DATABASE
SECURITY TECHNICAL IMPLEMENTATION GUIDE
Version 7, Release 2

30 November 2005

Developed by DISA for the DOD

UNCLASSIFIED

This page is intentionally left blank.

TABLE OF CONTENTS

Page

This page is intentionally left blank.

LIST OF TABLES

This page is intentionally left blank.

SUMMARY OF CHANGES

Changes made since the previous version/release (Version 7, Release 1) are listed below.

Appendix B.5.3. A note was added to inform the reader that the removal of execute privileges to PUBLIC for the 10 listed packages may disable the functioning of some default database applications.

1 INTRODUCTION

This *Database Security Technical Implementation Guide* (STIG) is published as a tool to assist in the improvement of the security of Department of Defense (DOD) information systems. The document is meant for use in conjunction with the appropriate Operating System (OS) STIG as well as other STIGs related to the requirements of any applications accessing the database. Frequently, this includes the DISA *Web Server STIG*, and the DISA *Application Security Document*. The most effective way to improve security in DOD database systems is to include security in the initial design and development of the application. To that end, this document is also intended to be useful to application program managers/developers in the design phase of DOD applications. As such, it provides the technical security policies, requirements, and implementation details for applying security concepts to database servers. It covers generally all database servers and specifically Oracle, Microsoft SQL Server, and DB2 database servers supporting storage and retrieval of data from local, intranet, or Internet clients.

1.1 Background

More and more frequently, data stored within a database management system (DBMS) has become a target of attack for malicious users. The effect of such an attack can result in identity and/or credit card theft, financial loss, loss of privacy, a breach of national security or any other type of corruption that can result from unauthorized access to sensitive data. As database products have evolved, more and more security options are becoming available. DBMSs have also joined the ranks of victims of malicious attacks and DBMS vendors have had to respond by issuing fixes for discovered vulnerabilities. This STIG presents the known security configuration items, vulnerabilities, and issues required to be addressed by DOD policy. In addition to this STIG, compliance validation tools and checklists are available to **.mil** and **.gov** customers to assist in the efforts to implement the required configuration.

It should be noted that Defense Information Systems Agency (DISA) Field Security Operations (FSO) Support for the STIGs, Checklists, and Tools is only available to DOD Customers.

1.2 Authority

DOD Directive 8500.1 requires that “all information assurance (IA) and IA-enabled IT products incorporated into DOD information systems shall be configured in accordance with DOD-approved security configuration guidelines” and tasks DISA to “develop and provide security configuration guidance for IA and IA-enabled IT products in coordination with the Director, National Security Agency (NSA).” This document is provided under the authority of DOD Directive 8500.1.

The use of the principles and guidelines in this STIG will provide an environment that meets or exceeds the security requirements of DOD systems operating at the Mission Assurance Category (MAC) II Sensitive level, containing unclassified but sensitive information.

1.3 Scope

This document applies to all DBMSs presently in use within DOD with specific guidelines provided for Oracle Database Server, Microsoft SQL Server, and IBM DB2. Requirements specific to a particular product are covered in separate appendices. All requirements in the main body of the document are meant to apply to all databases generally. Databases not specifically covered in this STIG should be secured according to the general guidance as well as by following vendor-recommended security configurations and application of all vendor-supplied security fixes.

Operating system-specific considerations are supplied for some platforms. Specific platform coverage is limited to those for which resources are available for research. Typically, Windows and UNIX coverage will be included where applicable. IBM mainframe coverage is provided when possible.

This document addresses all known common security features as well as specific security configuration requirements for the DBMSs and versions that are listed below:

DBMS VENDORS	***DBMS VERSION(S) ADDRESSED***	***HOST PLATFORM***
Oracle	Database Server 8i, 9i, 10g	Windows, UNIX, OS/390
Microsoft	SQL Server 7 and 8 (2000)	Windows
IBM	DB2 Universal Database 8.1	Windows, UNIX

Table 1. DATABASES COVERED IN THIS STIG

1.4 Writing Conventions

Throughout this document, statements are written using words such as "**will**" and "**should**." The following paragraphs are intended to clarify how these STIG statements are to be interpreted.

