databases of the entity being audited? The entity skould have knowledge of the information being stored, and should have made a classification of the importance of the different elements of information in the databases. The importance would here relate to the potensial damage if the data was not available, incorrectly changed, or if the data was inadvertely made public in breech of confidentiality. The classification of information should be done by the data owner, e.g. the department which update the information in the database, as they should have the best knowledge of the sensitivity involved.

The classification will be the most important input as to how much weight is put on access controls, and as a consequence which access controls are estalished and the granularity of the controls. In Oracle it is possible to increase security by using certain options. For instance the option "Advanced Security" could be used if it is deemed necessary to increase security in relation to authentication with the use of digital signatures and encryption. The option "Label security" could be used to increase granularity of database controls, even to the extent of actually being able to prevent the dba from doing certain tasks. Label security was first introduced in a later version of Oracle 8. As far as I know neither of the mentioned options are common in use as yet.

Classification of information will also be essential input as the basis for determining how roles, privileges and profiles should be designed for the database being evaluated. These terms and how they are technically implemented are explained in more detail below, but it is very important to remember that the foundation for the implementation are a non-technical classification of information.

3.3. Maintain a list of authorized users and their authorization level

Authorized users

The questions here are quite simple: who should have access to our databases and how wide should their access be ? Unfortunately experience has shown that these questions are not as easy as they seem, both to answer and to put into practice. A company should have a policy as to who should have access at the different levels. With this as a basis, a company need to develop procedures for establishing, modifying and retracting user access rights. There should also be developed procedures for reviewing access rights periodically.

A security audit would include an evaluation of these procedures for maintaining login accounts and access rights. It would also be appropiate to check lists of user accounts/rights against employee lists, etc. In any entity of a reasonable size I would presume the auditor would find some employees with greater access rights than stipulated in the policy, or some active user accounts for employees no longer in the service of the company under scrutiny. The reasons for these breaches of security should be investigated in order to find possible weaknesses in the procedures the entity is following.

Generic user-id's

Another common finding is an uncontrolled used of generic database id's. It is essential that even the IT department implement personal user accounts to the database with passwords known only to the owner of the user account. The use of generic user id's should be minimized as they jeopardize the safety of the user accounts with an increasing risk that passwords might be exposed, as well as making it virtually impossible to trace changes in the database.

Default user accounts

When an Oracle database is installed, a number of default user accounts will also be established. The actual number of user accounts which is established at installation will vary depending on the operating system of the server on which the database system runs, and also which options are chosen at installation. In "Exploiting and protecting Oracle" (Finnigan, p. 12-13) 52 default user accounts have been identified when the database is based on a system running Linux, while 57 user accounts has been identified when Windows NT is used. David Litchfield includes some accounts created by ordinary third-party products in an internet environment, and presents a list of a massive 160 default user accounts (Litchfield, p. 24-28).

The most important default user accounts would be the ones with dba-privileges or similar. These include user accounts "SYSTEM" (password manager), "SYS" (password CHANGE_ON_INSTALL) and CTXSYS (password CTXSYS). The account "MDSYS", default at all Windows NT installations, has "All privileges with admin", which is as near as you can get to dba.

When conducting a security audit it is important to verify that all default user accounts that don't need to be open, actually are locked. Of the accounts mentioned in the last paragraph, system and sys would normally need to be open, but ctxsys and mdsys can usually be locked. It is equally important to ensure that the password has been changed for all default user accounts that remain open. The passwords associated with all default user accounts are listed in the documents produced by Pete Finnigan and David Litchfield as referenced above. Of course the new passwords that are chosen should have adequate strength (ref. section about passwords below).

3.4. Establish physical and logical controls to prevent or detect unauthorized access

FISCAM advocates that an entity should have a cost-effective process for protecting data files, application programs, and hardware through a combination of physical and logical security controls. In this paper the focus will be on logical controls related to an Oracle database. When performing a security audit, physical security controls should of course also be considered, as well as logical controls related to the operating system or applications.

Roles, privileges and profiles

To understand and evaluate access controls it is necessary to be familiar with the terms roles, privileges and profiles. Below is a short explanation of each of these terms:

- Privileges:

  There are two categories of privileges in an Oracle database: system privileges and object privileges. System privileges give rights to log on to the system and to create or manipulate objects. Object privileges grant a user the right to access and possibly manipulate data within an object, or the right to execute a stored procedure.

- Roles:

  A role is simply a collection of privileges. Roles are assigned to users as a means of granting them the necessary privileges to perform their duties. Privileges can be granted directly to individual users, but this will very easily become very difficult to administer. Using roles will make administration easier. Finding privileges assigned to individual users rather than roles would be an indication of a higher risk for some users being given privileges they were not supposed to have.

