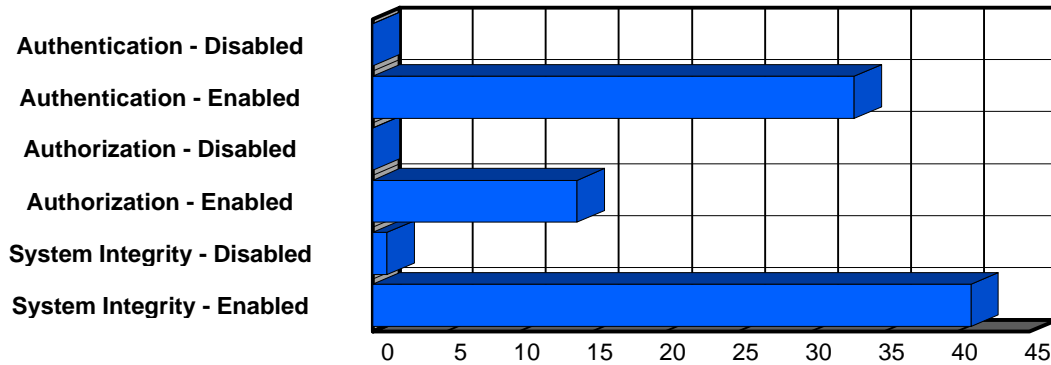


Description: Listed below are the details of your security policy. Policy details should be determined based on the acceptable level of risk for the system being assessed. More critical systems should adhere to a more stringent policy. The risk factors of the audited system are dynamic, and should be assessed periodically. The security policy should then be updated to allow for changes in the environment.

Checks By Category



Authentication

Check Name: Database Link Password Unencrypted

Description: Check that database link passwords are not stored in clear text. A database link is a mechanism used to provide a method of transparently accessing one server from another. When creating a link, a user name and password of an account on the remote server can be specified. If this is done, all queries using the link will have the privilege of the indicated account on the remote server. By omitting an account and password when creating a database link, the account and password of the user connecting through the link is used. Indicating the user name and password of an account to use for all connections through a link can lead to passwords being exposed. Database link passwords are stored unencrypted in the database. Users with SELECT privilege to the SYS.LINK\$ table can view the passwords in clear text. Setting up links to authenticate as the current user prevents unencrypted passwords from being exposed helps prevent linked servers from being compromised, and provides increased accountability.

Check Name: DBA Includes Non-default Account

Description: Check for non-typical members in the DBA role. The DBA role is very powerful and access to it should be restricted. Be sure any account with the DBA role is a legitimate member. (If the account is a legitimate member of this role, then this is not a vulnerability.) Access to the DBA role by unauthorized accounts may provide full access to the server.

Check Name: Default Accounts and Passwords

Description: Check for default passwords that have not been changed. Oracle databases have several well-known default username/password combinations. These combinations include the following: SCOTT/TIGER, DBSNMP/DBSNMP, SYSTEM/MANAGER, SYS/CHANGE_ON_INSTALL, TRACESVR/TRACE, CTXSYS/CTXSYS, MDSYS/MDSYS, DEMO/DEMO, CTXDEMO/CTXDEMO, APPLSYS/FND, PO8/PO8, NAMES/NAMES, SYSADM/SYSADM, ORDPLUGINS/ORDPLUGINS, OUTLN/OUTLN, ADAMS/WOOD, BLAKE/PAPER, JONES/STEEL, CLARK/CLOTH, AURORA\$ORB\$UNAUTHENTICATED/INVALID, and APPS/APPS. These default combinations may provide unauthorized access to the server.

Check Name: Default Internal Password

Description: Check that the internal password used to connect as internal has been changed from the default installation value ORACLE. Change this password immediately after installation. Leaving the default password could result in unauthorized users accessing the server as internal, allowing them full database administration privileges.

Check Name: Default Listener Password

Description: Check the listener.ora file to verify that the default listener password has been changed and is not blank. The listener password prevents unauthorized users from starting and stopping the listener service. Using the default password or a blank password could result in unauthorized users starting and stopping the listener service.

Check Name: Default SAP Account and Password

Description: Check that the default SAP password has been changed. SAP on Oracle uses the well-known default username/password combination of SAP/SAPR3. Leaving this default unchanged may allow unauthorized access to the SAP account.

Check Name: Default SNMP Account

Description: Check that the default SNMP password has been changed. SNMP on Oracle uses a well-known default username (DBSMP) and password combination. Leaving this default unchanged may allow unauthorized access to the SAP account.

Check Name: Excessive DBA Connections

Description: Check that an excessive number of connections do not have the DBA role at the time the scan is executed. Use of the DBA role should be limited to administrative tasks only. The Principle of Least Privileges dictates that accounts having admin privileges should not be used when doing non-admin tasks. When running tasks not requiring the privileges of the DBA role, it is recommended you use another, less powerful account. Finding more than a few users connected with the DBA role is an indicator that the DBA role is being used too freely.

Policy Setting: 3

Check Name: Expired Passwords Found in Oracle 7

Description: Check Oracle 7 servers for password expiration. Oracle 7 does not provide a password expiration mechanism. This check provides Oracle 7 databases the capability of monitoring password aging by looking for changes in the password hashes between scans. When a new password hash is detected, it is recorded as a password change and the password lifetime for the account is reset. Requiring password changes on a regular basis counters undetected password compromises. By determining and setting an appropriate password lifetime, the security risk associated with password authentication can be reduced. The longer a password is in use, the more likely the password will become exposed, whether through brute force, eavesdropping, or other avenues.

NOTE: The first time this scan runs establishes a baseline that Database Scanner uses to calculate password age. The first run of this scan will therefore not reveal expired passwords. Subsequent scans will determine password age based on earlier scans.

Policy Setting: Expires: 30 Warning: 20

Check Name: Expired Passwords Found in Oracle 8

Description: Check that password ages do not exceed a reasonable password lifetime. Oracle 8 introduced the ability to limit password lifetime through the use of profiles. This check uses the built-in password aging functionality of Oracle for version 8 servers and higher. The password lifetime will be taken from the profile associated with the accounts. Requiring password changes on a regular basis counters undetected password compromises. By determining and setting an appropriate password lifetime, the security risk associated with password authentication can be reduced. The longer a password is in use, the more likely the password will become exposed, whether through brute force, eavesdropping, or other avenues.