A reference that uses "**will**" implies mandatory compliance. All requirements of this kind will also be documented in the italicized policy statements in bullet format, which follow the topic paragraph. This will make all "**will**" statements easier to locate and interpret from the context of the topic. The Information Assurance Officer (IAO) will adhere to the instruction as written. Only an extension issued by the Designated Approving Authority (DAA) will table this requirement. The extension will normally have an expiration date, and does not relieve the IAO from continuing their efforts to satisfy the requirement.

A reference to "**should**" is considered a recommendation that further enhances the security posture of the site. These recommended actions will be documented in the text paragraphs but not in the italicized policy bullets. Nevertheless, all reasonable attempts to meet this criterion will be made.

For each italicized policy bullet, the text will be preceded by parentheses containing the italicized Short Description Identifier (SDID), which corresponds to an item on the checklist and the severity code of the bulleted item. An example of this will be as follows "(*G111: CAT II*)". If the item presently has no Potential Discrepancy Item (PDI), or the PDI is being developed, it will contain a preliminary severity code and "N/A" for the SDID (i.e., "(*N/A: CAT III)*").

1.5 Vulnerability Severity Code Definitions

Category I	Vulnerabilities that allow an attacker immediate access into a machine, allow superuser access, or bypass a firewall.
Category II	Vulnerabilities that provide information that have a high potential of giving access to an intruder.
Category III	Vulnerabilities that provide information that potentially could lead to compromise.
Category IV	Vulnerabilities, when resolved, will prevent the possibility of degraded security.

1.6 DISA Information Assurance Vulnerability Management (IAVM)

The DOD has mandated that all IAVMs are received and acted on by all commands, agencies, and organizations within the DOD. The IAVM process provides notification of these vulnerability alerts and requires that each of these organizations take appropriate actions in accordance with the issued alert. IAVM notifications can be accessed at the Joint Task Force – Global Network Operations (JTF-GNO) web site (http://www.cert.mil).

1.7 STIG Distribution

Parties within the DOD and Federal Government's computing environments can obtain the applicable STIG from the Information Assurance Support Environment (IASE) web site. This site contains the latest copies of any STIG, as well as checklists, scripts, and other related security information.

The Non-classified Internet Protocol Router Network (NIPRNet) Uniform Resource Locator (URL) for the IASE site is http://iase.disa.mil/. Access to the STIGs on the IASE web server requires a network connection that originates from a **.mil** or **.gov** address. The STIGs are available to users that do not originate from a **.mil** or **.gov** address by contacting the DISA Field Security Operations (FSO) Support Desk at DSN 570-9264, commercial 717-267-9264, or e-mail to **fso_spt@disa.mil** or from the National Institute of Standards (NIST) at http://csrc.nist.gov/pcig/cig.html.

1.8 Document Revisions

Comments or proposed revisions to this document should be sent via e-mail to **fso_spt@disa.mil**. DISA FSO will coordinate all change requests with the relevant DOD organizations before inclusion in this document.

This page is intentionally left blank.

2 INTEGRITY

Sites achieve improved database integrity by managing the overall processing environment. Proper security and system management helps to protect system hardware, software, application and data from unauthorized access and improper modification and contributes to secure operation of database systems. System integrity is most vulnerable to malicious intrusion before systems have been completely configured for secure operation. Newly built or configured systems are more vulnerable to a compromise of data integrity as soon as they are connected to a production network if they are not STIG compliant before connection. Database Administrators (DBA) will ensure that database systems conform to the security directives presented in this STIG before they are connected to the network. Conformance includes the successful completion of the extension process for any security items that cannot be met. Existing systems should be self-assessed for compliance as soon as possible if not already assessed.

2.1 Software Integrity

The integrity of the DBMS software depends in part upon the integrity of its supporting host system software. Operating system integrity is covered in each OS STIG and will not be duplicated in this document.

Some DBMSs have been evaluated in accordance with the Common Criteria (ISO 15408). These evaluations provide configuration standards to establish a level of security assurance as defined in their evaluation criteria. DBMSs, as do all Commercial-Off-The-Shelf (COTS) software used within DOD, that do not appear in the National Information Assurance Partnership (NIAP) Validated Products List (http://niap.nist.gov/cc-scheme/ValidatedProducts.html) require DAA approval and acceptance of risk.

