

# Oracle Database Security Checklist

*An Oracle White Paper*  
*November 2005*

# Oracle Database Security Checklist

<b>Protecting the Database Environment</b>	<b>4</b>
<b>Checklist Guidelines</b>	<b>5</b>
<b>Step 1: Install only what is required</b>	<b>5</b>
Options and Products	5
Sample Schemas	5
<b>Step 2: Lock and Expire Default User Accounts</b>	<b>5</b>
Enterprise Manager Accounts	6
<b>Step 3: Change Default User Passwords</b>	<b>6</b>
Change default passwords of administrative users	6
Change default passwords of all users	7
Enforce password management	7
<b>Step 4: Enable Data Dictionary Protection</b>	<b>7</b>
<b>Step 5: Practice the principle of least privilege</b>	<b>8</b>
Grant necessary privileges only	8
Revoke unnecessary privileges from the public user group	9
Grant a role to users only if they need all privileges of the role	10
Restrict permissions on run-time facilities	11
<b>Step 6: Enforce access controls effectively and authenticate clients stringently</b>	<b>11</b>
Authenticate client properly	11
<b>Step 7: Restrict Operating System Access</b>	<b>12</b>
<b>Step 8: Restrict Network Access</b>	<b>12</b>
Use a firewall	12
Never poke a hole through a firewall	13
Protect the Oracle listener	13
Monitor who accesses your systems	15
Check network IP addresses	15
Encrypt network traffic	16
Harden the operating system	16
<b>Step 9: Apply all security patches and</b>	<b>17</b>

<b>Step 10: Contact Oracle Security products if you come across a vulnerability in Oracle Database</b>	<b>17</b>
<b><i>APPENDIX A - Oracle Database 10g Release 1 and Release 2 Enterprise Edition Default Accounts and their Status (Standard install)</i></b>	<b>18</b>
<b><i>APPENDIX B - Oracle9i R2 Enterprise Edition Default Accounts and their Status (Standard Install)</i></b>	<b>20</b>
<b><i>APPENDIX C - Oracle9i R1 Enterprise Edition Default Accounts and their Status (Standard Install)</i></b>	<b>21</b>

## PROTECTING THE DATABASE ENVIRONMENT

For several major releases of the database, the Oracle documentation has provided a security checklist for customers to follow to help secure Oracle database environments. Since Oracle9i, Oracle has been working with customers to better understand their desired default configurations and harden the Oracle environment. Oracle Database 10g Release 1 Enterprise Manager (EM) Configuration pack introduced the policy manager which has the ability to scan databases and look for security related configuration status. The EM policy manager continues to be enhanced and provides a powerful tool for DBAs and security officers to scan installations for incorrect configuration settings such as unlocked default accounts. The Oracle Technology Network provides additional information on the EM policy manager tool and its configuration scanning capabilities.

The Oracle database provides robust security features which can be used to enhance existing application security. Technology such as client identifier can be used to propagate identities across application tiers, secure application roles can be used to better restrict access to database privileges based upon wide range of customer specific criteria, and Oracle database auditing can be used to help enforce the trust but verify principle, making sure DBAs work within their boundaries.

However, in order to fully maximize the security features offered by Oracle Database in any business environment, it is imperative that the database itself be well-protected. Furthermore, proper use of its security features and adherence to basic security practices will help protect against database-related threats and attacks. Such an approach provides a much more secure operating environment for Oracle Database.

This paper recaps the security checklist which can be found in newer versions of the Oracle Database Security Guide. This document provides guidance on configuring the Oracle Database in a secure manner by adhering to and recommending industry-standard and advisable security practices for operational database deployments.

Details on specific database-related tasks and actions can be found throughout the Oracle documentation set.

## CHECKLIST GUIDELINES

The following security checklist includes guidelines that help secure your database:

### Step 1: Install only what is required

#### Options and Products

The Oracle Database CD pack contains a host of options and products in addition to the database server. Install additional products and options only as necessary. Use Custom Installation to avoid installing unnecessary products or, perform a typical installation, and then deinstall unrequired options and products. There is no need to maintain additional products and options if they are not being used. They can always be properly and easily reinstalled as required.

#### Sample Schemas

Oracle Corporation provides Sample Schemas to provide a common platform for examples. If your database will be used in a production environment, then do not install the Sample Schema. If you have installed the Sample Schema on a test database, then before going production, remove or relock the Sample Schema accounts.

### Step 2: Lock and Expire Default User Accounts

Lock and expire default user accounts

Oracle Database installs with a number of default (preset) database server user accounts. Upon successful installation of the database server, the Database Configuration Assistant automatically locks and expires most default database user accounts.

