| | | | |
|---|---|---|---|
| References | ♦ Personal Knowledge | Objective / Subjective | Objective |
| Risk | Accounts with blank passwords provide an easy entry point into the database by unauthorized users. The level of exposure, should an account with blank password be exploited, is dependant on the permissions granted to the specific account. The likelihood of the account being found is moderate. | Risk Level | Medium |
| Compliance Criteria | Pass = No user accounts are found with blank passwords.<br>Fail = At least one user account was found with a blank password. | | |
| Audit Procedure | 1. Execute the following command via SQL*Plus:<br>select username, password from dba_users;<br>2. Review the password column to ensure that all users have a password assigned. | | |

| *2.2.1.4 Inactive Accounts* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that database user accounts do not exist even though they are never used. | | |
| References | ♦ Personal Knowledge | Objective / Subjective | Objective |
| Risk | Accounts that are not needed provide an unnecessary entry point to the database. Should an inactive account be exploited, the level of exposure is dependent on the permissions granted to the specific account. The likelihood of an account being exploited is low provided the appropriate password controls are in place. | Risk Level | Low |

| | |
|---|---|
| Compliance Criteria | To comply with this item, a process needs to be in place to review database accounts for inactivity (not used for 60 days or more). In addition, a review of all database accounts must not find any that are enabled and not used for the past 60 days.<br>Pass = Accounts are reviewed (manually or via tools) for inactivity duration. Accounts that have been inactive for 60 days are disabled / deleted.<br>Fail = Inactive accounts are not reviewed.<br>And / Or<br>Fail = Inactive accounts are found enabled. |
| Audit Procedure | 1. Ask the system administrator if there is a process in place to review inactive accounts.<br>2. Review a sample number of accounts for inactivity by checking the access log for each. |

| *2.2.1.5 Shared Accounts* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that user accounts are not shared. | | |
| References | ♦ Personal Knowledge<br>♦ Newallis, p. 7. | Objective / Subjective | Subjective |
| Risk | The use of shared accounts eliminates the ability to track the integrity of information, because there is no way to tell whom the last person to access the data was. | Risk Level | Medium |
| Compliance Criteria | Compliance will depend on interviews with DBAs and users to determine if accounts are being shared, which makes compliance subjective. Overall compliance will be binary based on the following:<br>Pass: Accounts are being shared.<br>Fail: No accounts are shared. | | |

| | |
|---|---|
| Audit Procedure | 1. Obtain a list of all database user accounts by executing the following command within SQL*Plus:<br>*select username from dba_users;*<br>2. Review the output of the above statement for accounts that are not assigned to one particular individual (e.g. helpdesk, training, and test).<br>3. Ask the database administrator if the accounts found in step 2 are used by a group of individuals (e.g. shared password).<br>4. A log review can also be performed to determine if any user accounts are logging in from more than one terminal. Logging in from more than one terminal can be an indication of an account being shared. Obtain a list of accounts that have logged in from more than one terminal by executing the following within SQL*Plus.<br>*select count(distinct(terminal)) Count, username*<br>*from dba_audit_session*<br>*having count(distinct(terminal))>1*<br>*group by username;*<br>5. Review the output from step 4 to identify user accounts that appear to be shared by a number of users.<br>6. Interview the DBA and appropriate users to determine why an account is being used from more than one terminal. |

| *2.2.1.6 “Public” Permissions* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that the permissions granted to the user group “PUBLIC” are appropriate. | | |
| References | ♦ Personal Knowledge<br>♦ Newman, p. 38. | Objective / Subjective | Subjective |
| Risk | Permissions granted to the PUBLIC user group are provided to all database users because all database users are members of the PUBLIC user group. Permissions granted to PUBLIC should be limited to those that all users need to have. It is very likely that the permissions assigned to PUBLIC will be greater than necessary if they are not reviewed. | Risk Level | High |

| | |
|---|---|
| Compliance Criteria | Compliance will depend on the analysis of the permissions / roles / privileges granted to the PUBLIC user group. This review will need to be done in coordination with the DBA to determine what permissions are required of all users.<br><br>Pass: The analysis has concluded that all permissions / roles / privileges granted to the PUBLIC user group are appropriate.<br><br>Fail: PUBLIC has been granted permissions / roles / privileges that are not appropriate for all users to have.<br>. |
| Audit Procedure | 1. Obtain a list of all the objects where permissions have been granted to PUBLIC.<br>*select table_name, privilege from dba_tab_privs where grantee='PUBLIC';*<br>2. Review the output of step 1 with the DBA to determine if the permissions are necessary for the PUBLIC user group.<br>3. Obtain a list of all roles that have been granted to PUBLIC.<br>*select granted_role from dba_role_privs where grantee='PUBLIC';*<br>4. Review the output of step 3 with the DBA to determine if the roles are necessary for the PUBLIC user group.<br>5. Obtain a list of all system privileges that have been granted to PUBLIC.<br>*select privilege from dba_sys_privs where grantee='PUBLIC';*<br>6. Review the output of step 3 with the DBA to determine if the roles are necessary for the PUBLIC user group. |

| *2.2.1.7 Remote OS Authentication* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that authentication via a remote Operating System is not trusted. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 10, 35. | Objective / Subjective | Objective |
| Risk | An attacker could gain access to the database without having to supply a password. If this setting were enabled, the attacker would only need a valid username that is trusted for remote authentication. The likelihood of this happening is low due to the limited number of operating system users and the parameter having been checked during system certification. | Risk Level | Low |

| | |
|---|---|
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: remote_os_authent must be set equal to FALSE.<br>Fail: remote_os_authent is set equal to TRUE. |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the database administrator.<br>2. Perform a find in the file for the parameter "remote_os_authent=".<br>3. Review the value assigned to the parameter to see if it meets the compliance criteria. |

| *2.2.1.8 Password Settings* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that database user accounts have password settings enforced upon them. | | |
| References | ♦ Personal Knowledge<br>♦ Newallis, p. 9.<br>♦ "Oracle Audit Checklist", p. 1.<br>♦ "Oracle Database Security Benchmark v1.0", p. 19-20, 36. | Objective / Subjective | Objective |
| Risk | Users will not follow password guidelines if they are not forced to comply via technical measures. The likelihood that users not change their password appropriately is almost a certainty. | Risk Level | High |

| | |
|---|---|
| Compliance Criteria | Password settings are made at the Profile level within oracle. For each database profile used, review for the following information.<br>To PASS, all of the following must be true:<br>• password_life_time must be < 60: This sets the parameter that controls how often a password must be changed.<br>• password_reuse_max must be > 10: This parameter controls how many times a password must change before it can be reset to a previous password.<br>• password_lock_time > 30: This parameter controls how long an account is locked if the number of failed logins is exceeded.<br>• password_grace_time < 5: This parameter controls the number of days an account can be logged into after the password has expired.<br>• password_verify_function = a valid function (meaning it is not null or unlimited): This parameter sets the function to be used when a password is being changed. The function identified verifies the composition of the password to ensure that weak passwords are not allowed. |

| | |
|---|---|
| Audit Procedure | 1. Obtain a copy of the password_life_time setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name='PASSWORD_LIFE_TIME';*<br>2. Review this setting against the compliance criteria<br>3. Obtain a copy of the password_reuse_max setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name='PASSWORD_REUSE_MAX';*<br>4. Review this setting against the compliance criteria.<br>5. Obtain a copy of the password_lock_time setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name='PASSWORD_LOCK_TIME';*<br>6. Review this setting against the compliance criteria.<br>7. Obtain a copy of the password_grace_time setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name='PASSWORD_GRACE_TIME';*<br>8. Review this setting against the compliance criteria.<br>9. Obtain a copy of the password_verify_function setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name='PASSWORD_VERIFY_FUNCTION';*<br>10. Review this setting against the compliance criteria. |

| *2.2.1.9 Account Lockout* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that database user accounts lockout after a specified number of failed login attempts. | | |
| References | ♦ Personal Knowledge<br>♦ "Oracle Database Security Benchmark v1.0", p. 19 | Objective / Subjective | Objective |
| Risk | Failure to lockout accounts after a number of failed login attempts increases the chance that an account will be compromised due to a brute force password guessing attack. The likelihood of a password guessing attack is moderate. The impact of a compromise of this nature would depend on the permissions granted to the account that is compromised. | Risk Level | High |

| | |
|---|---|
| Compliance Criteria | Account lockout settings are made at the Profile level within Oracle.  For each database profile used, review the failed login attempt parameter to ensure that it meets the following:<br>This parameter can have a range of values assigned to it.  Following SJK policy, the number of attempts shall be limited to 5 or less.  This parameter needs to be verified for all profiles that are assigned to user accounts.<br>PASS = failed_login_attempts ≤ 5<br>FAIL = failed_login_attempts > 5 |
| Audit Procedure | 1. Obtain a copy of the failed_login_attempts setting for each profile by issuing the following statement within SQL*Plus: *select profile, limit from dba_profiles where resource_name=’FAILED_LOGIN_ATTEMPTS’;*<br>2. Review this setting against the compliance criteria.<br>3. In addition to checking the setting, test to ensure that the setting is effective by attempting to login using an invalid password six times in a row.<br>a. First display the account status for an account.<br>b. Next attempt to connect as the account six times.<br>c. Finally, display the account status again to show that the account is now locked. |