Policy Setting: 30

Check Name: Failed Login Attempts

Description: Check that any existing profiles have Failed Login Attempts limits within the allowed policy range. The FAILED_LOGIN_ATTEMPTS value serves as a limit to the number of failed login attempts allowed before an account is locked. Setting this value limits the ability of unauthorized users to guess passwords and alerts the DBA as to when password guessing occurs (accounts display as locked). Once an account is locked, it cannot be logged on to for a specified number of days or until the DBA unlocks the account. (See the Password Lock Time check.) FAILED_LOGIN_ATTEMPTS can be set to a number of attempts; UNLIMITED, meaning never lock an account; or to DEFAULT, which then uses the value indicated in the DEFAULT profile. This feature is set for profiles. These profiles then must be associated with an account. This check verifies that all profiles have a minimum level of security set.

Policy Setting: 3

Check Name: Internal Password in Spoolmain

Description: Check to determine if the internal password has been logged in the spoolmain.log file during install. This password can be stored unencrypted, providing easy access to unauthorized users.

Check Name: Listener Cleartext Password

Description: Check for the listener password being stored in clear text. If the listener password is being stored in clear text, it is very important to restrict read access to the listener.ora file. Easy access to the listener password may allow unauthorized users to control the listener service.

Check Name: Login Encryption Setting

Description: Check that encryption of passwords is enabled when connecting to Oracle from a client. Oracle 7.2 and later provide encryption of passwords between the client and server. If the parameter ORA_ENCRYPT_LOGIN is not set to TRUE (i.e., absent or set to FALSE) in the environment variables of the client, failed logon attempts will be retried sending the password in clear text.

Check Name: OS Authentication Prefix

Description: Check that the OS_AUTHENT_PREFIX setting is in compliance with the policy. Oracle can be configured to allow operating system accounts to be authenticated to Oracle without having to specify a password. When set up this way, OS accounts are mapped to Oracle accounts of the same name prefixed with the string specified by the OS_AUTHENT_PREFIX configuration parameter. By default, this value is OPSS\$. This means that the OS user account jdoe will be authenticated to Oracle as the Oracle account OPSS\$jdoe, if that account exists.

If the Oracle account being accessed has a valid password, then users may also login into Oracle using a username/password combination. If you set the prefix to anything other than OPSS\$, users can log into Oracle without specifying a password or by entering a valid username/password, but not both. Using the default prefix OPSS\$ allows remote users to attempt to guess passwords of accounts that have the OPSS\$ prefix but are not created using IDENTIFIED EXTERNALLY. By using a different prefix, accounts configured with the prefix must be IDENTIFIED EXTERNALLY if they are to use operating system authentication. Using any prefix other than OP\$ significantly reduces any chance of remote password guessing and makes guessing account names with the prefix harder.

Policy Setting: \$

Check Name: Password Attacks

Description: Check for evidence of password attacks. A password attack is a method of attempting to compromise a system by connecting using words from a dictionary for the password. People typically pick passwords that are easy to remember, such as names, birthdays, or words found in a dictionary. To prevent and detect this type of attack, set the Password Lockout feature for Oracle 8 and periodically review the audit logs for evidence of attacks. This check requires that auditing of failed connections be enabled and that auditing data be written to the SYS.AUD\$ table. Oracle 7 does not have a Failed Login Limit function. Check the audit log for evidence of successful attack.

Policy Setting: 3

Check Name: Password Grace Time

Description: Check that all profiles have a Password Grace Time within the limits of the policy. The PASSWORD_GRACE_TIME value serves as a limit to the number of days during which a password must be changed following the first successful login after password expiration. Setting this value ensures users are changing their passwords. PASSWORD_GRACE_TIME can be set to a number of days; UNLIMITED, meaning never require an account to change the password; or to DEFAULT, which then uses the value indicated in the DEFAULT profile. Leaving this value as UNLIMITED allows users to ignore the Change Password prompt indefinitely. This feature is set for profiles. These profiles then must be associated with an account. This check verifies that all profiles have a minimum level of security.

Policy Setting: 10

Check Name: Password Life Time

Description: Check that Oracle 8 profiles have not exceeded the allowed limit for Password Life Time. The PASSWORD_LIFE_TIME value serves as a limit to the number of days after which a password expires. Setting this value ensures users are changing their passwords. PASSWORD_LIFE_TIME can be set to a number of days; UNLIMITED, meaning never require an account to change the password; or to DEFAULT, which then uses the value indicated in the DEFAULT profile. Leaving this value on UNLIMITED allows users to use the same passwords indefinitely. This feature is set for profiles. These profiles then must be associated with an account. This check verifies that all profiles have a minimum level of security set.

Policy Setting: 20

Check Name: Password Lock Time

Description: Check that Oracle 8 profiles have not exceeded the allowed limit for PASSWORD_LOCK_TIME. The PASSWORD_LOCK_TIME value specifies the number of days to lock an account after the designated number of failed login attempts is reached. PASSWORD_LOCK_TIME can be set to a number of days; UNLIMITED; or to DEFAULT which then uses the value indicated in the DEFAULT profile. Setting this value on UNLIMITED requires that the database administrator unlock the account. This feature is set for profiles. These profiles then must be associated with an account. This check verifies that all profiles have a minimum level of security set.

Policy Setting: UNLIMITED

Check Name: Password Reuse Max

Description: Check that Oracle 8 profiles have not exceeded the allowed limit for PASSWORD_REUSE_MAX. The PASSWORD_REUSE_MAX value specifies the number of password changes before a password can be reused. PASSWORD_REUSE_MAX can be set to a number of reuses; UNLIMITED; or to DEFAULT, which then uses the value indicated in the DEFAULT profile. Setting this value to UNLIMITED allows passwords to be reused immediately. This feature is set for profiles. These profiles then must be associated with an account. PASSWORD_REUSE_MAX is mutually exclusive with PASSWORD_REUSE_TIME. If PASSWORD_REUSE_MAX is set to a value for a given profile, PASSWORD_REUSE_TIME must be set to UNLIMITED for the same profile. This check verifies that all profiles have a minimum level of security set.