If a manual (i.e. without using Database Configuration Assistant) installation of Oracle Database is performed, then *no default database users are locked upon successful installation of the database server*. Left open in their default states, these user accounts can be exploited to gain unauthorized access to data or disrupt database operations.

Therefore, after performing any kind of initial installation that does not use the Database Configuration Assistant, you should *lock* and *expire* all default database user accounts. Oracle Database provides SQL statements to perform such operations.

Installing additional products and components later also results in creating more default database server accounts. Database Configuration Assistant automatically locks and expires all additionally created database server user accounts. Unlock only those accounts that need to be accessed on a regular basis and assign a strong, meaningful password to each of these unlocked accounts. Oracle provides SQL and password management to perform such operations.

Please see appendixes A, B & C for a complete listing of default database accounts and their status after a standard Oracle database installation using Database Configuration Assistant.

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

#### **Enterprise Manager Accounts**

The preceding list of accounts depends on whether you choose to install Enterprise Manager. If so, `SYSMAN` and `DBSNMP` are open as well, unless you configure Enterprise Manager for Central Administration. In this case, the `SYSMAN` account (if present) will be locked as well.

If you do not install Enterprise Manager, then only `SYS` and `SYSTEM` are open. Database Configuration Assistant locks and expires all other accounts (including `SYSMAN` and `DBSNMP`).

### **Step 3: Change Default User Passwords**

#### **Change default user passwords**

The most trivial method by which Oracle Database can be compromised is a default database server user account which still has a default password associated with it *even after installation*. The following guidelines are recommended:

#### **Change default passwords of administrative users**

Oracle Database 10g enables you to use the same or different passwords for the `SYS`, `SYSTEM`, `SYSMAN` and `DBSNMP` administrative accounts. Use different passwords for each: in any Oracle environment (production or test), assign strong, secure, and distinct passwords to these administrative accounts. If Database Configuration Assistant is used, then it requires you to enter

passwords for the SYS and SYSTEM accounts, disallowing the use of the defaults `CHANGE_ON_INSTALL` and `MANAGER`.

Similarly, for production environments, do not use default passwords for any administrative accounts, including SYSMAN and DBSNMP.

At the end of database creation, Database Configuration Assistant displays a page requiring you to enter and confirm new passwords for the SYS and SYSTEM user accounts.

#### **Change default passwords of all users**

In Oracle Database, `SCOTT` no longer installs with default password `TIGER`, but instead is locked and expired, as is DBSNMP. Each of the other accounts install with a default password that is exactly the same as that user account (For example, user `MDSYS` installs with the password `MDSYS`).

If any of the default user accounts that were locked and expired upon installation need to be activated, then assign a new secure password to each such user account.

Even though Oracle does not explicitly mandate changing the default password for the user `SCOTT`, Oracle recommends that this user account also be locked in a production environment.

#### **Enforce password management**

Oracle recommends that basic password management rules (such as password length, history, complexity, and so forth) 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, using Oracle Advanced Security (an option to the Enterprise Edition of Oracle Database) 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 Oracle Database.

### **Step 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.

Enable data dictionary protection

To enable dictionary protection, set the following configuration parameter to `FALSE`, in the `init<sid>.ora` control file:

```
O7_DICTIONARY_ACCESSIBILITY = FALSE
```

By doing so, only those authorized users making DBA-privileged (for example `CONNECT / AS SYSDBA`) connections can use the `ANY` system privilege on the data dictionary. If `O7_DICTIONARY_ACCESSIBILITY` is not set to `FALSE`, then 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 *needs* view access to the data dictionary, then it is permissible to grant that user the `SELECT ANY DICTIONARY` system privilege.

**Notes:**

- Regarding `O7_DICTIONARY_ACCESSIBILITY`, note that in Oracle Database, the default is `FALSE`. However, in Oracle8i, this parameter is set to `TRUE` by default and must specifically be changed to `FALSE` to enable this security feature.
- Regarding the `SELECT ANY DICTIONARY` privilege: this privilege is not included in the `GRANT ALL PRIVILEGES` statement, but it can be granted through a role.

**Step 5: Practice the principle of least privilege**

The following guidelines are recommended:

**Practice the principle of least privilege**

**Grant necessary privileges only**

Do not provide database users more privileges than are necessary. In other words, the *principle of least privilege* is that users be given only those privileges that are actually required to efficiently perform their jobs.

To implement this principle, restrict the following as much as possible:

- 1) The number of `SYSTEM` and `OBJECT` privileges granted to database users, and
- 2) The number of people who are allowed to make `SYS`-privileged connections to the database.

