A Security Checklist for Oracle9*i*

An Oracle white paper

Information security and privacy and protection of corporate assets and data are of pivotal importance in any business. Oracle9*i* comprehensively addresses the need for information security by offering cutting-edge security features such as deep data protection, auditing, scaleable security, secure hosting and data exchange.

The Oracle9*i* database server leads the industry in security. However, in order to fully maximize the security features offered by Oracle9*i* in any business environment, it is imperative that Oracle9*i* itself is well-protected. Furthermore, proper use of its security features and adherence to basic security practices will help protect against database-related threats and attacks and provide a much more secure operating environment for the Oracle9*i* database.

This document provides guidance on configuring Oracle9*i* in a secure manner by adhering to and recommending industry-standard “best security practices” for operational database deployments.

Details on specific database-related tasks and actions can be found in the Oracle9*i* documentation set.

1. INSTALL ONLY WHAT IS REQUIRED

The Oracle9*i* CD pack contains a host of options and products in addition to the database server. Install additional products and options only as necessary. Or, following a typical installation (if avoiding a custom installation), de-install options and products that are not necessary. There is no need to maintain the additional products and options if they are not being used. They can always be properly and easily re-installed as required.

2. LOCK AND EXPIRE DEFAULT USER ACCOUNTS

Oracle9*i* installs with a number of default (preset) database server user accounts. The Database Client Administration tool (DBCA) automatically locks and expires all default database user accounts **except**

SYS, SYSTEM, SCOTT, DBSNMP, OUTLN and the three JSERV users upon successful installation of the database server.

If a manual (not utilizing DBCA) installation of Oracle9*i* is performed, none of the default database users are locked upon successful installation of the database server. If left open in their default states, these user accounts can be exploited to gain unauthorized access to data or disrupt database operations. `LOCK` and `EXPIRE` all default database user accounts **except** SYS, SYSTEM, SCOTT, DBSNMP, OUTLN and the three JSERV database users after performing any kind of initial installation that does not utilize DBCA. Oracle9*i* provides SQL to perform such operations.

Provided below is the table of database users after a typical Oracle9*i* installation utilizing DBCA.

USERNAME	**ACCOUNT_STATUS**
ADAMS	EXPIRED & LOCKED
AURORA$JIS$UTILITY$	OPEN
AURORA$ORB$UNAUTHENTICATED	OPEN
BLAKE	EXPIRED & LOCKED
CLARK	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	OPEN
HR	EXPIRED & LOCKED
JONES	EXPIRED & LOCKED
LBACSYS	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
OLAPDBA	EXPIRED & LOCKED
OLAPSVR	EXPIRED & LOCKED
OLAPSYS	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OSE$HTTP$ADMIN	OPEN
OUTLN	OPEN

PM	EXPIRED & LOCKED
QS	EXPIRED & LOCKED
QS_ADM	EXPIRED & LOCKED
QS_CB	EXPIRED & LOCKED
QS_CBADM	EXPIRED & LOCKED
QS_CS	EXPIRED & LOCKED
QS_ES	EXPIRED & LOCKED
QS_OS	EXPIRED & LOCKED
QS_WS	EXPIRED & LOCKED
SCOTT	OPEN
SH	EXPIRED & LOCKED
SYS	OPEN
SYSTEM	OPEN

If any default database server user account other the ones left open is required for any reason, a database administrator (DBA) need simply unlock and activate that account with a new, meaningful password.

3. CHANGE DEFAULT USER PASSWORDS

The most trivial method by which Oracle9*i* can be compromised is a default database server user account which still has a default password associated with it *even after installation*.

3a. Change default passwords of administrative users

In Oracle9*i*, SYS installs with a default password of CHANGE_ON_INSTALL and SYSTEM installs with a default password of MANAGER. Change the default passwords associated with users SYS and SYSTEM immediately upon installation of the database server.

3b. Change default passwords of all users

In Oracle9*i*, SCOTT installs with default password TIGER and the three JSERV accounts (AURORA$JIS$UTILITY$, AURORA$ORB$UNAUTHENTICATED and OSE$HTTP$ADMIN each install with randomly-generated passwords. Each of the other accounts install with a default password that is exactly the same as that user account (e.g., user MDSYS installs with password MDSYS).

Change the passwords for SCOTT, DBSNMP, OUTLN and the three JSERV user accounts immediately upon installation as well. If any of the other default user accounts that were locked and expired upon installation need to be activated, assign a new meaningful password to that user account.

Even though Oracle does not explicitly mandate changing the default password for user SCOTT, Oracle nevertheless recommends that this user account also be locked unless it is being actively used.