Policy Setting: UNLIMITED

Check Name: Password Reuse Time

Description: Check that Oracle 8 profiles are not within the allowed limit for PASSWORD_REUSE_TIME. Oracle 8 introduces a new profile value, PASSWORD_REUSE_TIME. This value specifies the number of days before a password can be reused. PASSWORD_REUSE_TIME can be set to a number of days; UNLIMITED; or to DEFAULT, which then uses the value indicated in the DEFAULT profile. Setting this value to UNLIMITED allows passwords to be reused immediately. This feature is set for profiles. These profiles then must be associated with an account. PASSWORD_REUSE_TIME is mutually exclusive with PASSWORD_REUSE_MAX. If PASSWORD_REUSE_TIME is set to a value for a given profile, PASSWORD_REUSE_MAX must be set to UNLIMITED for the same profile. This check verifies that all profiles have a minimum level of security set.

Policy Setting: 365

Check Name: Password Verify Function

Description: Check that the Password Verify Function is specified properly. The PASSWORD_VERIFY_FUNCTION value specifies a PL/SQL function to be used for password verification when users who are assigned this profile log into a database. This function can be used to validate password strength by requiring passwords to pass a strength test written in PL/SQL. The function must be locally available for execution on the database to which this profile applies. Oracle provides a default script (utlpwdmg.sql), but you can also create your own function. The password verification function must be owned by SYS. The default setting for this profile parameter is NULL, meaning no password verification is performed.

Policy Setting: VERIFY_FUNCTION

Check Name: Remote Login Password File

Description: Check that the Oracle parameter REMOTE_LOGIN_PASSWORDFILE is in compliance with the policy. REMOTE_LOGIN_PASSWORDFILE specifies whether Oracle checks for a password file and how many databases can use the password file. Setting the parameter to NONE signifies that Oracle should ignore any password file (and only operating systems accounts in the dba group can connect INTERNAL). Setting the parameter to EXCLUSIVE signifies that the password file can be used by only one database and the password file can contain names other than SYS and INTERNAL (operating system users can still connect INTERNAL). Setting the parameter to SHARED allows more than one database to use a password file. However, the only users recognized by the password file are SYS and INTERNAL (operating system users can still connect INTERNAL). Setting the parameter to NONE, the recommended setting, prevents remote users from connecting as INTERNAL.

Policy Setting: NONE

Check Name: Role Passwords

Description: Check for roles without passwords. Oracle roles can be configured to require password authentication to use the role. In secure environments, sensitive roles should have passwords assigned to them. Oracle roles defined without password verification allow easy access.

Check Name: Stale Accounts

Description: Check for stale Oracle accounts. Oracle does not provide a mechanism to purge accounts that are unused. Accounts that are not being used expose several risks to a system. Stale accounts provide a point of attack for unauthorized users. Long term dictionary attacks are possible when a password is never changed. Stale accounts could also reflect accounts that were setup but never used and possibly have default passwords. Strong security policies require unused accounts to be removed. This check requires that auditing of successful connections be enabled and auditing data be written to the SYS.AUD\$ table.

Policy Setting: 30

Check Name: Trusting Remote OS Authentication Setting

Description: Check that the REMOTE_OS_AUTHENT parameter is not set to TRUE. Setting this value to TRUE allows operating system authentication over a non-secure connection. Trusting remote operating systems can allow a user to impersonate another operating system user and connect to the database without having to supply a password. If REMOTE_OS_AUTHENT is set to true, the only information a remote user needs to connect to the database is the name of any user whose account is setup to be authenticated by the operating system.

Check Name: Trusting Remote OS for Roles Setting

Description: Check that Oracle is not configured to enable roles based on remote operating system user group membership. Setting REMOTE_OS_ROLES to TRUE allows operating system groups to control Oracle roles. The default value of FALSE causes roles to be identified and managed by the database. If REMOTE_OS_ROLE is set to TRUE, a remote user could impersonate another operating system user over a network connection. It is a security risk to use operating system role authentication for network clients.

Check Name: Unencrypted SNMP Password

Description: Check to determine if the SNMP password is stored unencrypted in the snmp.ora or snmp_rw.ora file. The SNMP password is used to prevent unauthorized users from issuing commands to the database via SNMP. An unencrypted password may be easily stolen by unauthorized users.

Check Name: Weak Account Passwords

Description: Check that Oracle passwords are not easy to guess. Versions prior to Oracle 8 do not provide a mechanism to enforce password strength. In Oracle 8.0 and higher, a password verification script can be used to include some validation of password strength. Accounts configured with easily-guessed passwords are targets for password guessing. This check ensures that account passwords are not found in the specified dictionary or are not one of the following common passwords:

- password same as the account name
- password same as the account name reversed
- password same as the server name
- password is the account name appended with a number between 1 and 100

Policy Setting: <Use larger password dictionary>

Check Name: Weak Internal Password

Description: Check for a weak internal password. Compare the internal password against a dictionary of commonly used passwords. Passwords found in dictionaries should be considered weak since they are vulnerable to password guessing. A compromised internal password could result in unauthorized users accessing the server as sys allowing them full database administration privileges.

Policy Setting: <Use larger password dictionary>

Check Name: Weak Listener Passwords

Description: Check for weak passwords in the listener.ora file. Compare the password against a dictionary of commonly used passwords. Passwords found in dictionaries should be considered weak since they are vulnerable to password guessing. The listener password is used to prevent unauthorized users from issuing commands to the listener service. Easily guessed passwords could result in unauthorized users starting and stopping the listener service.

NOTE: Some versions of Oracle encrypt the password stored in the file. The current version of Database Scanner only determines if the listener password is weak if the password is stored unencrypted in the listener.ora file.

Policy Setting: <Use larger password dictionary>

Check Name: Weak Passwords for SYSDBA/SYSOPER

Description: Check that accounts with the SYSDBA or SYSOPER role do not have weak passwords. The passwords of accounts in the SYSDBA or SYSOPER role were compared against a dictionary of passwords and a match was found. Passwords that are found in dictionaries should be considered weak since they are vulnerable to password guessing. The password hashes for these passwords are not only stored in the SYS.USER\$ table, but also in the operating system file orapw<SID> (for Unix systems) or pwd<SID>.ora (for NT Systems). Compromising the password of accounts in the SYSDBA or SYSOPER role allows unauthorized users access to the server with full database administration privileges.