- Profiles:

  Two different types of profiles are included in Oracle. Product profiles limit user access to certain Oracle commands or products, while system resource profiles puts limits on system use and provides password management functions.

The auditor should evaluate how roles, privileges and profiles are used by the entity being evaluated. Here are some of the issues the auditor might want to evaluate and test:

- The principle of least privilege

  The auditor should evaluate if the roles, privileges and profiles assigned to different users are in accordance with the principle of least privilege. A user should only be given those privileges that are actually required to efficiently and succinctly perform his or her job.

- System privileges

  Even after creating a user account, the user does not have the necessary privileges to log on to the database. The user must be granted the privilege CREATE SESSION in order to log on. So every user need this privilege, but that might be

the only system privilege needed by an ordinary user. A DBA on the other needs a lot of system privileges to administer the system. In Oracle 8 the standard DBA role had 89 system privileges. The auditor must evaluate if privileges given to each role or individual is in accordance with policies and will help the entity reach their objectives.

- The ADMIN option

  The Grant privilege allows the user receiving the privilege to grant privileges to other users. This powerful privilege should be restricted to the database administrators. The admin option indicates whether a privilege can be passed on with administrative privilege. If the privilege is later revoked from the original grantee, the other user will still retain the privilege passed on to him/her. The auditor should evaluate whether or not the admin option is restricted to users with the highest authority levels (e.g. database administrators).

- User group PUBLIC

  ”PUBLIC” acts as a default role granted to every user in an Oracle database. Any database user can exercise privileges that are granted to the group public. Obviously the privileges granted to this group should be limited to what is absolutely necessary.

- Default roles

  An installation of Oracle will provide a number of default roles. Among these are the dba. Other default roles are “Connect” and “Resource”. The auditor should be aware that the definitions of these roles have changed over time. Using these roles could mean that privileges change when the entity installs a new version of Oracle. It is also important to know that the role “Connect” are more permissive than what the name indicates. This is not a suitable role for an end user.

- Views

  Views are definitions that describe how data is to be retrieved from one or more tables. They can be used to limit a user’s access to sensitive data to match exactly what is needed. The user doesn’t need to have access to the table where the data is stored in order to have access to a view presenting information from this table. The auditor needs to evaluate the use of views and the access this gives to users.

- Stored procedures

  Programs written in PL/SQL can be stored in compiled form in the database. In general stored procedures execute on behalf of the user, but with the privileges of the creator. For this reason it is important to be aware who is the owner or creator of a stored procedure.

User privileges to these should be limited to select and execute as required by user job descriptions.

- Product profiles

  Product profiles are provided when running the script pupbld.sql, which creates two tables: Product_profile and user_profile. These tables can be used to restrict access to a number of SQL/SQL*Plus commands. It is possible to block access to commands such as “set role”, which enables a user to change his current role, or to “host”, which gives access to a operating system prompt, as well as commands such as connect, create, grant, delete, etc. End users should not have access to the database with the help of utilities like SQL*Plus. This is one way of limiting such access.

- Passwords

  System resource profiles include a number of security-related parameters, in particular related to the use of passwords. While features of this kind has been common in operating systems for a long time, they have been relative rarely seen in databases. With Oracle these features were first seen in Oracle 8. It is possible to set restrictions on password composition, complexity, aging, expiration and history. In addition it is also possible to set rules for locking accounts after a number of failed login attempts, a maximum number of concurrent sessions for a user, and rules to disconnect idle users. Different profiles can be defined. Consequently users can be assigned to different profiles, which needs to be taken into consideration when evaluating security at a site. While the specific values of the parameters here will depend on the security policy, the features given by using system resource profiles should be used at all sites with production systems.

- SYS.USER$

  Oracle stores usernames and hashed passwords in the table SYS.USER$. According to Pete Finnigan (Finnigan, p. 21-22) the 16 character hashes that appear as the passwords in the table are created from both username and actual password by the use of an algorithm used in several versions of Oracle. PenTest Limited has developed a password cracker tool for Oracle, but the tool is not available from their internet site as yet. It can however be assumed that gaining access to the hashed passwords, would enable an attacker to find out at least some of the original passwords. This illustrates the importance of protecting the table SYS.USER$ and the data dictionary in general, see the section about securing the data dictionary. Password hashes can also be found in backup files, ref. the comments to these under the heading “Securing resources at the OS level”.