### 2.2.2 Network Listener

| *2.2.2.1  Listener Patches* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that all the security patches related to the Oracle TNS Listener have been applied. | | |
| References | ♦ Personal Knowledge<br>♦ “Oracle Database Listener Security Guide”, p. 9-10. | Objective / Subjective | Objective |
| Risk | Failure to apply a security patch to the network listener could leave the system vulnerable to attack.  The likelihood of an attacker attempting to exploit the listener using known vulnerabilities is high. | Risk Level | High |

| | |
|---|---|
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = All relevant patches have been applied.<br>Fail = Not all patches have been applied. |
| Audit Procedure | 1. Obtain a list of all Oracle security patches (from the Oracle web site) related to the network listener.<br>2. Obtain a list of all the Oracle security patches that have been installed on the server from the database administrator.<br>3. Compare the two lists to ensure that all relevant patches have been applied. |

| *2.2.2.2 TNS Listener Password* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that the Listener is password protected. | | |
| References | ♦ "Oracle Database Listener Security Guide", p. 7-9.<br>♦ Newman, p. 11. | Objective / Subjective | Objective |
| Risk | Failure to password protect the Oracle Listener provides anyone who can communicate with it the ability to create a denial of service to the database, as well as obtain detailed configuration information. The likelihood of anyone trying to exploit the listener is low due to the complexity of the attack and the filtering at the firewall. | Risk Level | Medium |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = The TNS Listener requires a password.<br>Fail = A password is not required. | | |

| | |
|---|---|
| Audit Procedure | 1. Obtain a copy of the listener.ora file off of DBSRV1 from the administrator.<br>*cat /u01/app/oracle/product/8.1.7/network/admin/listener.ora*<br>2. Review the file for the parameter "PASSWORDS_LISTENER=".<br>3. Another way to check is to execute the tnscmd perl script and run the services command.<br>*perl tnscmd services --indent -h dbsrv1*<br>4. Review the output of the tnscmd services command to determine if a password has been configured. Output will be limited if a password is required.<br>5. A third way to verify the password requirement is to execute the tnscmd perl script and run the status command.<br>*perl tnscmd status --indent -h dbsrv1*<br>*6.* Review the output of the script for the parameter SECURITY. If the value for this parameter is ON, a password has been set. |

| *2.2.2.3 Listener Admin Restrictions* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that changes to the listener cannot be made dynamically. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 15. | Objective / Subjective | Objective |
| Risk | Without this parameter turned on, changes can be made to the listener settings without having to reload it. Exploiting this would be rather difficult and likelihood low since the listener is required to have a password. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: admin_restrictions_*listener_name* must be set equal to "on".<br>Fail: If admin_restrictions_*listener_name* is set equal to "off". | | |
| Audit Procedure | 1. Obtain a copy of the listener.ora file from the database administrator.<br>2. Perform a find in the file for the parameter "admin_restrictions_*listener_name*=".<br>3. Review the value assigned to the parameter to see if it meets the compliance criteria. | | |

| *2.2.2.4 Listener Audit Settings* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that auditing has been enabled on the Oracle network listener. | | |
| References | ♦ "Oracle Database Listener Security Guide", p. 11,25. | Objective / Subjective | Objective |
| Risk | Failure to enable auditing on the network listener would make it difficult to identify and track down a security incident. | Risk Level | Medium |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = Auditing is enabled and working (the log file has data in it).<br>Fail = Auditing is not enabled. | | |
| Audit Procedure | 1. Obtain a copy of the listener.ora file from the database administrator. This file can be found in the "$ORACLE_HOME/network/admin/" directory.<br>2. Review the file for the "LOG_STATUS=ON" or "LOG_STATUS=OFF".<br>3. Review the contents of the listener log file. The location and name of the listener log file can be found in the listener.ora file.<br>4. Review the information obtained to see if the compliance criteria has been met. | | |

| *2.2.2.5 Unused Listener Services* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that only necessary listener services are installed. | | |
| References | ♦ "Oracle Database Listener Security Guide", p. 13-14.<br>♦ Newman, p. 22.<br>♦ "Oracle Database Security Benchmark v1.0", p. 31. | Objective / Subjective | Subjective |
| Risk | "ExtProc (PLSExtProc) functionality allows external C and Java functions to be called from within PL/SQL." (Oracle Database Security Benchmark v1.0, p. 31). | Risk Level | Low |

| | |
|---|---|
| Compliance Criteria | Compliance of this item is subjective based on the analysis and determination made by the database administrator.<br>Pass:  ExtProc (PLSExtProc) is not started, or it is started and required.<br>Fail:  ExtProc (PLSExtProc) is started, but not required. |
| Audit Procedure | 1. Obtain a copy of the listener.ora file from the database administrator.  This file can be found in the "$ORACLE_HOME/network/admin/" directory.<br>2. Review the file for the services configured by looking for the lines with Service  "ExtProc" or Service "PLSExtProc".<br>3. If either of these lines are found, ask the database administrator if the service is required as they are generally enabled by default. |

| *2.2.2.6  Listener Ports* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that the Oracle TNS listener is not running on the default ports of 1521 or 1526. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 31. | Objective / Subjective | Objective |
| Risk | The default ports are well known by attackers.  Running the listener on a non-default port will make it more difficult for an attacker to determine the location of critical database resources.  The likelihood that an attacker would attempt to connect to the default port is high, but the risk of not changing the value from the default is low because a determined attacker would only be slowed down by this change. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = The listener port is neither 1521 nor 1526.<br>Fail = The listener port is 1521 or 1526. | | |

| | |
|---|---|
| Audit Procedure | 1. Obtain a copy of the listener.ora file off of DBSRV1 from the administrator.<br>*cat /u01/app/oracle/product/8.1.7/network/admin/listener.ora*<br>2. Review the file for the listener port setting.<br>3. Run the tnscmd perl script and run the status command.<br>*perl tnscmd status --indent -h dbsrv1*<br>4. Review the output of the script for the listener port.<br>5. Execute an nmap scan of the database server.<br>*nmap -sS -O -p 1-65535 -v -oN dbsrv1.txt dbsrv1*<br>6. Review the output of the nmap scan to determine if the default listener port (1521) is used. |

### 2.2.3 Tru64

| *2.2.3.1 Tru64 Patches* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that all the Tru64 security related patches have been applied. | | |
| References | ♦ Personal Knowledge | Objective / Subjective | Objective |
| Risk | Failure to install security patches in a timely manner will leave the system exposed to a known vulnerability. The longer a system is exposed to a known vulnerability, the more likely an exploit for that vulnerability will be run against the system. | Risk Level | High |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: All Tru64 security patches have been installed.<br>Fail: At least one Tru64 security patch has not been installed. | | |
| Audit Procedure | 1. Obtain a list of all security patches that have been installed from the system administrator.<br>2. Obtain a list of all Tru64 security patches available from HP for the version of Tru64 installed.<br>3. Compare what is installed on the system to what is available. | | |

| *2.2.3.2 Tru64 Audit Settings* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that auditing is enabled on Tru64 and that at a minimum logon and logoffs are being recorded. | | |
| References | ♦ Personal Knowledge | Objective / Subjective | Subjective |

| | | | |
|---|---|---|---|
| Risk | Failure to record audit data leads to the inability to identify and respond to security incidents. | Risk Level | Medium |
| Compliance Criteria | Compliance with this item is subjective because there is a wide array of potential audit requirements. For the purpose of this audit, the following criteria have been set:<br>Pass: Auditing is enabled for at a minimum logon and logoff attempts. The audit log must be reviewed on a regular basis.<br>Fail: Auditing is not enabled. | | |
| Audit Procedure | 1. Interview the system administrator and inquire if auditing is enabled. Also, ask what the procedures are for monitoring the logs.<br>2. Have the administrator display the current audit log as evidence. | | |

| *2.2.3.3 Oracle Account & Group* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that only authorized personnel have access to the Oracle account and / or are members of the Oracle groups. | | |
| References | ♦ Personal Knowledge<br>♦ Plusnina, p. 8.<br>♦ "Oracle Database Security Benchmark v1.0", p. 26. | Objective / Subjective | Subjective |
| Risk | Having access to the Oracle account, or being a member of the Oracle group provides privileges to access files in the Oracle home directory and therefore sensitive information. | Risk Level | High |
| Compliance Criteria | Determining who has access to the Oracle account and who is a member of the Oracle group is objective. Deciding that they need that level of access is subjective. The following criteria will be used to base compliance for this item:<br>Pass: Only authorized users are members of the Oracle DBA group and only authorized individuals know the Oracle account password.<br>Fail: Any other scenario | | |

|  |  |
|---|---|
| Audit Procedure | 1. Obtain a copy of the Tru64 password file to get a list of user accounts on the system. (*cat /etc/passwd*)<br>2. Obtain a copy of the Tru64 group file from the system administrator. (*cat /etc/group*)<br>3. Review the files with the database administrator to determine that Oracle group membership is appropriate.<br>4. Ask the DBA to tell you who knows the Oracle password.<br>5. Review the people who know the Oracle password and decide if they have a need-to-know. |