For example, there is generally no need to grant `CREATE ANY TABLE` to any non-DBA-privileged user.

#### Revoke unnecessary privileges from the public user group

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, potentially enabling a minimally-privileged user to access and execute functions that he would not otherwise be permitted to access directly. The more powerful packages that may potentially be misused are listed in the following table:

Package	Description
<code>UTL_SMTP</code>	This package permits arbitrary mail messages to be sent from one arbitrary user to another arbitrary user. Granting this package to <code>PUBLIC</code> may permit unauthorized exchange of mail messages.
<code>UTL_TCP</code>	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.
<code>UTL_HTTP</code>	This package allows the database server to request and retrieve data using HTTP. Granting this package to <code>PUBLIC</code> may permit using HTML forms to send data to a malicious Web site.
<code>UTL_FILE</code>	If configured improperly, then this package allows text level access to any file on the host operating system. Even when properly

Package	Description
	<p>configured, this package may allow unauthorized access to sensitive operating system files, such as trace files, because it does not distinguish between its calling applications. The result can be that one application accessing <code>UTL_FILE</code> may write arbitrary data into the same location that is written to by another application.</p>

These packages should be revoked from `PUBLIC` and made executable for an application only when absolutely necessary.

These packages are extremely useful to the applications that need them. They require proper configuration and usage for safe and secure operation, and may not be suitable for most applications.

For applications that need these packages, create roles with `EXECUTE` privilege on the particular packages needed and assign those roles only to applications that specifically need to use them. Oracle intends to revoke such privileges from `PUBLIC` in subsequent releases.

**Grant a role to users only if they need all privileges of the role**

Roles (groups of privileges) are useful for quickly and easily granting permissions to users. Although you can use Oracle-defined roles, you have more control and continuity if you create your own roles containing only the privileges pertaining to your requirements. Oracle may change or remove the privileges in an Oracle-defined role, as it has with `CONNECT`, which in Oracle Database 10g Release 2 now has only the `CREATE SESSION` privilege. Formerly this role had eight other privileges. Both `CONNECT` and `RESOURCE` roles may be deprecated in future Oracle versions.

Ensure that the roles you define contain only the privileges that reflect job responsibility. If your application users do not need all the privileges encompassed by an existing role, then apply a different set of roles that supply just the right privileges. Alternatively, create and assign a more restricted role.

For example, it is imperative to strictly limit the privileges of `SCOTT`. Drop the `CREATE DBLINK` privilege for `SCOTT`. Then drop the entire role for the user, because privileges acquired by

Similarly, for even better security, drop the create database link privilege from all users who do not require it.

means of a role cannot be dropped individually. Recreate your own role with only the privileges needed, and grant that new role to that user. Similarly, for even better security, drop the `CREATE DBLINK` privilege from all users who do not require it.

#### 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.

Here is an example of a vulnerable run-time call:

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

Here is an example of a better (more secure) run-time call:

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

Enforce access controls and  
authenticate clients stringently

## Step 6: Enforce access controls effectively and authenticate clients stringently

#### Authenticate client properly

By default, Oracle allows operating-system-authenticated logins only over secure connections, which precludes using Oracle Net and a shared server configuration. This default restriction prevents a remote user from impersonating another operating system user over a network connection.

Setting the initialization parameter `REMOTE_OS_AUTHENT` to `TRUE` forces the RDBMS to accept the client operating system user name received over a nonsecure connection and use it for account access. Since clients, such as PCs, are not trusted to perform operating system authentication properly, it is very poor security practice to turn on this feature.

The default setting, `REMOTE_OS_AUTHENT = FALSE`, creates a more secure configuration that enforces proper, server-based authentication of clients connecting to an Oracle database.

You should not alter the default setting of the `REMOTE_OS_AUTHENT` initialization parameter, which is `FALSE`.

Setting this parameter to `FALSE` does not mean that users cannot connect remotely. It simply means that the database will not trust that the client has been already authenticated, and will therefore apply its standard authentication processes.

### Step 7: Restrict Operating System Access

Restrict operating system access

Limit the number of operating system users. Limit the privileges of the operating system accounts (administrative, root-privileged or DBA) on the Oracle Database host (physical machine) to the least privileges needed for the user's tasks.

Oracle also recommends:

- Restricting the ability to modify the default file and directory permissions for the Oracle Database home (installation) directory or its contents. Even privileged operating system users and the Oracle owner should not modify these permissions, unless instructed otherwise by Oracle.
- Restricting symbolic links. Ensure that when providing a path or file to the database, neither the file nor any part of the path is modifiable by an untrusted user. The file and all components of the path should be owned by the DBA or some trusted account, such as *root*.