Policy Setting: <Use larger password dictionary>

Check Name: Weak SNMP Password

Description: Check for weak passwords in the SNMP file. Compare the password against a dictionary of commonly used passwords. Words found in dictionaries should be considered weak passwords since they are vulnerable to password guessing. The SNMP password is used to prevent unauthorized users from issuing commands to the database via SNMP. Easily guessed passwords could result in unauthorized users starting and stopping the SNMP service.

NOTE: Some versions of Oracle encrypt the password stored in the file. The current version of Database Scanner only determines if the SNMP password is weak if the password is stored unencrypted in the SNMP file.

Policy Setting: <Use larger password dictionary>

Number of Authentication Checks Enabled:

33

Authorization

Check Name: Account Permissions

Description: Check for account permissions not in compliance with the policy. Granting permissions to accounts is error prone and repetitive. Instead, consider assigning permissions to roles and then granting the roles to accounts.

Allowed Account Permissions

None

Check Name: Audit Table Permissions

Description: Check permissions on the audit table. Permissions to this table should be restricted to only those accounts requiring access. Granting excessive permissions could lead to tampering of the audit trail data. Check that only the appropriate accounts have permissions to perform select, insert, delete, or update operations on the table where the audit data is stored (SYS.AUD\$).

Check Name: Data Dictionary Accessibility

Description: Check that the parameter O7_DICTIONARY_ACCESSIBILITY is set to false. Oracle 8 provides the parameter O7_DICTIONARY_ACCESSIBILITY to prevent accounts with the privilege SELECT ANY TABLE from selecting on the data dictionary tables. Setting this parameter to FALSE helps restrict access to sensitive data in the data dictionary such as the encrypted passwords.

Policy Setting: FALSE

Check Name: Database Link Permissions

Description: Check for accounts with permissions to view the table SYS.LINK\$. Access to view the table SYS.LINK\$ should be restricted because database link passwords are stored unencrypted in this table.

Check Name: Default Tablespace

Description: Check if accounts are using the SYS or SYSTEM tablespaces. Use of the SYS or SYSTEM table space as the default tablespace is highly discouraged. New objects created by the account will be placed on this tablespace. The SYS or SYSTEM tablespace contains the data dictionary and should not be used for other tables.

Check Name: Logon Hours Violations

Description: Review audit logs for after hours connections. Security attacks often take place during non-business hours. Attackers are more active during this time. Physical connections are often left unattended, and the chance of being detected is less. Reviewing connections made during these hours is important in finding unauthorized activity.

	AM											PM												
Day	12	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11
Sun																								
Mon									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Tue									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Wed									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Thu									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Fri									X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Sat																								

Check Name: Privileges Granted With Admin

Description: Check that privileges having the WITH ADMIN OPTION have not been granted. Revoking system privileges having the WITH ADMIN OPTION does not revoke those privileges from other accounts that have been given the privilege by the account being revoked. This makes revoking system privileges that were granted WITH ADMIN OPTION difficult as the privilege can be given to an account and then granted back after it is revoked.

Check Name: PUBLIC Object Permissions

Description: Check for object permissions granted to PUBLIC. Granting permissions to PUBLIC should be restricted to regulate access to objects.

Check Name: PUBLIC System Privileges

Description: Check for system privileges granted to PUBLIC. System privileges can be granted to users and roles. Many of these privileges convey considerable authority over database objects and should only be granted to those persons responsible for administering the database. In general, these privileges should be granted to roles and then the appropriate roles should be granted to users. System privileges should never be granted to PUBLIC as this would allow users to compromise the database.

Check Name: Role Permissions

Description: Check that role permissions are in compliance with the policy. Oracle does not provide a method of granting row-level permissions for SELECT, DELETE, or UPDATE statements. It is good security practice to limit the granting of these permissions. You should instead provide access to these statements through functions, procedures, or views that provide validation that only the appropriate rows are being modified or viewed. To ensure maximum security, the only permissions allowed should be execute permissions on procedures and functions or select from views.

Allowed Role Permissions

EXECUTE from Package/Function
SELECT from Sequence

Check Name: Roles Granted With Admin

Description: Check for roles granted using the WITH ADMIN OPTION. Revoking roles having WITH ADMIN OPTION does not cascade to the accounts that were assigned the role by the account having the role revoked. This makes it difficult to revoke roles that were granted using the WITH ADMIN OPTION because the role can be granted to a third account and then granted back after it is revoked.

Check Name: Use of CONNECT Default Role

Description: Check that accounts have not been granted the CONNECT role. The CONNECT role includes the system privileges CREATE TABLE, CREATE DATABASE LINK, and several others which give users more privileges than required to connect to a database. Use of this role is strongly discouraged. Instead of using the CONNECT role to grant access to Oracle, consider creating a user defined role containing only the CREATE SESSION privilege and then granting this role to accounts.

Check Name: Use of RESOURCE Default Role

Description: Check for use of the RESOURCE role. The built-in role RESOURCE is not recommended for application users. This role conveys privileges that are not required by most accounts functioning within the bounds of an application and conveys no indication of the purpose of the privilege by its name.

Check Name: With Grant Option

Description: Check for object privileges granted using the WITH GRANT OPTION. A user having object privileges with the GRANT OPTION can grant privileges to other users just as if he were the owner of the object. Use of the WITH GRANT OPTION may not be appropriate in certain database environments and use of this option should be limited to security administrators.

Number of Authorization Checks Enabled:

14

System Integrity

Check Name: Audit Table Tablespace

Description: Check that the audit trail table (SYS.AUD\$) has not been installed in the system tablespace. The audit trail table should be placed in its own tablespace to avoid fragmentation of the system tablespace and to avoid running out of space in the system tablespace.

Check Name: Audit Trail Location

Description: Check the audit trail destination. Oracle auditing can be set to log audit data to the database or operating system files. Logging events to the database prevents operating system users from viewing the data, while logging events to operating system files prevents malicious database users from accessing the data.