<table>
<tr><th colspan="4"><i>2.2.3.4 Database file permissions.</i></th></tr>
<tr><td>Control Objective</td><td colspan="3">This control objective is designed to ensure that the Oracle database files have the appropriate file permissions.<br><br>This should include reviewing the Oracle home directory, Oracle temporary directories and all of their subdirectories and files.</td></tr>
<tr><td>References</td><td>♦ Personal Knowledge<br>♦ Plusnina, p. 7<br>♦ “Oracle Database Listener Security Guide”, p. 13.<br>♦ Newman, p. 11.<br>♦ “Oracle Database Security Benchmark v1.0”, p. 9,13.</td><td>Objective / Subjective</td><td>Subjective</td></tr>
<tr><td>Risk</td><td>Access to Oracle files should be controlled to prevent an unauthorized user from gaining access to sensitive information. Due to the limited number of operating system users, the likelihood of someone gaining access to any Oracle files is limited.</td><td>Risk Level</td><td>Low</td></tr>
<tr><td>Compliance Criteria</td><td colspan="3">Compliance with this item is subjective based on the analysis performed with the system and database administrators.<br><br>Pass: The administrators determine that the Oracle file permissions are appropriate.<br><br>Fail: File permissions are not appropriate.</td></tr>
<tr><td>Audit Procedure</td><td colspan="3">1. Obtain a list of all Oracle related files (showing the file permissions) from the system administrator.<br>2. Review the list of files with the system and database administrators to determine that the permissions are appropriate.</td></tr>
</table>

| *2.2.3.5 Database File Integrity* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that file integrity monitoring is in place for the Oracle installation.<br>This should include reviewing Oracle scripts and executables in the Oracle home directory. | | |
| References | ♦ Personal Knowledge | Objective / Subjective | Objective |
| Risk | Changes to Oracle executables and scripts can be indication that the system has been penetrated and the files replaced with Trojaned versions. The likelihood of this happening is low due to the limited number of users who can log into the operating system. | Risk Level | Low |
| Compliance Criteria | File integrity monitoring needs to be in place, this can be either a manual process or automated tool. The automated tool option is preferred.<br>Pass: File integrity software (e.g. Tripwire) is configured to monitor the integrity of database executables and scripts.<br>Pass: A manual process is in place to monitor the integrity of database executables and scripts on a regular basis.<br>Fail: File integrity monitoring is not being completed. | | |
| Audit Procedure | 1. Interview the database administrator (system administrator) to determine if file integrity is being monitored, and if so is being completed through a manual process or an automated tool.<br>2. Obtain documented procedures from the DBA that describes the process / tool being used and which database files are being monitored. | | |

### 2.2.4 Oracle

| *2.2.4.1 Oracle Patches* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that Oracle related security patches are installed on the system. | | |
| References | ♦ Personal Knowledge<br>♦ “Oracle Database Security Benchmark v1.0”, p. 7. | Objective / Subjective | Objective |

| | | | |
|---|---|---|---|
| Risk | Failure to install security patches in a timely manner will leave the system exposed to a known vulnerability. The longer a system is exposed to a known vulnerability, the more likely an exploit for that vulnerability will be run against the system. | Risk Level | High |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: All relevant Oracle security patches have been installed.<br>Fail: At least one relevant Oracle security patch has not been installed. | | |
| Audit Procedure | 1. Obtain a list of all security patches that have been installed from the database administrator.<br>2. Obtain a list of all Oracle security patches available that are relevant to Oracle version 8.1.7.<br>3. Compare what is installed on the system to what is available. | | |

| | | | |
|---|---|---|---|
| *2.2.4.2 Oracle Audit Settings* | | | |
| Control Objective | This control objective is designed to ensure that Oracle auditing has been enabled for a predetermined set of events. | | |
| References | ♦ Personal Knowledge<br>♦ Finnigan, Pete. “Introduction to Simple Oracle Auditing”, p. 4 | Objective / Subjective | Objective |
| Risk | Failure to record audit data leads to the inability to identify and respond to security incidents. | Risk Level | Medium |

| | |
|---|---|
| Compliance Criteria | Check the audit settings to determine that the following are being audited:<br>• ALTER USER<br>• Any CREATE statement.<br>• Any DROP statement.<br>• GRANT ANY PRIVILEGE<br>• GRANT ANY ROLE<br>• INSERT failures<br>• LOGON and LOGOFF<br><br>Pass: Auditing is enabled and configured to record at a minimum the above events.<br><br>Fail: Auditing is not enabled, or does not record the minimum events above. |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the DBA.<br>2. Review the file for the parameter, audit_trail =, to determine if auditing is enabled.<br>3. Another check to see if auditing is enabled would be to execute the following within SQL*Plus:<br>*select name, value from v$parameter where name like 'audit%';*<br>4. Next, check to see what privilege audit options are enabled by executing the following within SQL*Plus:<br>*select * from dba_priv_audit_opts;*<br>5. Check to see what statement audit options are enabled by executing the following within SQL*Plus:<br>*select * from dba_stmt_audit_opts;* |

<table>
<tr><th colspan="4"><i>2.2.4.3 Database Link Settings</i></th></tr>
<tr><td>Control Objective</td><td colspan="3">This control objective is designed to ensure that database links do not have usernames and passwords stored.</td></tr>
<tr><td>References</td><td>♦ Personal Knowledge<br>♦ “Oracle Audit Checklist”, p. 1.<br>♦ “Oracle Database Security Benchmark v1.0”, p. 24.<br>♦ Finnigan, “Exploiting and Protecting Oracle”, p. 11.</td><td>Objective / Subjective</td><td>Objective</td></tr>
</table>

| | | | |
|---|---|---|---|
| Risk | Database links that store the username and password in the database do so in clear text. Should users gain access to the table that maintains this information, the information in the database being linked to could be compromised. The level of severity depends on the user account privileges associated with the account compromised. | Risk Level | Medium |
| Compliance Criteria | Compliance with this item is objective based on a review of all database links.<br>Pass: There are no database links that have usernames and passwords hardcoded.<br>Fail: At least one database link has a hardcoded username and password. | | |
| Audit Procedure | 1. Obtain a list of all database links by executing the following within SQL*Plus:<br>*select * from all_db_links;*<br>2. Review the output of this statement to determine if any hardcoded usernames and passwords are found. | | |

| *2.2.4.4 Trace Files* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that "users do not have the ability to read trace files". (Oracle Database Security Benchmark v1.0, p. 9) | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 9. | Objective / Subjective | Objective |
| Risk | Public access to trace files could reveal sensitive information to people who do not have a need to know. The likelihood of this happening is low due to the limited number of people who have access to the database outside of the application interfaces and host operating system. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: _trace_files_public must be set equal to FALSE.<br>Fail: _trace_files_public is set equal to TRUE. | | |

| | |
|---|---|
| Audit Procedure | 1. Obtain a copy of the init.ora file from the database administrator.<br>2. Perform a find in the file for the parameter “_trace_files_public=”.<br>3. Review the value assigned to the parameter to see if it meets the compliance criteria. |

| *2.2.4.5 SQL92 Security* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that SELECT privileges are required to execute an update or delete on table values. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 12. | Objective / Subjective | Objective |
| Risk | “This parameter will enforce the requirement that a user must have SELECT privilege on a table in order to be able to execute UPDATE and DELETE statements using WHERE clauses on a given table.” (Oracle Database Security Benchmark v1.0, p. 12) Users could gain access to information they don’t have a need to know should this setting not be set. Users are not likely to take advantage should this not be configured appropriately. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass: sql92_security must be set equal to TRUE.<br>Fail: sql92_security is set equal to FALSE. | | |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the database administrator.<br>2. Perform a find in the file for the parameter “sql92_security=”.<br>3. Review the value assigned to the parameter to see if it meets the compliance criteria. | | |

| *2.2.4.6 Views* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that access to sensitive views is appropriate. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 21-22,37. | Objective / Subjective | Subjective |

| | | | |
|---|---|---|---|
| Risk | There are a number of database views that contain sensitive information. Users who have permission to access these views can use this information in an attempt to compromise application data. The likelihood of this happening is low due to the restriction on users to only access the database from within applications. | Risk Level | Medium |
| Compliance Criteria | Review the permissions that users have to the following database views:<br>• DBA_% ; this represents a wildcard for all views that begin with DBA_.<br>• V$_% ; this represents a wildcard for all views that begin with V$_.<br>• ALL_% ; this represents a wildcard for all views that begin with ALL_.<br>• ROLE_ROLE_PRIVS<br>• USER_TAB_PRIVS<br>• USER_ROLE_PRIVS<br>Compliance for this item is subjective based on the different views and who the database administrator feels should have access to them.<br>Pass: The permissions set on all sensitive views listed above are appropriate.<br>Fail: The permissions are not appropriate on all sensitive views listed above. | | |
| Audit Procedure | 1. Execute the following SQL*Plus statement:<br>*select grantee, privilege, table_name from dba_tab_privs where (owner=’SYS’ or table_name like ’DBA_%’ or table_name like ’V$_%’ or table_name like ’ALL_%’ or table_name=‘ROLE_ROLE_PRIVS’ or table_name=‘USER_TAB_PRIVS’ or table_name=‘USER_ROLE_PRIVS’);*<br>2. Review the output from step 1 with the DBA to determine if access to all the views are appropriate. | | |