3c. Enforce password management

Oracle recommends that basic password management rules (such as password length, history, complexity, etc.) as provided by the database be applied to all user passwords and that all users be required to change their passwords periodically.

Oracle also recommends, if possible, utilizing Oracle Advanced Security (an option to the Enterprise Edition of Oracle9*i*) with network authentication services (such as Kerberos), token cards, smart cards or X.509 certificates. These services enable strong authentication of users to provide better protection against unauthorized access to Oracle9*i*.

4. ENABLE DATA DICTIONARY PROTECTION

Oracle recommends that customers implement data dictionary protection to prevent users having the 'ANY' system privileges from using such privileges on the data dictionary.

To enable dictionary protection, set the `init<sid>.ora` (Oracle9*i* control file) configuration parameter, in the following manner:

```
O7_DICTIONARY_ACCESSIBILITY = FALSE
```

By doing so, only those authorized users making DBA-privileged (e.g. `CONNECT / AS SYSDBA`) connections can use the 'ANY' system privilege on the data dictionary. If this parameter is not set to the value recommended above, any user with a `DROP ANY TABLE` (for example) system privilege will be able to maliciously drop parts of the data dictionary.

However, if a user requires view access to the data dictionary, it is permissible to grant that user the `SELECT ANY DICTIONARY` system privilege.

Note that in Oracle9*i*, `O7_DICTIONARY_ACCESSIBILITY = FALSE` by default; in Oracle8*i*, the parameter is set to `TRUE` by default and must specifically be changed to `FALSE` to enable this security feature.

5. PRACTICE PRINCIPLE OF LEAST PRIVILEGE

5a. Grant necessary privileges only

Do not provide database users more privileges than are necessary. In other words, *principle of least privilege* is that a user be given only those privileges that are actually required to efficiently and succinctly perform his or her job.

To implement least privilege, restrict: 1) the number of SYSTEM and OBJECT privileges granted to database users, and 2) the number of SYS-privileged connections to the database as much as possible. For example, there is generally no need to grant `CREATE ANY TABLE` to any non DBA-privileged user.

5b. Revoke unnecessary privileges from PUBLIC

Revoke all unnecessary privileges and roles from the database server 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 PUBLIC. Such privileges include `EXECUTE` on various PL/SQL packages that may permit a minimally privileged user to access and execute packages that he may not directly be permitted to access. The more powerful packages that may potentially be misused include:

- `UTL_SMTP`

  This package permits arbitrary mail messages to be sent from one arbitrary user to another arbitrary user. Granting this package to PUBLIC may permit unauthorized exchange of mail messages.

- `UTL_TCP`

  This package permits outgoing network connections to be established by the database server to any receiving (or waiting) network service. Thus, arbitrary data may be sent between the database server and any waiting network service.

- `UTL_HTTP`

  This package allows the database server to request and retrieve data via HTTP. Granting this package to PUBLIC may permit data to be sent via HTML forms to a malicious web site.

- `UTL_FILE`

  If configured improperly, this package allows text level access to any file on the host operating system. Even when properly configured, this package does not distinguish between its calling applications with the result that one application with access to `UTL_FILE` may write arbitrary data into the same location that is written to by another application.

- DBMS_RANDOM

  This package can be used to encrypt stored data. Generally, most users should not have the privilege to encrypt data since encrypted data may be non-recoverable if the keys are not securely generated, stored, and managed.

These packages are extremely useful to some applications that need them and require proper configuration and usage. These packages may not be suitable for other applications. Thus, unless absolutely necessary, revoke them from PUBLIC.

5c. Restrict permissions on run-time facilities

Do not assign "all permissions" to any database server run-time facility such as the Oracle Java Virtual Machine (OJVM). Grant specific permissions to the explicit document root file paths for such facilities that may execute files and packages outside the database server.

An example of a vulnerable run-time call:

```
call dbms_java.grant_permission('SCOTT',
'SYS:java.io.FilePermission','<<ALL FILES>>','read');
```

An example of a better (more secure) run-time call:

```
call dbms_java.grant_permission('SCOTT',
'SYS:java.io.FilePermission','<<actual directory
path>>','read');
```

6. ENFORCE ACCESS CONTROLS EFFECTIVELY

6a. Authenticate clients properly

Remote authentication is a security feature provided by Oracle9*i* such that if turned on (`TRUE`), it defers authentication of users to the remote client connecting to an Oracle database. Thus, the database implicitly trusts any client to have authenticated itself properly. Note that clients, in general, such as PCs, are not trusted to perform operating system authentication properly and therefore, it is very poor security practice to turn on this feature.