Policy Setting: DB

Check Name: Auditing of Commands

Description: Check that system-wide auditing of statements and privileges is configured in accordance with policy. Configuring proper auditing is critical to recording any malicious events or detecting when attacks on the database occur. Auditing can be turned on for any SQL statement or use of a system privilege. Auditing can be enabled for all users (systemwide) or specific users. You may indicate whether one audit record for each access to an object or one audit record for the entire session is generated. You can enable auditing for commands that result in success, commands that result in failure, or both. Not all audit options can be audited by session. Audit options set using the BY SESSION clause for those actions that will not produce a session audit record will default to BY ACCESS.

<i>Audit Options</i>	<i>Success</i>	<i>Failure</i>
CLUSTER	By Access	By Access
CONTEXT	By Access	By Access
DATABASE LINK	By Access	By Access
DIMENSION	By Access	By Access
DIRECTORY	By Access	By Access
INDEX	By Access	By Access
NOT EXISTS	By Access	By Access
PROCEDURE	By Access	By Access
PROFILE	By Access	By Access
PUBLIC DATABASE LINK	By Access	By Access
PUBLIC SYNONYM	By Access	By Access
ROLE	By Access	By Access
ROLLBACK SEGMENT	By Access	By Access
SEQUENCE	By Access	By Access
SESSION	By Access	By Access
SYNONYM	By Access	By Access
SYSTEM AUDIT	By Access	By Access
SYSTEM GRANT	By Access	By Access
TABLE	By Access	By Access
TABLESPACE	By Access	By Access
TRIGGER	By Access	By Access
TYPE	By Access	By Access
USER	By Access	By Access
VIEW	By Access	By Access
ALTER ANY CLUSTER	By Access	By Access
ALTER ANY DIMENSION	By Access	By Access
ALTER ANY INDEX	By Access	By Access
ALTER ANY INDEXTYPE	By Access	By Access
ALTER ANY LIBRARY	By Access	By Access
ALTER ANY OUTLINE	By Access	By Access
ALTER ANY PROCEDURE	By Access	By Access
ALTER ANY ROLE	By Access	By Access
ALTER ANY SEQUENCE	By Access	By Access
ALTER ANY SNAPSHOT	By Access	By Access
ALTER ANY TABLE	By Access	By Access
ALTER ANY TRIGGER	By Access	By Access
ALTER ANY TYPE	By Access	By Access
ALTER DATABASE	By Access	By Access
ALTER PROFILE	By Access	By Access
ALTER RESOURCE COST	By Access	By Access
ALTER ROLLBACK SEGMENT	By Access	By Access
ALTER SEQUENCE	By Access	By Access
ALTER SESSION	By Access	By Access
ALTER SYSTEM	By Access	By Access
ALTER TABLE	By Access	By Access
ALTER TABLESPACE	By Access	By Access
ALTER USER	By Access	By Access
ANALYZE ANY	By Access	By Access
AUDIT ANY	By Access	By Access
BACKUP ANY TABLE	By Access	By Access
BECOME USER	By Access	By Access

COMMENT ANY TABLE	By Access	By Access
COMMENT TABLE	By Access	By Access
CREATE ANY CLUSTER	By Access	By Access
CREATE ANY CONTEXT	By Access	By Access
CREATE ANY DIMENSION	By Access	By Access
CREATE ANY DIRECTORY	By Access	By Access
CREATE ANY INDEX	By Access	By Access
CREATE ANY INDEXTYPE	By Access	By Access
CREATE ANY LIBRARY	By Access	By Access
CREATE ANY OPERATOR	By Access	By Access
CREATE ANY OUTLINE	By Access	By Access
CREATE ANY PROCEDURE	By Access	By Access
CREATE ANY SEQUENCE	By Access	By Access
CREATE ANY SNAPSHOT	By Access	By Access
CREATE ANY SYNONYM	By Access	By Access
CREATE ANY TABLE	By Access	By Access
CREATE ANY TRIGGER	By Access	By Access
CREATE ANY TYPE	By Access	By Access
CREATE ANY VIEW	By Access	By Access
CREATE CLUSTER	By Access	By Access
CREATE DATABASE LINK	By Access	By Access
CREATE DIMENSION	By Access	By Access
CREATE INDEXTYPE	By Access	By Access
CREATE LIBRARY	By Access	By Access
CREATE OPERATOR	By Access	By Access
CREATE PROCEDURE	By Access	By Access
CREATE PROFILE	By Access	By Access
CREATE PUBLIC DATABASE LINK	By Access	By Access
CREATE PUBLIC SYNONYM	By Access	By Access
CREATE ROLE	By Access	By Access
CREATE ROLLBACK SEGMENT	By Access	By Access
CREATE SEQUENCE	By Access	By Access
CREATE SESSION	By Access	By Access
CREATE SNAPSHOT	By Access	By Access
CREATE SYNONYM	By Access	By Access
CREATE TABLE	By Access	By Access
CREATE TABLESPACE	By Access	By Access
CREATE TRIGGER	By Access	By Access
CREATE TYPE	By Access	By Access
CREATE USER	By Access	By Access
CREATE VIEW	By Access	By Access
DELETE ANY TABLE	By Access	By Access
DELETE TABLE	By Access	By Access
DROP ANY CLUSTER	By Access	By Access
DROP ANY CONTEXT	By Access	By Access
DROP ANY DIMENSION	By Access	By Access
DROP ANY DIRECTORY	By Access	By Access
DROP ANY INDEX	By Access	By Access
DROP ANY INDEXTYPE	By Access	By Access
DROP ANY LIBRARY	By Access	By Access
DROP ANY OPERATOR	By Access	By Access
DROP ANY OUTLINE	By Access	By Access