- Protecting the data dictionary

  The data dictionary contains the metadata that describes all objects in the database. Here is the definitions of users, tables, views, triggers, etc. These are

contained in the sys-schema. Access to the metadata would give someone the opportunity to understand, change or destroy the database. Access to the data dictionary should be limited to database administrators. One important aspect to remember is the privileges that contain the ANY qualifier e.g. create any table, drop any table or grant any role. A user with such a privilege can perform the task specified in the privilege under any schema or userid, including the sys-schema. To protect the data dictionary Oracle provides a configuration parameter (`O7_DICTIONARY_ ACCESSIBILITY)` in the init<sid>.ora-file that can limit access to the data dictionary to users with dba-authority. Setting the value of this parameter to FALSE, would hinder the danger in connection with privileges with the ANY qualifier. When performing a security audit, the value of this parameter should be checked, as well as a general evaluation of whether access to the data dictionary has been limited to those users who need this to perform their duties.

Authentication

Oracle provide for various methods of authentication. The most usual method would probably be Oracle-based authentication based on username and password. It is also possible to use host-based authentication, which is based on operating system user accounts being passed on to Oracle. The DBMS would trust the operating system and no password would have to be given. These user accounts generally have the prefix OPS$. If an attacker could break into the system at operating system level and some accounts were set up based on host-based authentication, this would make it easier to gain access to the database. If a session started by a user account with host-based authentication is left idle, no password would be required for any bypasser to access the database. The auditor should evaluate if any user accounts utilizing host-based authentication exists, and if so make sure procedures are in place to secure these. Relevant parameters in initializations files should also be checked by the auditor.

The Oracle Advanced Security option gives possibilities to use other authentication methods.

Securing resources at the OS level

This paper concentrates on access controls within the DBMS, but it is important not to forget to protect the database server(s) properly from unauthorized access also at the operating system level. Betty Dorsey (Dorsey, p.17) puts insecure environment – the operating system and the network – at the top of her list of most common findings when evaluating security at sites using Oracle.

Hardening the operating system is beyond the subject of this paper, but protection of Oracle's resources at the operating system level is something that should be mentioned. Files which need to be protected include executable files, tablespace datafiles, control files, log files, export files, trace files, initialization files and configuration files. Users should not have any operating system privileges on these files beyond the privileges Oracle instructs to set in the installation guide. Users in general need no access to these files.

What could an unauthorized user learn from gaining read access to Redo log files, as they are not stored in a human readable form ? They might not be instantly readable, but it is possible to dump the logs to a trace file. Oracle provides a tool to read trace files. Given access to these files, thus means that it is possible to extract a considerable amount of information without having access to the database itself.

Export files can naturally contain a wealth of information. A full export will contain all data, including the system tables and the hashed passwords. Access to copy export files will give the possibility to recreate the database at another location. Just reading an export file could give an unauthorized user the possibility to see the password hashes stored in the database, and with that as a basis it could be possible to crack the passwords.

Scripts with usernames/passwords.

While scripts with hard-coded usernames and passwords obviously should not exist on a system, several sources report that it is not uncommon to find such scripts. Hard-coded usernames and passwords obviously makes it easier to find a valid combination of username and password, and as a consequence can enable unauthorized access. A part of a security audit could be to search the server for scripts containing hard-coded usernames and passwords.

SQL tools

End users would in most circumstances not have any need to access the database with the help of utilities such as SQL*Plus. The auditor should evaluate if the entity have procedures that try to prevent users access to such tools. If someone needed a limited access with the help of such a tool, it would be possible to limit access to commands as described under product profiles above.

Installed functionality needed ?

A standard installation of Oracle will include some functionality that might not be needed in the entity being evaluated. This functionality might increase the risk for unauthorized access. The auditor should evaluate whether the entity know what functionality is needed and check if only this functionality is installed.

An example of functionality that increase the risk of unauthorized access is the Oracle PL/SQL package for External Procedures. A security vulnarability in this package could enable a malicious user to gain unauthorized administrative access to the machine hosting the Oracle database server. The easiest way of avoiding this vulnerability would be to remove the functionality from the server. If the functionality is needed, other work-arounds should be used.

Test and development databases

Test and development databases are often not secured as well as production databases as this is considered of less importance. However they can include information that should not be disclosed. Test databases can as an example often include copies of real live data. An attacker might not need to access the production database if a test database contains almost the same data. Another point to consider is access rights to test and development databases. In a development environment it is likely that more persons are given higher privileges than what they would have in a production environment. Achieving access to a development database with high privileges can let an attacker gain valuable knowledge about a production database. It is also quite possible that usernames, passwords and privileges have been carried over from the development to the production database.

During a security audit it should be considered what information that could be drawn from test and development databases, and how well they are secured.

Links between databases

It is not uncommon that applications are created that access more than one database. This requires a link between the databases. The links can also be utilized by others to gain access to databases which one initially might not have access to. When conducting a security audit it should be considered whether the database studied has any links to other bases, and, if so, could access controls in these bases put the the base we are evaluating at risk.