This recommendation applies to all types of files: data files, log files, trace files, external tables, bfiles, and so on.

### Step 8: Restrict Network Access

Restrict network access

The following guidelines are recommended:

#### Use a firewall

Keep the database server behind a firewall. Oracle Database 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 Gauntlet from Network Associates and Raptor from Axent . Supported packet-filtering firewalls include PIX Firewall from Cisco, and supported stateful inspection firewalls (more

sophisticated packet-filtered firewalls) include Firewall-1 from CheckPoint .

#### **Never poke a hole through a firewall**

If Oracle Database is behind a firewall, then do not, under any circumstances, poke a hole through the firewall. For example, do not leave open port 1521 for Oracle Listener to make a connection to the Internet or vice versa.

Doing this 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 databases behind the firewall. Furthermore, an Oracle Listener running without an established password may be probed for critical details about the databases on which it is listening such as trace and logging information, banner information and database descriptors and service names.

All this information and the availability of an ill-configured firewall will provide an attacker ample opportunity to launch malicious attacks on the target databases.

#### **Protect the Oracle listener**

Because the listener acts as the database gateway to the network, it is important to limit the consequences of malicious interference:

- Restrict the privileges of the listener, so that it cannot read or write files in the database or the Oracle server address space.

This restriction prevents external procedure agents spawned by the listener (or procedures executed by such an agent) from inheriting the ability to do such reads or writes. The owner of this separate listener process should not be the owner that installed Oracle or executes the Oracle instance (such as [ORACLE](#), the default owner).

Sample configuration:

```
EXTPROC_LISTENER=  
  (DESCRIPTION=  
    (ADDRESS=  
      (PROTOCOL=ipc)(KEY=extproc)))  
SID_LIST_EXTPROC_LISTENER=  
  (SID_LIST=  
    (SID_DESC=  
      (SID_NAME=plsextproc)  
      (ORACLE_HOME=/u1/app/oracle/9.0)
```

```
(PROGRAM=extproc)))
```

- Secure administration of the database by doing the following:
  1. Prevent online administration by requiring the administrator to have write privileges on the `LISTENER.ORA` file and the listener password:

Add or alter this line in the `LISTENER.ORA` file

```
ADMIN_RESTRICTIONS_LISTENER=ON
```

Then `RELOAD` the configuration.

2. Use SSL when administering the listener, by making the `TCPS` protocol the first entry in the address list as follows:

```
LISTENER=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=
        (PROTOCOL=tcps)
        (HOST = ed-pdsun1.us.oracle.com)
        (PORT = 8281)))
```

To administer the listener remotely, you need to define the listener in the `listener.ora` file on the client computer. For example, to access listener `USER281` remotely, use the following configuration:

```
user281 =
  (DESCRIPTION =
    (ADDRESS =
      (PROTOCOL = tcps)
      (HOST = ed-pdsun1.us.oracle.com)
      (PORT = 8281))
    )
  )
```

**Always establish a secure, well-formed password for the Oracle listener to prevent remote configuration**

3. For Oracle database releases prior to Oracle Database 10g Release 1, it is very important to set a password for the Oracle TNS listener. Always establish a secure, well-formed password for the Oracle listener to prevent remote configuration of the Oracle listener. Password protect the listener as follows:

```
LSNRCTL> CHANGE_PASSWORD
Old password: lsnrc80
New password: lsnrc90
Reenter new password: lsnrc90
LSNRCTL> SET PASSWORD
Password:
The command completed successfully
LSNRCTL> SAVE_CONFIG
```

The command completed successfully

For Oracle Database 10g Release 1 and higher the default authentication mode is local OS authentication which requires administrator to be a member of the local dba group. Setting a password for the TNS listener in Oracle Database 10g Release 1 and higher simplifies administration. However, setting a password requires good password management to prevent unauthorized users from guessing the password and potentially gaining access to privileged listener operations. Customers may wish to consider not setting a password for the TNS listener starting with Oracle Database 10g Release 1.

4. Remove the external procedure configuration from the `listener.ora` file if you do not intend to use such procedures.
5. Monitor listener activity.

#### **Monitor who accesses your systems**

Authenticating client computers over the Internet is problematic. Do user authentication instead, which avoids client system issues that include falsified IP addresses, hacked operating systems or applications, and falsified or stolen client system identities. The following steps improve client computer security:

- Configure the connection to use SSL. Using SSL (Secure Sockets Layer) communication makes eavesdropping unfruitful and enables the use of certificates for user and server authentication.
- Set up certificate authentication for clients and servers such that:
  - i. The organization is identified by unit and certificate issuer and the user is identified by distinguished name and certificate issuer.
  - ii. Applications test for expired certificates.
  - iii. Certificate revocation lists are audited.