DROP ANY PROCEDURE	By Access	By Access
DROP ANY ROLE	By Access	By Access
DROP ANY SEQUENCE	By Access	By Access
DROP ANY SNAPSHOT	By Access	By Access
DROP ANY SYNONYM	By Access	By Access
DROP ANY TABLE	By Access	By Access
DROP ANY TRIGGER	By Access	By Access
DROP ANY TYPE	By Access	By Access
DROP ANY VIEW	By Access	By Access
DROP PROFILE	By Access	By Access
DROP PUBLIC DATABASE LINK	By Access	By Access
DROP PUBLIC SYNONYM	By Access	By Access
DROP ROLLBACK SEGMENT	By Access	By Access
DROP TABLESPACE	By Access	By Access
DROP USER	By Access	By Access
EXECUTE ANY INDEXTYPE	By Access	By Access
EXECUTE ANY LIBRARY	By Access	By Access
EXECUTE ANY OPERATOR	By Access	By Access
EXECUTE ANY PROCEDURE	By Access	By Access
EXECUTE ANY TYPE	By Access	By Access
EXECUTE LIBRARY	By Access	By Access
EXECUTE PROCEDURE	By Access	By Access
EXTENDS ANY TYPE	By Access	By Access
FORCE ANY TRANSACTION	By Access	By Access
FORCE TRANSACTION	By Access	By Access
GRANT ANY PRIVILEGE	By Access	By Access
GRANT ANY ROLE	By Access	By Access
GRANT DIRECTORY	By Access	By Access
GRANT PROCEDURE	By Access	By Access
GRANT SEQUENCE	By Access	By Access
GRANT TABLE	By Access	By Access
GRANT TYPE	By Access	By Access
INSERT ANY TABLE	By Access	By Access
INSERT TABLE	By Access	By Access
LOCK ANY TABLE	By Access	By Access
LOCK TABLE	By Access	By Access
MANAGE TABLESPACE	By Access	By Access
NETWORK	By Access	By Access
RESTRICTED SESSION	By Access	By Access
SELECT ANY SEQUENCE	By Access	By Access
SELECT ANY TABLE	By Access	By Access
SELECT SEQUENCE	By Access	By Access
SELECT TABLE	By Access	By Access
SYSDBA	By Access	By Access
SYSOPER	By Access	By Access
UNLIMITED TABLESPACE	By Access	By Access
UPDATE ANY TABLE	By Access	By Access
UPDATE TABLE	By Access	By Access

Check Name: Auditing of Schema Objects

Description: Collect object auditing configuration. This data can be reviewed in the Audit Configuration report. Object level auditing can be enabled for Oracle objects such as tables, views, stored procedures, functions, and packages. One or more operations for each object can be audited for all users or specific users. In addition, you may choose to generate one record for each access to an object (BY ACCESS) or to create one audit record per session (BY SESSION).

Check Name: Composite Resource Usage Limit

Description: Check that profiles do not exceed the specified resource usage on the Composite Resource Usage parameter. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should not exceed the specified Composite Resource Usage. Limits help prevent some Denial of Service attacks. Composite Resource Usage limits the total cost of resources used for a session. The resource cost for a session is the weighted sum of the CPU time used in the session, the connect time, the number of reads made in the session, and the amount of private SGA space allocated.

Policy Setting: 1000000

Check Name: Concurrent Sessions Resource Usage Limit

Description: Check that profiles do not exceed the specified resource usage on the Concurrent Sessions Resource Usage parameter. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should be checked to ensure they do not exceed the specified Concurrent Sessions Resource Usage. Concurrent Sessions Resource Usage limits the number of connections that a user can establish without releasing previous connections.

Policy Setting: 1

Check Name: Connect Time Resource Usage Limit

Description: Check that any existing profiles have Connect Time Resource Usage limits within the policy-allowed range. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should not exceed the specified Connect Time Resource Usage limit. Connect Time Resource Usage limits restrict the maximum elapsed time allowed for a session. The limit is expressed in minutes. Setting a Connect Time Resource Usage limit helps prevent users from monopolizing a system and can ensure that resources are released when a user leaves his workstation without logging off the system.

Policy Setting: 90

Check Name: CPU/Call Resource Usage Limit

Description: Check that any existing profiles have CPU/Call Resource Usage limits within the policy-allowed range. Oracle allows limitation of system resources to be set. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should not exceed the specified CPU/Call Resource Usage limit. Limits help prevent Denial of Service attacks. CPU/Call limits restrict the maximum amount of total CPU time allowed for a call (a parse, execute, or fetch). The limit is expressed in seconds.

Policy Setting: 100000

Check Name: CPU/Session Resource Usage Limit

Description: Check that any existing profiles have CPU/Session Resource Usage limits within the policy-allowed range. Oracle allows limitation of system resources to be set. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should not exceed the specified CPU/Session Resource Usage limit. Limits help prevent some Denial of Service attacks. CPU/Session limits restrict the maximum amount of total CPU time allowed in a session. The limit is expressed in seconds.

Policy Setting: 1000000

Check Name: Database Link Password Encryption

Description: Check that the database link password encryption is properly set. The Oracle configuration parameter DBLINK_ENCRYPT_LOGIN specifies whether attempts to connect to remote Oracle databases through database links should use encrypted passwords. Prior to Oracle 7.2, passwords were not encrypted before being sent over the network. In order to connect to older servers, Oracle included this parameter to retry failed connections using the unencrypted format. If the DBLINK_ENCRYPT_LOGIN parameter is TRUE, and the connection fails, Oracle does not re-attempt the connection. If this parameter is FALSE, Oracle re-attempts the connection using an unencrypted version of the password. Servers with DBLINK_ENCRYPT_LOGIN set to FALSE can be coerced into sending unencrypted passwords by machines between linked servers.

Check Name: Default Password Verify Function

Description: Check that the default password verify function, VERIFY_FUNCTION, has not been modified.

Check Name: File Checksums

Description: Look for changes to the checksums for files in the \$ORACLE_HOME\bin directory tree against the values recorded in the previous scan. Changes in checksum values may indicate unauthorized access and/or change to a system file. If a change is detected, Database Scanner records a new value and generates a vulnerability alert.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

NOTE: The first time this scan runs establishes a baseline that Database Scanner uses to detect file changes. The first run of this scan will therefore not reveal file changes. Subsequent scans will detect file changes based on the last scan run.