In a more secure configuration where this feature is turned off (`FALSE`), it enforces proper, server-based authentication of clients connecting to an Oracle database.

To restrict remote authentication and thereby defer client trust to the database, set the `init<sid>.ora` (Oracle9*i* control file) database configuration parameter in the following manner:

```
REMOTE_OS_AUTHENT = FALSE
```

6b. Limit the number of operating system users

Limit the number of users with operating system accounts (administrative, root-privileged or minimally privileged)on the Oracle9*i* host (physical machine) to the least number possible.

Oracle also recommends that neither any privileged operating system user nor the Oracle owner be permitted to modify the default file and directory permissions within and on the Oracle9*i* home (installation) directory unless instructed otherwise by Oracle Corporation.

7. RESTRICT NETWORK ACCESS

7a. Utilize a firewall

Keep the database server behind a firewall. Oracle9*i*'s network infrastructure, Oracle Net (formerly known as Net8 and SQL*Net), offers support for a variety of firewalls from various vendors. Supported proxy-enabled firewalls include Network Associates' Gauntlet and Axent's Raptor. Supported packet-filtered firewalls include Cisco's PIX Firewall and supported stateful inspection firewalls (more sophisticated packet-filtered firewalls) include CheckPoint's Firewall-1.

7b. Never poke a hole through a firewall

If Oracle9*i* is behind a firewall, do not, under any circumstances, poke a hole through the firewall; for example, do not leave open Oracle Listener's 1521 port to make a connection to the Internet or vice versa.

Doing so will introduce a number of significant security vulnerabilities including more port openings through the firewall, multi-threaded operating system server issues and revelation of crucial information on database(s) behind the firewall. Furthermore, an Oracle Listener running without an established password may be probed for critical details about the database(s) on which it is listening such as trace and logging information, banner information and database descriptors and service names.

Such a plethora of information and the availability of an ill-configured firewall will provide an attacker ample opportunity to launch malicious attacks on the target database(s).

7c. Prevent unauthorized administration of the Oracle Listener

Always establish a meaningful, well-formed password for the Oracle Listener to prevent remote configuration of the Oracle Listener. Additionally, set the `listener.ora` (Oracle Listener control file) security configuration parameter in the following manner:

```
ADMIN_RESTRICTIONS_listener_name=ON
```

Doing so will also prevent unauthorized administration of the Oracle Listener.

7d. Check network IP addresses

Utilize the Oracle Net "valid node checking" security feature to allow or deny access to Oracle server processes from network clients with specified IP addresses. To use this feature, set the following `protocol.ora` (Oracle Net configuration file) parameters:

```
tcp.validnode_checking = YES

tcp.excluded_nodes = {list of IP addresses}

tcp.invited_nodes = {list of IP addresses}
```

The first parameter turns on the feature whereas the latter two parameters respectively deny or allow specific client IP addresses from making connections to the Oracle Listener (and thereby preventing potential Denial of Service attacks).

7e. Encrypt network traffic

If possible, utilize Oracle Advanced Security to encrypt network traffic between clients, databases and application servers. (Note that Oracle Advanced Security is available only with the Enterprise Edition of the Oracle database).

7f. Harden the operating system

Harden the host operating system by disabling all unnecessary operating system services. Both UNIX and Windows platforms provide a variety of operating system services, most of which are not necessary for most deployments. Such services include FTP, TFTP, TELNET, etc. Be sure to close both the UDP and TCP ports for each service that is being disabled. Disabling one type of port and not the other does not make the operating system more secure.

8. APPLY ALL SECURITY PATCHES AND WORKAROUNDS

Always apply all relevant and current security patches for both the operating system on which Oracle9*i* resides and Oracle9*i* itself, and for all installed Oracle9*i* options and components thereof.

Periodically check the security site on Oracle Technology Network for details on security alerts released by Oracle Corporation.

```
http://otn.oracle.com/deploy/security/alerts.htm

http://technet.oracle.com/deploy/security/alerts.htm
```

Also check Oracle Worldwide Support Service's site, Metalink, for details on available and upcoming security-related patches.

`http://metalink.oracle.com`

9. CONTACT ORACLE SECURITY PRODUCTS

If you believe that you have found a security vulnerability in Oracle9*i*, submit an iTAR to Oracle Worldwide Support Services via Metalink, or e-mail `SECALERT_US@ORACLE.COM` a complete description of the problem, including product version and platform, together with any exploit scripts and / or examples.

A Security Checklist for Oracle9*i*
March 2001
Author: Rajiv Sinha
Contributing Authors: Kristy Browder, Mary Ann Davidson

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7410
www.oracle.com

Oracle Corporation provides the software
that powers the internet.