| *2.2.4.7 With Admin* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that privileges and roles have not been granted with the administrator option turned on. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 23. | Objective / Subjective | Subjective |

| | | | |
|---|---|---|---|
| Risk | Granting privileges / roles with the admin option set allows the user to act as an administrator for that privilege / role. The likelihood of a user taking advantage of this ability is minimized due to the restriction of accessing the database via the application. | Risk Level | Medium |
| Compliance Criteria | Compliance with this item is subjective because certain users may require this capability and the identification of these users is at the discretion of the database administrator.<br>Pass: All users who have been granted privileges / roles with admin option are appropriate.<br>Fail: Users have been granted privileges / roles with admin option inappropriately. | | |
| Audit Procedure | 1. Obtain a list of all users that have been granted privileges with the admin option set by executing the following within SQL*Plus:<br>*select grantee, privilege from dba_sys_privs where grantee not in ('SYS', 'SYSTEM', 'DBA') and admin_option='YES';*<br>2. Review the list with the database administrator to verify whether all the users require the privilege with admin option set.<br>3. Obtain a list of all users that have been granted roles with the admin option set by executing the following within SQL*Plus:<br>*select grantee, granted_role from dba_role_privs where grantee not in ('SYS', 'SYSTEM', 'DBA') and admin_option='YES';*<br>4. Review the list with the database administrator to verify whether all the users require the role with admin option set. | | |

| *2.2.4.8 With Grant Privileges* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that privileges have not been granted with the with grant option enabled. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 23. | Objective / Subjective | Subjective |
| Risk | Users that have privileges to objects with the "with grant" option set are able to grant access to those same objects to other users. This risk is minimized by the inability of users to access the database outside of the application interface. | Risk Level | Medium |

| | |
|---|---|
| Compliance Criteria | Compliance with this item is subjective because certain users may require this capability and the identification of these users is at the discretion of the database administrator.<br>Pass: All users who have been granted privileges with grant option are appropriate.<br>Fail: Users have been granted privileges with grant option inappropriately. |
| Audit Procedure | 1. Obtain a list of all users that have been granted privileges with the with grant option set by executing the following within SQL*Plus:<br>*select owner, grantee, table_name from dba_tab_privs where grantable='YES' and owner not in ('SYS', 'SYSTEM', 'DBA') order by grantee;*<br>2. Review the list with the database administrator to verify whether all the users require the privilege with admin option set. |

| *2.2.4.9 Select Any Table Privilege* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that users do not have the ability to SELECT ANY TABLE. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 23. | Objective / Subjective | Objective |
| Risk | Users with the privilege to SELECT ANY TABLE can view all the data in any table within the database and essentially bypass the permissions granted to them via roles. The likelihood that a user would attempt to select data from tables the don't have access to is low due to the fact that users are restricted from accessing the database via tools such as SQL*PLUS. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting for each user account. Each user account either has this privilege or it doesn't so a simple review of all user accounts will provide the information necessary to assess this item.<br>PASS = No users have been granted the privilege SELECT ANY TABLE.<br>FAIL = Users have been granted the privilege SELECT ANY TABLE. | | |

| | |
|---|---|
| Audit Procedure | 1. Obtain a list of all users who have the privilege assigned to them by submitting the following from within SQL*PLUS.<br>*select * from dba_sys_privs where privilege=’SELECT ANY TABLE’;*<br>2. Review the output of this command to determine if the compliance criteria is met. |

| *2.2.4.10 Audit System Privilege* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that only authorized users have audit privileges. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 23. | Objective / Subjective | Subjective |
| Risk | As the audit trail has sensitive information in it, a user that has audit privileges may gain access to information they don’t have a need-to-know. It is not likely that a user would use these privileges due to a lack of knowledge and the restriction of connecting through the application interface. | Risk Level | Low |
| Compliance Criteria | Determining whether a person has audit privileges or not is objective. Deciding that the person needs audit privileges is subjective. This item is therefore subjective based on the decision of the DBA as to who can have audit privileges.<br>Pass: All users with audit privileges are appropriate.<br>Fail: Users have audit privileges inappropriately. | | |
| Audit Procedure | 1. Execute the following statement within SQL*Plus:<br>*select * from dba_sys_privs where privilege like '%AUDIT%';*<br>2. Review the output of step 1 with the DBA to decide if the privileges are appropriate. | | |

| *2.2.4.11 Package Access* | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that the access to packages that provide additional capabilities have not been granted to PUBLIC. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 24. | Objective / Subjective | Subjective |

| | | | |
|---|---|---|---|
| Risk | Access to the packages provides users with additional privileges that may or may not be necessary for their job function. Unnecessary access to these packages provides privileges that are not required and could be exploited. Use of these packages is complex and the ability to work outside of the application interface is limited, therefore the likelihood of exploitation is low. | Risk Level | Medium |
| Compliance Criteria | Compliance for this item is objective based on the PUBLIC user group being granted access to the following packages.<br>• UTL_FILE<br>• UTL_TCP<br>• UTL_HTTP<br>• UTL_SMTP<br>• DBMS_LOB<br>• DBMS_SYS_SQL<br>• DBMS_JOB<br>Pass: None of the packages have been granted to the PUBLIC user group.<br>Fail: At least one of the packages has been granted to the PUBLIC user group. | | |
| Audit Procedure | 1. Execute the following SQL*Plus statement to find packages that have been granted to PUBLIC with execute privileges.<br>*select table_name from dba_tab_privs where grantee='PUBLIC' and privilege='EXECUTE' and table_name in ('UTL_FILE','UTL_TCP','UTL_HTTP','UTL_SMTP','DBMS_LOB','DBMS_SYS_SQL','DBMS_JOB');*<br>2. Review the results of step 1, if any are provided then this item fails compliance. | | |

| | | | |
|---|---|---|---|
| *2.2.4.12 Data Dictionary* | | | |
| Control Objective | This control objective is designed to ensure that users with the privilege SELECT ANY TABLE cannot access the data dictionary. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 30. | Objective / Subjective | Objective |

| | | | |
|---|---|---|---|
| Risk | The data dictionary contains sensitive information about the database and should not be accessible by users. It is not very likely that users have the privilege to SELECT ANY TABLE, however this control will reduce the risk should a user get this privilege. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>PASS = The parameter O7_dictionary_accessibility is set to FALSE.<br>FAIL = The parameter O7_dictionary_accessibility is set to TRUE. | | |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the database administrator.<br>2. Review this file for value assigned to the parameter O7_dictionary_accessibility. | | |

# 3 Audit Evidence

Using the customized audit checklist, it was time to perform the audit of DBSRV1. Because I (the independent auditor) didn't have access to the system being audited, completing the audit required the involvement of the database administrator and the Tru64 system administrator. These administrators reviewed all the commands / scripts and then executed them on my behalf, providing the results in a text file. In order to demonstrate the execution of the audit, I selected ten checklist items to provide detailed audit results.

## *3.1 Audit Results*

The results of the ten checklist items include the checklist item details with the additional audit evidence information (results of the commands / tests), a pass / fail determination based on the compliance criteria, and reference information to the discussion of the item in the audit report.

### 3.1.1 Default Account / Passwords

| Reference: Checklist #: 2.2.1.2 Default Accounts / Passwords on page 19<br>Analysis in final report on page 73. | |
|---|---|
| Control Objective | This control objective is designed to ensure that all of the default accounts have been disabled / deleted if the account is not necessary, and that the password has been changed on all default accounts that are necessary. |

| | | | |
|---|---|---|---|
| References | ♦ Personal Knowledge<br>♦ Plusnina, p. 9.<br>♦ Newallis, p. 7.<br>♦ “Oracle Audit Checklist”, p. 1.<br>♦ Newman, p. 38.<br>♦ “Oracle Database Security Benchmark v1.0”, p. 35. | Objective / Subjective | Objective |
| Risk | Leaving default accounts active and with default passwords provides an easy entry point for unauthorized access. It is highly likely that authorized users and attackers will attempt to log in to the database using the vendor supplied default accounts. | Risk Level | High |
| Compliance Criteria | A review of the database user accounts will determine if any vendor supplied accounts are present. Once the list of vendor supplied accounts present has been determined, they can be reviewed to determine if the account is necessary, if it has been disabled, and if the password has been changed.<br>Pass = If vendor supplied default accounts exist, they are either disabled, or do not have default passwords.<br>Fail = Multiple scenarios. Examples below:<br>1. A vendor supplied default account is enabled, and the password is still set to the default.<br>2. If the default account is enabled and it is not necessary. | | |
| Audit Procedure | 1. Obtain a list of all database user accounts by executing the following command via SQL*Plus:<br>*select username, password, profile from dba_users;*<br>2. Review the list of accounts to determine if any are vendor supplied.<br>3. For the vendor supplied accounts found in step 2, attempt to connect to the database with the default password (usually equals username). An extensive list of default username / password combinations can be found at: <http://www.cirt.net/cgi-bin/passwd.pl?method=showven&ven=Oracle>. Successful connections mean the default password has not been changed. | | |
| Audit Results | 1. Output from step 1 in the audit procedures, scrubbed for sensitive information. | | |