#### **Check network IP addresses**

Use 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 parameters respectively deny and allow specific client IP addresses from making connections to the Oracle listener (This helps in preventing potential Denial of Service attacks).

#### **Encrypt network traffic**

If possible, use Oracle Advanced Security to encrypt network traffic between clients, databases, and application servers.

#### **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, and so forth. 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.

### **Step 9: Apply all security patches and**

Always apply all relevant and current security patches for both the operating system on which Oracle Database resides and Oracle Database itself, and for all installed Oracle Database options and components.

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

<http://www.oracle.com/technology/deploy/security/alerts.htm>

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

<http://metalink.oracle.com>

### **Step 10: Contact Oracle Security products if you come across a vulnerability in Oracle Database**

If you believe that you have found a security vulnerability in Oracle Database, then submit an iTAR to Oracle Worldwide Support Services using Metalink, or e-mail a complete description of the problem, including product version and platform, together with any exploit scripts and examples to the following address:

`secalert_us@oracle.com`

**APPENDIX A - ORACLE DATABASE 10G RELEASE 1 AND RELEASE 2  
ENTERPRISE EDITION DEFAULT ACCOUNTS AND THEIR STATUS  
(STANDARD INSTALL)**

<b>Username</b>	<b>Account Status</b>
ANONYMOUS	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	EXPIRED & LOCKED
DIP	EXPIRED & LOCKED
DMSYS	EXPIRED & LOCKED
EXFSYS	EXPIRED & LOCKED
HR	EXPIRED & LOCKED
MDDATA	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
MGMT_VIEW	EXPIRED & LOCKED
ODM	EXPIRED & LOCKED
ODM_MTR	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
OLAPSYS	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OUTLN	EXPIRED & LOCKED
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

<b>Username</b>	<b>Account Status</b>
RMAN	EXPIRED & LOCKED
SCOTT	EXPIRED & LOCKED
SH	EXPIRED & LOCKED
SI_INFORMTN_SCHEMA	EXPIRED & LOCKED
SYS	OPEN
SYSMAN	EXPIRED & LOCKED
SYSTEM	OPEN
TSMSYS <b>New in 10g Release 2</b>	EXPIRED & LOCKED
WK_TEST	EXPIRED & LOCKED
WKPROXY	EXPIRED & LOCKED
WKSYS	EXPIRED & LOCKED
WMSYS	EXPIRED & LOCKED
XDB	EXPIRED & LOCKED

**APPENDIX B - ORACLE9i R2 ENTERPRISE EDITION DEFAULT ACCOUNTS AND THEIR STATUS (STANDARD INSTALL)**

<b>Username</b>	<b>Account Status</b>
ADAMS	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	OPEN
HR	EXPIRED & LOCKED
MDSYS	EXPIRED & LOCKED
ODM	EXPIRED & LOCKED
ODM_MTR	EXPIRED & LOCKED
ORDPLUGINS	EXPIRED & LOCKED
ORDSYS	EXPIRED & LOCKED
OUTLN	EXPIRED & LOCKED
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
WKPROXY	EXPIRED & LOCKED
WKSYS	EXPIRED & LOCKED
WMSYS	EXPIRED & LOCKED
XDB	EXPIRED & LOCKED

**APPENDIX C - ORACLE9i R1 ENTERPRISE EDITION DEFAULT ACCOUNTS AND THEIR STATUS (STANDARD INSTALL)**

<b>Username</b>	<b>Account Status</b>
ADAMS	EXPIRED & LOCKED
AURORA\$JIS\$UTILITY\$	OPEN
AURORA\$ORB\$UNAUTHENTICATED	OPEN
BLAKE	EXPIRED & LOCKED
CLARK	EXPIRED & LOCKED
CTXSYS	EXPIRED & LOCKED
DBSNMP	OPEN
JONES	EXPIRED & LOCKED
OE	EXPIRED & LOCKED
HR	EXPIRED & LOCKED
MDSYS	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

<b>Username</b>	<b>Account Status</b>
QS_WS	EXPIRED & LOCKED
SCOTT	OPEN
SH	EXPIRED & LOCKED
SYS	OPEN
SYSTEM	OPEN

**Oracle Database Security Checklist**

**November 2005**

**Author: Oracle Corporation**

**Contributing Authors:**

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

**Worldwide Inquiries:**

**Phone: +1.650.506.7000**

**Fax: +1.650.506.7200**

**oracle.com**

**Copyright © 2005, Oracle. All rights reserved.**

**This document is provided for information purposes only and the contents hereof are subject to change without notice.**

**This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.**