Check Name: File Group

Description: Check the \$ORACLE_HOME directory and other Oracle common system files for group privileges. All files should be set to the group attached to the Oracle owner's account.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

Check Name: File Modifications

Description: Check for file size, date, file deletion and readditions for all files in the \$ORACLE_HOME directory to find file modifications. Examining modified files such as scripts, batch files, or executable files can help locate possible Trojan horses. File changes to non-executable files, such as log files and data files, do not usually reflect unauthorized changes. These files are modified by Oracle. These false positives can be ignored.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

NOTE: The first time this scan runs establishes a baseline that Database Scanner uses to detect file changes. The first run of this scan will therefore not reveal file changes. Subsequent scans will detect file changes based on the last scan run.

Check Name: File Owner

Description: Check that all files in the \$ORACLE_HOME directory and other Oracle common system files are owned by the Oracle software owner. All files should be owned by the Oracle software owner.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

Check Name: File Permissions

Description: Check permissions on system files. Permissions to files or directories in an Oracle installation should be restricted to the Oracle software owner and group. Look for files that have excessive permissions assigned. When checking Windows NT servers, every file that grants any permission to the group 'Everyone' will be reported as a violation. Checks on Unix servers will narrow the scope of potential violations based upon the setting of the 'Allowed File Permissions' field on the Security Policy Editor.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

Policy Setting: 750

Check Name: File Permissions - listener.ora

Description: Check the permissions of the operating system file listener.ora. This file contains the listener password. Permissions to this file should be restricted. Access to read this file could allow someone to determine the listener service password and to start and stop the listener service. When checking Windows NT servers, a file that grants any permission to the group 'Everyone' will be reported as a violation. Checks on Unix servers will narrow the scope of potential violations based upon the setting of the 'Listener file Permissions' field on the Security Policy Editor.

Policy Setting: 700

Check Name: File Permissions - snmp file

Description: Check the snmp.ora or snmp_rw.ora files for weak permissions. The SNMP password (contained in the SNMP file) is used to prevent unauthorized users from issuing commands to the database via SNMP. The file has 'Everyone' permissions allocated. When checking Windows NT servers, a file that grants any permission to the group 'Everyone' will be reported as a violation. Checks on Unix servers will narrow the scope of potential violations based upon the setting of the 'Allowed File Permissions' field on the Security Policy Editor.

Policy Setting: 700

Check Name: File Permissions - strtSID.cmd

Description: Check the permissions for the Oracle startup file. This file contains the internal password.

Check Name: File Permissions - SYSDBA password file

Description: Check the permissions of the operating system file orapw<SID> (on Unix systems) or pwd<SID>.ora (on Windows NT systems). Oracle stores the internal and SYS password and passwords of accounts granted the SYSDBA or SYSOPER role in this text file. Although the passwords are encrypted, privilege to this file should be restricted to prevent brute force attacks against the encrypted passwords. Access to read this file could allow someone to determine the internal or SYS password, and would allow an unauthorized user full administrative access to Oracle. When checking Windows NT servers, a file that grants any permission to the group 'Everyone' will be reported as a violation. Checks on Unix servers will narrow the scope of potential violations based upon the setting of the 'Password file Permissions' field on the Security Policy Editor.

Policy Setting: 700

Check Name: Idle Time Resource Usage Limit

Description: Check that any existing profiles have Idle Time Resource Usage limits within the policy-allowed range. This setting limits the maximum idle time allowed in a session. Idle time is a continuous period of inactive time during a session. Long-running queries and other operations are not subject to this limit. The limit is expressed in minutes. Setting a Idle Time Resource Usage limit helps prevent users from leaving applications open when they are away from their desks.

Policy Setting: 15

Check Name: Intelligent Agent Patch

Description: Check that the Intelligent Agent patch is installed. Vulnerabilities have been discovered in executables files used by the intelligent agent that are owned by root and have the setuid bit on. These vulnerabilities allow a user that is able to execute these files to gain access to the system as root. For additional information see ISS Security Advisory "Root Compromise Vulnerabilities in Oracle 8" dated August 23, 1999.

Check Name: Oracle Licensing Compliance

Description: Check that licensing is enabled and the warning level has not been exceeded. Oracle licensing is implemented in one of two ways - on a maximum user basis or on a maximum session basis. Setting the LICENSE_MAX_SESSIONS parameter controls the maximum number of concurrent user sessions allowed simultaneously. Setting the LICENSE_MAX_USERS parameter controls the maximum number of users you can create in the database. Setting these parameters to 0 causes these limits to be ignored. Concurrent usage (session) licensing and user licensing should not both be enabled. Either LICENSE_MAX_SESSIONS or LICENSE_MAX_USERS should be zero. The parameter LICENSE_SESSIONS_WARNING sets a warning limit on the number of concurrent user sessions.

Check Name: Oracle SQL92_SECURITY

Description: Check that the SQL92_SECURITY parameter is enabled. The configuration option SQL92_SECURITY specifies whether table-level SELECT privileges are required to execute an update or to delete that reference's table column values. If this option is not enabled, the UPDATE privilege can be used to determine values that should require SELECT privileges.

Check Name: Password Verify Function Changes

Description: Check for changes in the functions being used for password strength verification.

Check Name: Private SGA Resource Usage Limit

Description: Check that any existing profiles have Private SGA Resource Usage limits within the policy-allowed range. Limitations are assigned to system resource profiles, then system resource profiles should be checked to ensure they do not exceed the specified Private SGA Resource Usage limit. Limits help prevent some Denial of Service attacks. Private SGA Limits restrict the maximum amount of private space a session can allocate in the shared pool of the System Global Area (SGA). The Private SGA Resource Usage limit applies only if you are using the multi-threaded server architecture. The limit is expressed in kilobytes (Kbytes).

Policy Setting: 256

Check Name: Privileged OS Users

Description: Check for users that belong to operating system groups that give them access to the database with SYSDBA and/or SYSOPER privilege. Oracle allows operating system users to connect to the database as INTERNAL or with SYSDBA and SYSOPER privilege, based on membership in special operating system defined groups. On Unix platforms, the SYSDBA and SYSOPER groups are designated when the server is installed and both are set to the DBA group by default. For Windows NT these operating system groups are ORA_<sid>_DBA, ORA_<sid>_OPER, ORA_DBA, and ORA_OPER.

User in these groups can connect to the database locally without using a password. Users granted the SYSOPER role have considerable privileges within the database including the ability to shutdown and start the database. Users granted the SYSDBA role have full administrative rights.