User Accounts, passwords, profile, and owner on prd

```
Username        PASSWORD                 PROFILE        OWNER
--------------- ------------------------ ---------- --------------------
ADMN            6376D30E4EB575A4         DEFAULT
AJJJEEE         3EF9038440AB16AC         DEFAULT
ARRRRRR         5C24E1D83817511A         DEFAULT
ASSSSSSS        EFB49CC14DE72FC6         DEFAULT
AZZZZZZ         SQLCQR_DORMANT           DEFAULT
AUUUUU          97671BF1B68B8D27         DEFAULT
AVVVVVV         3F155D06629E7FB3         DEFAULT
BBBBB           98B716B9C56CCF0F         DEFAULT
CCC             6680F90D9146233F         DEFAULT
CCD             E90B20F9A44CFDCC         DEFAULT
CCE             E70CE40EAC4D28E6         DEFAULT
CCF             SQLCQR_DORMANT           DEFAULT
CCG             0BDF7ED9F2121522         DEFAULT
CCH             SQLCQR_DORMANT           DEFAULT
DBSNMP          4B206F502565E28D         DEFAULT
DDD             SQLCQR_DORMANT           DEFAULT
DDE             SQLCQR_DORMANT           DEFAULT
DDF             6A21EFAA1B16DCA1         DEFAULT
DDG             B07CED2B5242ECF7         DEFAULT
EEE             71BA501619B5A0C4         DEFAULT
EEG             D97C9B88DAF00843         DEFAULT
EEF             1CED4754B7751750         DEFAULT
EEC             647B48A0CA8F13E6         DEFAULT
EXX             432DE0E7F06E730C         DEFAULT
FFF             E6F19F9975978204         DEFAULT
FFB             08AF9455773E4AA1         DEFAULT
FED             A19B4FC124764887         DEFAULT
FTE             8CA985B8A4DBE1ED         DEFAULT
FAAAA           1DB08E3D9B25405C         DEFAULT
III             D81479E6F9FB2519         DEFAULT
IIIIII          SQLCQR_DORMANT           DEFAULT
IKKK            1CB4E995DE15C2D9         DEFAULT
ILLLLL          E2C0B4E91A55E62D         DEFAULT
IPPPPP          C50CC9CEA77D0E27         DEFAULT
ISSSSS          F82503F31B82A19B         DEFAULT
IYYYYY          0BB7B9963E0B3CC4         DEFAULT
IUUUUUU         SQLCQR_DORMANT           DEFAULT
OUTLN           C3B2A1D4C3B2A1D4         DEFAULT
SA              SQLCQR_DORMANT           DEFAULT
SAAAAAAC        C3B2A1D4C3B2A1D4         DEFAULT
SCC             SQLCQR_DORMANT           DEFAULT
SCO             C3B2A1D4C3B2A1D4         DEFAULT
SQLCQR          C3B2A1D4C3B2A1D4         DEFAULT
SQLCQR_VSM_AGENT         C3B2A1D4C3B2A1D4   DEFAULT
SYS             C3B2A1D4C3B2A1D4         DEFAULT
SYSTEM          C3B2A1D4C3B2A1D4         DEFAULT
```

2. A review of the accounts reveals that the following are vendor supplied:
   - DBSNMP
   - OUTLN

| | |
|---|---|
| | • SYS<br>• SYSTEM<br>3. A connection attempt was made to determine if the default username password combination was still set. The following was found:<br>• DBSNMP / DBSNMP = Did not work.<br>• OUTLN / OUTLN = Got connected.<br>• SYS / CHANGE_ON_INSTALL = Did not work.<br>• SYSTEM / MANAGER = Did not work. |
| Compliance Determination | Fail ✗ |

### 3.1.2 TNS Listener Password

| | | | |
|---|---|---|---|
| Reference: Checklist #: 2.2.2.2 TNS Listener Password on page 29 | | | |
| Control Objective | This control objective is designed to ensure that the Listener is password protected. | | |
| References | ♦ "Oracle Database Listener Security Guide", p. 7-9.<br>♦ Newman, p. 11. | Objective / Subjective | Objective |
| Risk | Failure to password protect the Oracle Listener provides anyone who can communicate with it the ability to create a denial of service to the database, as well as obtain detailed configuration information. The likelihood of anyone trying to exploit the listener is low due to the complexity of the attack and the filtering at the firewall. | Risk Level | Medium |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = The TNS Listener requires a password.<br>Fail = A password is not required. | | |

|  |  |
|---|---|
| Audit Procedure | 1. Obtain a copy of the listener.ora file off of DBSRV1 from the administrator.<br>*cat /u01/app/oracle/product/8.1.7/network/admin/listener.ora*<br>2. Review the file for the parameter “PASSWORDS_LISTENER=”.<br>3. Another way to check is to execute the tnscmd perl script and run the services command.<br>*perl tnscmd services --indent -h dbsrv1*<br>4. Review the output of the tnscmd services command to determine if a password has been configured.  Output will be limited if a password is required.<br>5. A third way to verify the password requirement is to execute the tnscmd perl script and run the status command.<br>*perl tnscmd status --indent -h dbsrv1*<br>*6.* Review the output of the script for the parameter SECURITY.  If the value for this parameter is ON, a password has been set. |
| Audit Results | 1. Copy of the listener.ora file.<br># LISTENER.ORA Network Configuration File: /u01/app/oracle/product/8.1.7/network/admin/listener.ora<br># Generated by Oracle configuration tools.<br><br>LISTENER =<br>  (DESCRIPTION_LIST =<br>    (DESCRIPTION =<br>      (ADDRESS_LIST =<br>        (ADDRESS = (PROTOCOL = TCP)(HOST = dbsrv1)(PORT = 1521))<br>      )<br>    )<br>  )<br><br>SID_LIST_LISTENER =<br>  (SID_LIST =<br>    (SID_DESC =<br>      (GLOBAL_DBNAME = prd)<br>      (ORACLE_HOME = /u01/app/oracle/product/8.1.7)<br>      (SID_NAME = prd)<br>    )<br>  )<br><br>PASSWORDS_LISTENER=(k3h8vy6w)<br>2. A review of the listener.ora file reveals that a password has been set.<br>3. Output from the tnscmd command.<br>perl tnscmd services -h dbsrv1<br>sending (CONNECT_DATA=(COMMAND=services)) to dbsrv1:1521<br>writing 91 bytes<br>reading<br>.e......"..Y(DESCRIPTION=(TMP=)(VSNNUM=135295744)(ERR=1169)(ERROR_STACK=(ERROR=(CODE=1169)(EMFI=4))))<br>4. Review of the output from the tnscmd command reveals that a password has been applied to the listener because |

we received error code 1169. Error code 1169 is for invalid password, and since we didn't supply a password, we can deduce that a password has been applied to the listener.

5. Output from the tnscmd status command.

```
Perl tnscmd status --indent -h dbsrv1
sending (CONNECT_DATA=(COMMAND=status)) to dbsrv1:1521
writing 89 bytes
reading
. .......6.........a. ...........k........
  DESCRIPTION=
    TMP=
    VSNNUM=135295744
    ERR=0
    ALIAS=LISTENER
    SECURITY=ON
    VERSION=TNSLSNR for DEC OSF/1 AXP: Version 8.1.7.3.0 -
Production
    START_DATE=15-OCT-2003 23:47:05
    SIDNUM=1
    LOGFILE=/u01/app/oracle/product/8.1.7/network/log/listener.log
    PRMFILE=/u01/app/oracle/product/8.1.7/network/admin/listener.ora
    TRACING=off
    UPTIME=2974532
    SNMP=OFF

."........
  ENDPOINT=
    HANDLER=
      STA=ready
      HANDLER_MAXLOAD=0
      HANDLER_LOAD=0
      ESTABLISHED=0
      REFUSED=0
      HANDLER_ID=C9C868DC8123-5CE8-E030-348CFDA7C274
      PRE=ttc
      SESSION=NS
      DESCRIPTION=
        ADDRESS=
          PROTOCOL=tcp
          HOST=dbsrv1
          PORT=1521
,,
  ENDPOINT=
    HANDLER=
      STA=ready
      HANDLER_MAXLOAD=0
      HANDLER_LOAD=0
      ESTABLISHED=0
      REFUSED=0
      HANDLER_ID=C9C868DC8124-5CE8-E030-348CFDA7C274
      PRE=ttc
      SESSION=NS
      DESCRIPTION=
        ADDRESS=
          PROTOCOL=ipc
```