3.5. Monitor access, investigate apparent security violations, and take appropriate remedial action

So far the focus have been on preventative controls. Here the focus is on detecting controls and corrective actions. Security incidents will happen however much we try to avoid them, but procedures should be established to detect these incidents in a timely manner and there should be procedures for appropiate reactions.

Auditing in Oracle is the monitoring and recording of activities within the database. Standard auditing functions in Oracle provides functions for auditing almost any action within the database (viewing, modifying information, executing programs, etc). However standard auditing is only supported on a table level, not at a row level. Auditing at this level can however be achieved through the use of triggers. The results from the auditing is stored in the sys.aud$ table.

It is not possible to say what should be audited, as this will depend on circumstances specific to each site. It is however important to appraise whether the entity being evaluated have been through a process where they have carefully considered what they need to audit. To evaluate whether the company have procedures for going through the logs, reporting on them, and take the relevant action if so is necessary, is equally important.

Auditing can affect performance to a certain degree. There are many stories of entities starting out with an ambitious degree of auditing. When the logs fill up the disks and performance are affected, the administrator then turns off the auditing functions altogether. It is important to get the balance right; auditing what is needed to achieve your objectives without affecting performance too much.

There are three kinds of auditing:

- Statement auditing

  Statement level audits are based on the type of SQL statement presented. One possibility is to audit successful and unsuccesful attempts at connecting to the database.

- Privilege auditing

  Privilege auditing is based on auditing actions connected to certain privileges. Actions connected to a dba-privilege could for instance be audited.

- Object auditing

  It is possible to audit all actions taken against specific objects. Key tables that normally aren't changed very often could for instance be audited for changes.

4. Conclusions

In this paper I have tried to show how the audit methodology presented in "Federal Information System Controls Audit Manual" (FISCAM) could be used as a foundation when auditing systems based around Oracle databases. The concepts in FISCAM have been linked to specific issues related to Oracle databases. An auditor should evaluate a wide range of controls when conducting an audit of a database system, and it has only been possible to cover some of them in this paper. The paper should illustrate that Oracle databases, while not unbreakable as claimed in Oracles advertising, contain a wide array of possibilities to implement a secure system. How secure the system will be in practice, does however as always depend on how the system is implemented and specific procedures established by the entity.

References

1) Finnigan, Pete. "Exploiting and Protecting Oracle". Oracle Security White paper series, PenTest Limited Version 1.5.
   URL: http://www.pentest-limited.com/oracle-security.pdf (5 March 2002)

2) Dorsey, Betty J. "How to Audit Oracle Databases". Lecture notes from the European Conference on The Control and Audit of Information Technology, 10 October 2001.

3) Litchfield, David. “Hackproofing Oracle Application Server (A guide to securing Oracle 9)”. NGSSoftware Insight Security Research Publication, 10 January 2002. URL: http://www.nextgenss.com/papers/hpoas.pdf (5 March 2002)

4) Theriault, Marlene and Heney, William. Oracle Security. Sebastopol, CA: O’Reilly & Associates, Inc, 1998.

5) “Federal Information System Controls Audit Manual”, Volume 1: Financial Statement Audits. United States General Accounting Office, Accounting and Information management Division, June 2001. URL: http://www.gao.gov/special.pubs/ai12.19.6.pdf (5 March 2002)

6) Sinha, Rajiv. ”A Security Checklist for Oracle*9i”*. An Oracle White paper, March 2001. URL*:* http://otn.oracle.com/deploy/security/oracle9i/pdf/9i_checklist.pdf (5 March 2002)

7) Plusnina, Svetlana. ”Oracle Database Audit Program”. 01.06.2000. URL: http://www.auditnet.org/docs/Oracle%20Database%20Audit%20Program.doc (5 March 2002)

8) ”Database Security in Oracle 8i”. An Oracle Technical White paper, November 1999. URL: http://otn.oracle.com/deploy/security/oracle8i/pdf/dbswp86.pdf (5 March 2002)

9) Loney, Kevin. ”Protecting Your Database”. Oracle Magazine, May 2000. URL: www.oracle.com/oramag/oracle/00-May/o30sec.html (5 March 2002)

10) Smith, Howard: ”Hack Proofing Oracle”. Oracle Corporation UK Ltd; URL: http://otn.oracle.com/deploy/security/pdf/oow00/orahack.pdf (5 March 2002)

11) ”Multiple Oracle Vulnerabilities”. Counterpane Security Alerts, 6 February 2002. URL: http://www.counterpane.com/alert-oracle.html (5 March 2002)