Check Name: Reads/Call Resource Usage Limit

Description: Check that any existing profiles have Reads/Call Resource Usage limits within the policy-allowed range. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should be checked to ensure they do not exceed the specified Reads/Call Resource Usage limit. Limits help prevent some Denial of Service attacks. Reads/Call Resource Usage limits restrict the Maximum number of data block reads allowed for a call (a parse, execute, or fetch) to process a SQL statement. The limit includes blocks read from memory and disk.

Policy Setting: 5000

Check Name: Reads/Session Resource Use Limit

Description: Check that any existing profiles have Reads/Session Resource Usage limits within the policy allowed range. Limitations are assigned to system resource profiles, then system resource profiles are assigned to users. System resource profiles should be checked to ensure they do not exceed the specified Reads/Session Resource Usage limit. Limits help prevent some Denial of Service attacks.

Reads/Session Resource Usage limits restrict the total number of data block reads allowed in a session. The limit includes blocks read from memory and disk.

Policy Setting: 50000

Check Name: Registry Permissions

Description: Checks that the group 'Everyone' does not have permissions to any sub keys or values in the Oracle registry key. Information such as the location of configuration files and password files are stored in the registry under the key

HKEY_LOCAL_MACHINE\Software\ORACLE. Permitting read access to these values could allow users to collect info to mount an attack. Allowing users to change these values can result in redirecting Oracle to use different configuration and password files. For example, being able to change the ORA_PWFILFILE value would allow a user to point an Oracle server to their own password file, allowing them into the database as sysdba. Access to the registry should be restricted to the Oracle software owner only.

Check Name: Resource Limits Not Enabled

Description: Check that the configuration option RESOURCE_LIMIT is set to TRUE. If Oracle resource limits are disabled any profile limits that are set will be ignored.

NOTE: This does not apply to password resources.

Check Name: Setgid Bit

Description: Check for Oracle files with the setgid bit enabled. Oracle files should not have the setgid bit enabled. Enabling the setgid bit on an executable file causes the file to execute using the permissions of the file's group rather than the permissions of the user executing the file. Enabling the setgid bit on a directory causes the setuid bit to be enabled for any new files created in the directory. This could allow malicious users to gain elevated privileges for executable files that are programmed insecurely.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

Check Name: Setuid Bit

Description: Check if any Oracle files have the setuid bit enabled. Enabling the setuid bit on an executable file causes the file to execute using the permissions of the file owner rather than the permissions of the user executing the file. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle user with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle user. Attackers may also be able to permanently elevate their privilege to that of the Oracle software owner.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Due to limited security exposure, this check does not scan files contained in the \$ORACLE_HOME\doc directory.

Check Name: Setuid Bit of File cmctl

Description: Check if the file \$ORACLE_HOME\bin\cmctl has the setuid bit on. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle software owner with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle user. Attackers may also be able to permanently elevate their privilege to that of the Oracle user.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Check Name: Setuid Bit of File onrsd

Description: Check if the file \$ORACLE_HOME\bin\onrsd has the setuid bit enabled. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle software owner with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle software owner. Attackers may also be able to permanently elevate their privilege to that of the Oracle software owner.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Check Name: Setuid Bit of File oracleO

Description: Check if the file \$ORACLE_HOME\bin\oracleO has the setuid bit enabled. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle software owner with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle user. Attackers may also be able to permanently elevate their privilege to that of the Oracle software owner.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Check Name: Setuid Bit of File oratclsh

Description: Check if the file \$ORACLE_HOME\bin\oratclsh has the setuid bit enabled. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle software owner with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle user. Attackers may also be able to permanently elevate their privilege to that of the Oracle software owner.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Check Name: Setuid Bit of File otrccref

Description: Check if the file \$ORACLE_HOME\bin\otrccref has the setuid bit on. The Oracle distribution is shipped with many administrative utilities that are owned by the Oracle software owner with the setuid bit enabled. Several of these utilities practice insecure file creation and manipulation. These utilities also trust Oracle-related environment variables. The combined effect of these vulnerabilities may allow local attackers to create, append to, or overwrite privileged Oracle files. Certain vulnerabilities exist that may allow local attackers to execute arbitrary commands as the Oracle user. Attackers may also be able to permanently elevate their privilege to that of the Oracle software owner.

Temporary files that follow symbolic links are a common source of vulnerabilities in setuid executables. Administrators should remove or restrict access to setuid executables if possible.

Check Name: UTL_FILE Permissions

Description: Check permissions on the UTL_FILE package. The UTL_FILE package allows Oracle accounts to read and write files on the host operating system. Access to this package should be restricted.

Check Name: UTL_FILE_DIR Setting

Description: Check that the Oracle parameter UTL_FILE_DIR is not set to * to allow the UTL_FILE package permissions on all directories. The UTL_FILE package allows file I/O from PL/SQL on the client and server side. The client implementation is subject to normal operating system file permission checking and does not need any additional security constraints. The server implementation might be running in a privileged mode and thus will need additional security restrictions that limit the power of this feature.

The parameter specification UTL_FILE_DIR = * has a special meaning. This entry in effect turns off directory access checking and makes any directory accessible to the UTL_FILE functions. The * option should be used with great caution. For security reasons, it is recommended that you not use this option in production systems. Also, do not include '.' (the current directory for Unix) in the accessible directories list.

To ensure security on file systems that allow symbolic links, users must not be allowed WRITE permission to directories accessible by PL/SQL file I/O functions. The symbolic links and PL/SQL file I/O could be used to circumvent normal operating system permission checking and allow users read/write access to directories to which they would not otherwise have access.

Check Name: View Missing With Check Option

Description: Check to see that any views which have been granted UPDATE or INSERT permissions contain the WITH CHECK option at the end of the WHERE clause. Views containing a WHERE clause without the WITH CHECK option allow users to insert or update rows that cannot be seen by the view because they do not meet the selection criteria. Depending on the sensitivity of your data, this may be undesirable behavior. Adding the WITH CHECK option to the WHERE clause will force the WHERE clause to be evaluated upon insert or update and will not allow a user to insert or update rows that would be otherwise be hidden by the view.

Number of System Integrity Checks Enabled:

41