<table>
<tr><td></td><td><pre>
      KEY=EXTPROC
,,
 SERVICE=
   SERVICE_NAME=prd
   INSTANCE=
     INSTANCE_NAME=prd
     NUM=2
     INSTANCE_CLASS=ORACLE
     NUMREL=2
,
 SERVICE=
   SERVICE_NAME=prd05
   INSTANCE=
     INSTANCE_NAME=prd05
     NUM=2
     INSTANCE_CLASS=ORACLE
     NUMREL=2

,,.........@
</pre>
6. A review the output of the tnscmd status command reveals that the parameter SECURITY is set to ON. This means that a password has been set.</td></tr>
<tr><td>Compliance Determination</td><td>Pass P</td></tr>
</table>

### 3.1.3 Data Dictionary

| Reference: Checklist #: 2.2.4.12 Data Dictionary on page 45. | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that users with the privilege SELECT ANY TABLE cannot access the data dictionary. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 30. | Objective / Subjective | Objective |
| Risk | The data dictionary contains sensitive information about the database and should not be accessible by users. It is not very likely that users have the privilege to SELECT ANY TABLE, however this control will reduce the risk should a user get this privilege. | Risk Level | Low |

| | |
|---|---|
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>PASS = The parameter O7_dictionary_accessibility is set to FALSE.<br>FAIL = The parameter O7_dictionary_accessibility is set to TRUE. |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the database administrator.<br>2. Review this file for value assigned to the parameter O7_dictionary_accessibility. |
| Audit Results | 1. A copy of the init.ora file can be found in Appendix A.<br>2. A review of the init.ora file reveals that the O7_DICTIONARY_ACCESSIBILITY parameter has been set to FALSE. |
| Compliance Determination | Pass P |

### 3.1.4 Account Lockout

| | | | |
|---|---|---|---|
| Reference: Checklist #: 2.2.1.9 Account Lockout on page 27 | | | |
| Control Objective | This control objective is designed to ensure that database user accounts lockout after a specified number of failed login attempts. | | |
| References | ♦ Personal Knowledge<br>♦ "Oracle Database Security Benchmark v1.0", p. 19 | Objective / Subjective | Objective |
| Risk | Failure to lockout accounts after a number of failed login attempts increases the chance that an account will be compromised due to a brute force password guessing attack. The likelihood of a password guessing attack is moderate. The impact of a compromise of this nature would depend on the permissions granted to the account that is compromised. | Risk Level | High |

| | |
|---|---|
| Compliance Criteria | Account lockout settings are made at the Profile level within Oracle. For each database profile used, review the failed login attempt parameter to ensure that it meets the following:<br>This parameter can have a range of values assigned to it. Following SJK policy, the number of attempts shall be limited to 5 or less. This parameter needs to be verified for all profiles that are assigned to user accounts.<br>PASS = failed_login_attempts ≤ 5<br>FAIL = failed_login_attempts > 5 |
| Audit Procedure | 1. Obtain a copy of the failed_login_attempts setting for each profile by issuing the following statement within SQL*Plus:<br>*select profile, limit from dba_profiles where resource_name=’FAILED_LOGIN_ATTEMPTS’;*<br>2. Review this setting against the compliance criteria.<br>3. In addition to checking the setting, test to ensure that the setting is effective by attempting to login using an invalid password six times in a row.<br>a. First display the account status for an account.<br>b. Next attempt to connect as the account six times.<br>c. Finally, display the account status again to show that the account is now locked. |

| | |
|---|---|
| Audit Results | 1. Output from the SQL*Plus statement:<br>PROFILE LIMIT<br>------------------------------ ----------------------------------------<br>DEFAULT 3<br><br>2. This setting meets the compliance critteria.<br><br>3.a.<br>select username, account_status from dba_users where username='SCOTT';<br>USERNAME ACCOUNT_STATUS<br>------------------------------ --------------------------------<br>SCOTT OPEN<br><br>3.b. Attempt to login six times with a bad password.<br>Connect scott/garbage@prd<br>ERROR:<br>ORA-01017: invalid username/password; logon denied<br>Repeat six times<br><br>3.c.<br>select username, account_status from dba_users where username='SCOTT';<br>USERNAME ACCOUNT_STATUS<br>------------------------------ --------------------------------<br>SCOTT LOCKED |
| Compliance Determination | Pass P |

### 3.1.5 Oracle Audit Settings

| Reference: Checklist #: 2.2.4.2 Oracle Audit Settings on page 37.<br>Analysis in final report on page 79. | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that Oracle auditing has been enabled for a predetermined set of events. | | |
| References | ♦ Personal Knowledge<br>♦ Finnigan, Pete. “Introduction to Simple Oracle Auditing”, p. 4 | Objective / Subjective | Objective |
| Risk | Failure to record audit data leads to the inability to identify and respond to security incidents. | Risk Level | Medium |

| | |
|---|---|
| Compliance Criteria | Check the audit settings to determine that the following are being audited:<br>- ALTER USER<br>- Any CREATE statement.<br>- Any DROP statement.<br>- GRANT ANY PRIVILEGE<br>- GRANT ANY ROLE<br>- INSERT failures<br>- CREATE SESSION<br><br>Pass: Auditing is enabled and configured to record at a minimum the above events.<br><br>Fail: Auditing is not enabled, or does not record the minimum events above. |
| Audit Procedure | 1. Obtain a copy of the init.ora file from the DBA.<br>2. Review the file for the parameter, audit_trail =, to determine if auditing is enabled.<br>3. Another check to see if auditing is enabled would be to execute the following within SQL*Plus:<br>*select name, value from v$parameter where name like 'audit%';*<br>4. Next, check to see what privilege audit options are enabled by executing the following within SQL*Plus:<br>*select * from dba_priv_audit_opts;*<br>5. Check to see what statement audit options are enabled by executing the following within SQL*Plus:<br>*select * from dba_stmt_audit_opts;* |

| | |
|---|---|
| Audit Results | 1. A copy of the init.ora file can be found in Appendix A.<br>2. Review of the file reveals that the audit_trial is set to true.<br>3. Output from SQL*Plus<br>NAME VALUE<br>-------------------- --------------------<br>audit_sys_operations FALSE<br>audit_file_dest ?/rdbms/audit<br>audit_trail TRUE<br>4. Output from SQL*Plus<br>USER_NAME PROXY_NAME<br>------------------------------ ------------------------------<br>PRIVILEGE SUCCESS FAILURE<br>---------------------------------------- ---------- ----------<br>CREATE SESSION BY ACCESS BY ACCESS<br>5. Output from SQL*Plus<br>USER_NAME PROXY_NAME<br>------------------------------ ------------------------------<br>PRIVILEGE SUCCESS FAILURE<br>---------------------------------------- ---------- ----------<br>CREATE SESSION BY ACCESS BY ACCESS |
| Compliance Determination | Fail ✗ |

### 3.1.6 With Admin Privileges

| Reference: Checklist #: 2.2.4.7 With Admin Privileges on page 41. | | | |
|---|---|---|---|
| Control Objective | This control objective is designed to ensure that privileges and roles have not been granted with the administrator option turned on. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 23. | Objective / Subjective | Subjective |
| Risk | Granting privileges / roles with the admin option set allows the user to act as an administrator for that privilege / role. The likelihood of a user taking advantage of this ability is minimized due to the restriction of accessing the database via the application. | Risk Level | Medium |

| | |
|---|---|
| Compliance Criteria | Compliance with this item is subjective because certain users may require this capability and the identification of these users is at the discretion of the database administrator.<br>Pass: All users who have been granted privileges / roles with admin option are appropriate.<br>Fail: Users have been granted privileges / roles with admin option inappropriately. |
| Audit Procedure | 1. Obtain a list of all users that have been granted privileges with the admin option set by executing the following within SQL*Plus:<br>*select grantee, privilege from dba_sys_privs where grantee not in (‘SYS’, ‘SYSTEM’, ‘DBA’) and admin_option=’YES’;*<br>2. Review the list with the database administrator to verify whether all the users require the privilege with admin option set.<br>3. Obtain a list of all users that have been granted roles with the admin option set by executing the following within SQL*Plus:<br>*select grantee, granted_role from dba_role_privs where grantee not in (‘SYS’, ‘SYSTEM’, ‘DBA’) and admin_option=’YES’;*<br>4. Review the list with the database administrator to verify whether all the users require the role with admin option set. |
| Audit Results | 1. The output from SQL*Plus revealed that no users were granted privileges with the admin option set.<br>2. Nothing to review.<br>3. The output from SQL*Plus revealed that no users were granted roles with the admin option set.<br>4. Nothing to review. |
| Compliance Determination | Pass P |

### 3.1.7 Select Any Table Privilege

| | | | |
|---|---|---|---|
| Reference: Checklist #: 2.2.4.9 Select Any Table Privilege on page 43. | | | |
| Control Objective | This control objective is designed to ensure that users do not have the ability to SELECT ANY TABLE. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 23. | Objective / Subjective | Objective |

| | | | |
|---|---|---|---|
| Risk | Users with the privilege to SELECT ANY TABLE can view all the data in any table within the database and essentially bypass the permissions granted to them via roles. The likelihood that a user would attempt to select data from tables the don't have access to is low due to the fact that users are restricted from accessing the database via tools such as SQL*PLUS. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting for each user account. Each user account either has this privilege or it doesn't so a simple review of all user accounts will provide the information necessary to assess this item.<br>PASS = No users have been granted the privilege SELECT ANY TABLE.<br>FAIL = Users have been granted the privilege SELECT ANY TABLE. | | |
| Audit Procedure | 1. Obtain a list of all users who have the privilege assigned to them by submitting the following from within SQL*PLUS.<br>*select * from dba_sys_privs where privilege='SELECT ANY TABLE';*<br>2. Review the output of this command to determine if the compliance criteria is met. | | |
| Audit Results | 1. Output from SQL*Plus<br>No rows selected. | | |
| Compliance Determination | Pass P | | |

### 3.1.8 Listener Ports

| | | | |
|---|---|---|---|
| Reference: Checklist #: 2.2.2.6 Listener Ports on page 32.<br>Analysis in final report on page 77. | | | |
| Control Objective | This control objective is designed to ensure that the Oracle TNS listener is not running on the default ports of 1521 or 1526. | | |
| References | ♦ "Oracle Database Security Benchmark v1.0", p. 31. | Objective / Subjective | Objective |

| | | | |
|---|---|---|---|
| Risk | The default ports are well known by attackers.  Running the listener on a non-default port will make it more difficult for an attacker to determine the location of critical database resources.  The likelihood that an attacker would attempt to connect to the default port is high, but the risk of not changing the value from the default is low because a determined attacker would only be slowed down by this change. | Risk Level | Low |
| Compliance Criteria | This item either passes or fails compliance based on a binary setting of this parameter.<br>Pass = The listener port is neither 1521 or 1526.<br>Fail = The listener port is 1521 or 1526. | | |
| Audit Procedure | 1. Obtain a copy of the listener.ora file off of DBSRV1 from the administrator.<br>*cat /u01/app/oracle/product/8.1.7/network/admin/listener.ora*<br>2. Review the file for the listener port setting.<br>3. Run the tnscmd perl script and run the status command.<br>*perl tnscmd status --indent -h dbsrv1*<br>4. Review the output of the script for the listener port.<br>5. Execute an nmap scan of the database server.<br>*nmap -sS -O -p 1-65535 -v -oN dbsrv1.txt dbsrv1*<br>6. Review the output of the nmap scan to determine if the default listener port (1521) is used. | | |
| Audit Results | 1. Copy of the listener.ora file.<br># LISTENER.ORA Network Configuration File: /u01/app/oracle/product/8.1.7/network/admin/listener.ora<br># Generated by Oracle configuration tools.<br><br>LISTENER =<br>  (DESCRIPTION_LIST =<br>    (DESCRIPTION =<br>      (ADDRESS_LIST =<br>        (ADDRESS = (PROTOCOL = TCP)(HOST = dbsrv1)(PORT = 1521))<br>      )<br>    )<br>  )<br><br>SID_LIST_LISTENER =<br>  (SID_LIST =<br>    (SID_DESC =<br>      (GLOBAL_DBNAME = prd)<br>      (ORACLE_HOME = /u01/app/oracle/product/8.1.7)<br>      (SID_NAME = prd) | | |

```
   )
  )

PASSWORDS_LISTENER=(k3h8vy6w)
```

2. A review of the file reveals that the listener port is the default of 1521.
3. Output from the tnscmd status command.

```
Perl tnscmd status --indent -h dbsrv1
sending (CONNECT_DATA=(COMMAND=status)) to dbsrv1:1521
writing 89 bytes
reading
. .......6.........a. ...........k........
  DESCRIPTION=
    TMP=
    VSNNUM=135295744
    ERR=0
    ALIAS=LISTENER
    SECURITY=ON
    VERSION=TNSLSNR for DEC OSF/1 AXP: Version 8.1.7.3.0 -
Production
    START_DATE=15-OCT-2003 23:47:05
    SIDNUM=1
    LOGFILE=/u01/app/oracle/product/8.1.7/network/log/listener.log
    PRMFILE=/u01/app/oracle/product/8.1.7/network/admin/listener.ora
    TRACING=off
    UPTIME=2974532
    SNMP=OFF

."........
  ENDPOINT=
    HANDLER=
      STA=ready
      HANDLER_MAXLOAD=0
      HANDLER_LOAD=0
      ESTABLISHED=0
      REFUSED=0
      HANDLER_ID=C9C868DC8123-5CE8-E030-348CFDA7C274
      PRE=ttc
      SESSION=NS
      DESCRIPTION=
        ADDRESS=
          PROTOCOL=tcp
          HOST=dbsrv1
          PORT=1521
,,
  ENDPOINT=
    HANDLER=
      STA=ready
      HANDLER_MAXLOAD=0
      HANDLER_LOAD=0
      ESTABLISHED=0
      REFUSED=0
      HANDLER_ID=C9C868DC8124-5CE8-E030-348CFDA7C274
      PRE=ttc
    SESSION=NS
    DESCRIPTION=
      ADDRESS=
        PROTOCOL=ipc
        KEY=EXTPROC

,,
  SERVICE=
    SERVICE_NAME=prd
    INSTANCE=
      INSTANCE_NAME=prd
      NUM=2
      INSTANCE_CLASS=ORACLE
      NUMREL=2

,
  SERVICE=
    SERVICE_NAME=prd05
    INSTANCE=
      INSTANCE_NAME=prd05
      NUM=2
      INSTANCE_CLASS=ORACLE
      NUMREL=2

,,.........@
```

4. A review the output of the tnscmd status command reveals the listener is running on port 1521.
5. Output from an nmap scan of DBSRV1.

```
Interesting ports on dbsrv1 (10.11.12.13):
(The 65493 ports scanned but not shown below are in state: closed)
Port       State       Service
21/tcp     open        ftp
22/tcp     open        ssh
25/tcp     open        smtp
110/tcp    open        pop-3
111/tcp    open        sunrpc
143/tcp    open        imap2
316/tcp    open        decauth
515/tcp    open        printer
596/tcp    open        smsd
619/tcp    open        unknown
886/tcp    open        unknown
1024/tcp   open        kdm
1025/tcp   open        NFS-or-IIS
1026/tcp   open        LSA-or-nterm
1028/tcp   open        unknown
1521/tcp   open        oracle
2121/tcp   open        unknown
2481/tcp   open        unknown
2793/tcp   open        unknown
3354/tcp   open        unknown
6000/tcp   open        X11
6112/tcp   open        dtspc
7150/tcp   open        unknown
7152/tcp   open        unknown
7153/tcp   open        unknown
```

| | |
|---|---|
| | 7650/tcp   open        unknown<br>7651/tcp   open        unknown<br>7652/tcp   open        unknown<br>9150/tcp   open        unknown<br>9151/tcp   open        unknown<br>9152/tcp   open        ms-sql2000<br>9153/tcp   open        unknown<br>9154/tcp   open        unknown<br>9155/tcp   open        unknown<br>9650/tcp   open        unknown<br>9651/tcp   open        unknown<br>9652/tcp   open        unknown<br>9653/tcp   open        unknown<br>9654/tcp   open        unknown<br>9655/tcp   open        unknown<br>10401/tcp  open        unknown<br>10402/tcp  open        unknown<br>Device type: general purpose<br>Running: Compaq Tru64 UNIX 5.X<br>OS details: Compaq Tru64 UNIX V5.1A (Rev. 1885)<br>TCP Sequence Prediction: Class=truly random<br>                        Difficulty=9999999 (Good luck!)<br>IPID Sequence Generation: Incremental<br><br>6. Reviewing the output of the nmap scan finds the Oracle listener running on port 1521. |
| Compliance Determination | Fail ✘ |

### 3.1.9 Package Access

| | | | |
|---|---|---|---|
| Reference: Checklist #: 2.2.4.11 Package Access on page 44.<br>Analysis in final report on page 80. | | | |
| Control Objective | This control objective is designed to ensure that the access to packages that provide additional capabilities have not been granted to PUBLIC. | | |
| References | ♦ “Oracle Database Security Benchmark v1.0”, p. 24. | Objective / Subjective | Subjective |
| Risk | Access to the packages provides users with additional privileges that may or may not be necessary for their job function. Unnecessary access to these packages provides privileges that are not required and could be exploited. Use of these packages is complex and the ability to work outside of the application interface is limited, therefore the likelihood of exploitation is low. | Risk Level | Medium |

| | |
|---|---|
| Compliance Criteria | Compliance for this item is objective based on the PUBLIC user group being granted access to the following packages.<br>• UTL_FILE<br>• UTL_TCP<br>• UTL_HTTP<br>• UTL_SMTP<br>• DBMS_LOB<br>• DBMS_SYS_SQL<br>• DBMS_JOB<br>Pass:  None of the packages have been granted to the PUBLIC user group.<br>Fail:  At least one of the packages has been granted to the PUBLIC user group. |
| Audit Procedure | 1. Execute the following SQL*Plus statement to find packages that have been granted to PUBLIC with execute privileges.<br>*select table_name from dba_tab_privs where grantee=’PUBLIC’ and privilege=’EXECUTE’ and table_name in (‘UTL_FILE’,’UTL_TCP’,’UTL_HTTP’,’UTL_SMTP’,’DBMS_LOB’,’DBMS_SYS_SQL’,’DBMS_JOB’);*<br>2. Review the results of step 1, if any are provided then this item fails compliance. |
| Audit Results | 1. Output from SQL*Plus:<br>UTL_FILE<br>2. After review, this item fails because one of the packages provides public execute privileges. |
| Compliance Determination | Fail ✘ |

3.1.10 “Public” Permissions

| | | | |
|---|---|---|---|
| Reference:  Checklist #: 2.2.1.6  “Public” Permissions on page 23.<br>Analysis in final report on page 74. | | | |
| Control Objective | This control objective is designed to ensure that the permissions granted to the user group “PUBLIC” are appropriate. | | |
| References | ♦ Personal Knowledge<br>♦ Newman, p. 38. | Objective / Subjective | Subjective |
| Risk | Permissions granted to the PUBLIC user group are provided to all database users because all | Risk Level | High |

| | | | |
|---|---|---|---|
| | database users are members of the PUBLIC user group. Permissions granted to PUBLIC should be limited to those that all users need to have. It is very likely that the permissions assigned to PUBLIC will be greater than necessary if they are not reviewed. | | |
| Compliance Criteria | Compliance will depend on the analysis of the permissions / roles / privileges granted to the PUBLIC user group. This review will need to be done in coordination with the DBA to determine what permissions are required of all users.<br>Pass: The analysis has concluded that all permissions / roles / privileges granted to the PUBLIC user group are appropriate.<br>Fail: PUBLIC has been granted permissions / roles / privileges that are not appropriate for all users to have.<br>. | | |
| Audit Procedure | 1. Obtain a list of all the objects where permissions have been granted to PUBLIC.<br>*select table_name, privilege from dba_tab_privs where grantee='PUBLIC';*<br>2. Review the output of step 1 with the DBA to determine if the permissions are necessary for the PUBLIC user group.<br>3. Obtain a list of all roles that have been granted to PUBLIC.<br>*select granted_role from dba_role_privs where grantee='PUBLIC';*<br>4. Review the output of step 3 with the DBA to determine if the roles are necessary for the PUBLIC user group.<br>5. Obtain a list of all system privileges that have been granted to PUBLIC.<br>*Select privilege from dba_sys_privs where grantee='PUBLIC';*<br>6. Review the output of step 3 with the DBA to determine if the roles are necessary for the PUBLIC user group. | | |
| Audit Results | 1. Output from SQL*Plus:<br>BAT.NTUSERS: SELECT<br>BAT.BATT_ACTIVE_USERS: SELECT<br>BAT.BATT_APPLICATIONS: SELECT<br>BAT.BATT_NONR_EMP: SELECT<br>BAT.BATT_PUBLIC_READ_USAGE: SELECT<br>BAT.BATT_USER_ROLES: SELECT<br>DKAT.DKATT_USERS: SELECT<br>CIRWS.TMP_WSRT_BUILDING: SELECT<br>CIRWS.TMP_WSRT_OUTPUT: SELECT<br>CIRWS.TMP_WSRT_OUTPUT_CCC: SELECT<br>CIRWS.TMP_WSRT_OUTPUT_EPACODES: SELECT<br>CIRWS.TMP_WSRT_OUTPUT_LDR_SUBCAT: SELECT<br>CIRWS.TMP_WSRT_PROCESS: SELECT | | |

| | |
|---|---|
| | BAT.BATV_PRD_USERS: SELECT<br>SQLCQR.SQLCQRV_PWDSTATS: SELECT<br>BAT.BATV_ACTIVE_USERS: SELECT<br>BAT.ROLE_STRINGS: EXECUTE<br>SQLCQR.SQLCQR_PWD: EXECUTE<br>BAT.BATT_DB_SERVER: SELECT<br>PRD.IMPORTCONTAINERBINDINGTAB2: SELECT<br>BAT.BATT_APP_DB: SELECT<br>TMS.TOOLT_USERS: SELECT<br>SPARCS.SPARCSP_SECURITY: EXECUTE<br>SPARCS.SPARCSP_SYSTEM: EXECUTE<br>BAT.BATT_ALL_EMP: SELECT<br>CUE.CUET_PHONES: SELECT<br>CWEF.CWEFT_D_NAMES: SELECT<br>CWEF.CWEFT_D_JOB: SELECT<br>CWEF.CWEFT_D_HISTORY: SELECT<br>FCW.FCWS_EMP_SEQ: SELECT<br>FCW.FCWS_EQUIP_SEQ: SELECT<br>FCW.FCWS_LOCATION: SELECT<br>FCW.FCW_SEQ_NO: SELECT<br>FCW.FCWT_ARCH_ADM: SELECT<br>FCW.FCWT_ARCH_EQU: SELECT<br>FCW.FCWT_ARCH_ORI: SELECT<br>FCW.FCWT_ARCH_PRO: SELECT<br>FCW.FCWT_ARCH_RES: SELECT<br>FCW.FCWT_ARCH_SHI: SELECT<br>FCW.FCWT_ARCH_SOU: SELECT<br>FCW.FCWT_DAVIS_BACON: SELECT<br>FCW.FCWT_FACTOR: SELECT<br>FCW.FCWT_MEQ: SELECT<br>FCW.FCWT_MINOR: SELECT<br>FCW.FCWT_MINOR_CRAFT: SELECT<br>FCW.FCWT_RANK: SELECT<br>FCW.FCWT_REPORT: SELECT<br>FCW.FCWT_RISK: SELECT<br>FCW.FCWT_RPT_COLUMN: SELECT<br>FCW.FCWT_SM_ACCESS: SELECT<br>FCW.FCWT_SM_BLDG_CODE: SELECT<br>FCW.FCWT_SM_CLOSE_CODE: SELECT<br>FCW.FCWT_SM_CLUSTER: SELECT<br>FCW.FCWT_SM_DAVIS_BACON: SELECT<br>FCW.FCWT_SM_DESIGNEE: SELECT<br>FCW.FCWT_SM_EMPLOYEE: SELECT<br>FCW.FCWT_SM_EQUIP_MASTER: SELECT<br>FCW.FCWT_SM_OPS_AREA: SELECT<br>FCW.FCWT_SM_ORG_CODE: SELECT<br>FCW.FCWT_SM_PRIORITY_CODE: SELECT<br>FCW.FCWT_SM_PROG: SELECT<br>FCW.FCWT_SM_RESPONSE_CODE: SELECT<br>FCW.FCWT_SM_RISK: SELECT<br>FCW.FCWT_SM_ROLE: SELECT<br>FCW.FCWT_SM_SAFETY_CODE: SELECT<br>FCW.FCWT_SM_SHIFT_PEOPLE: SELECT<br>FCW.FCWT_SM_SOURCE_CODE: SELECT<br>FCW.FCWT_SM_SYS_CATEGORY: SELECT<br>FCW.FCWT_SM_VSS_BLDG: SELECT |

| | |
|---|---|
| | FCW.FCWT_SM_VSS_DSGNR: SELECT<br>FCW.FCWT_SM_VSS_TCOMP: SELECT<br>FCW.FCWT_SM_WBS_CODE: SELECT<br>FCW.FCWT_SM_WORK_DSC: SELECT<br>FCW.FCWT_SOURCE: SELECT<br>FCW.FCWT_STATUS_MMS: SELECT<br>FCW.FCWT_SYS_EQU: SELECT<br>FCW.FCWT_SYS_NEWS: SELECT<br>FCW.FCWT_SYS_OPS_WBS: SELECT<br>FCW.FCWT_TRANS_LOG: SELECT<br>FCW.FCWT_XFER: SELECT<br>FCW.FCWV_DAVIS_BACON: SELECT<br>FCW.FCWV_FCWS_CUET_PHONES: SELECT<br>FCW.FCWV_CLUSTERS: SELECT<br>FCW.FCWV_CWBST_CHARGES: SELECT<br>FCW.FCWV_SM_EQUIP_MASTER: SELECT<br>FCW.FCWV_CRAFT: SELECT<br>FCW.FCWP_UTILITY: EXECUTE<br>FCS.FCSP_CALCS: EXECUTE<br>SMP.MMST_CLUSTERS: SELECT<br>SMP.FCWV_SM_EQUIP_MASTER: SELECT<br>CMEW.NUM_VARRAY: EXECUTE<br>CMEW.STR_VARRAY: EXECUTE<br>2. After review with the DBA it has been determined that PUBLIC does not need privileges to the majority of the above tables.<br>3. Output from SQL*Plus:<br>FCW_READ<br>4. After review with the DBA it has been determined that PUBLIC does not need the FCW_READ role.<br>5. The output from SQL*Plus did not contain any rows. This means that there are no system privileges granted to PUBLIC.<br>6. Nothing to review. |
| Compliance Determination | Fail ✗ |

### 3.2 Residual Risk

Given the number of items that failed the audit (15 out of 41), you could expect a fair amount of residual risk. However, most of the items that failed can be fixed relatively easily without a large impact to the system and its' associated business processes. The following table is a summary of all the failed items, with an