Application software that utilizes data stored in a DBMS must be evaluated independently from the DBMS. However, DBMS policy declared in this STIG must be followed.

The operational integrity of the OS is the primary concern of the System Administrator (SA). The operational integrity of the DBMS is the concern of the DBA. The IAO is concerned primarily with the security integrity of both the operating system and the database.

2.1.1 Current DBMS Version

The integrity of the DBMS software executables and datafiles is crucial to the optimal and correct operation of all applications using the DBMS. To protect the DBMS environment, the IAO will ensure that the DBMS version is a vendor-supported product version. Vendor supported product versions are those that continue to receive security updates by the vendor in response to the discovery of vulnerabilities. The DBA will ensure that the DBMS patch level is current. Systems unable to support upgrades require an extension for non-compliance filed with a signed acceptance of risk by the system DAA.

The DBMS host should be an approved/certified platform to host the database.

- *(DO0100, DM0590, DM0710: CAT I) The IAO will ensure that unsupported DBMS software is removed or upgraded prior to a vendor dropping support.*

- *(DG0002: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(DO0100, DM1769, DM0710: CAT I) The IAO will ensure that the DBMS version has all patches applied*.

2.1.2 DBMS Software/Object Modification

The DBMS software installed on the host system will be monitored monthly or more frequently for unauthorized modification. Trojan horses and other malicious code could be implanted in standard database executables that could corrupt database integrity or allow unauthorized access. Host systems should baseline their systems after application installation to collect data on application directories and files for future comparison in order to determine unauthorized modification. Third party application software, if installed on the database server, will be installed on separate logical storage partitions from the database software to provide separate resource controls and security contexts.

DBMS application objects including but not limited to functions and procedures will be monitored monthly for unauthorized modification. Like the host system, a baseline database of application objects is necessary in order to detect unauthorized changes such as procedure recompiles or object restructuring. Auditing of modification of or access to application data is not required by this policy.

Software configuration management policies will be implemented and strictly enforced on production database systems to ensure unauthorized software is not implemented.

Database software other than software such as SQL Server Enterprise Manager and Oracle Enterprise Manager intended for DBA use to administer the database will not use Data Definition Language (DDL) statements to alter the database schema.

- *(DG0010: CAT III) The IAO will ensure that DBMS software is monitored on a regular basis no less frequently than weekly to detect unauthorized modifications.*

- *(DG0010: CAT III) The DBA will provide to the SA and IAO a list of database software directories to be included in software backup, baselining, and monitoring.*

- *(DG0050: CAT IV) The IAO will ensure that configuration management policies and procedures are implemented for database software modifications.*

- *(DG0015: CAT III) The IAO will ensure that database applications do not use DDL statements.*

- *(DG0012: CAT III) The SA will ensure that all third-party database application software is installed in a logical partition separate from the DBMS software and datafiles.*

2.1.3 Unused Database Software/Components

All binary files, database accounts, database objects, and application code stored in the database that are solely associated with an unused or decommissioned application or database component will be removed from the database and host system. Aside from administrative overhead in maintaining unused software, unnecessary vulnerabilities associated with them remain in the database. DBAs should take precautions to remove any objects or components that are no longer required and that a backup of the database and applications is current. The Oracle Universal Installer (OUI) is the preferred and safest method to de-install unused Oracle Components.

- *(DG0016: CAT II) The DBA will ensure that unused database components and database software applications and any associated database objects are removed from the database and host system.*

2.2 Database Software Development

Database software development introduces unknown vulnerabilities to the database system. Software development will not be done on production databases. Software development on hosts supporting production databases will be separated from the production database and production application software. Development databases will be configured in accordance with all security requirements for production databases with the exception of specifically noted differences. This ensures that the application is developed within the boundaries of good security.

Separated databases on the same host require strict partitioning of database datafiles, executables, and process/service host system resources. Labeling including file names, instance names, and other related variables must clearly distinguish the production and development database resources in order to avoid any inadvertent access to the wrong database system. Database account names should differ between the two systems.

No database links will be created or used between production and development database systems. Applications and databases under development will not access production databases.

Development databases created from production databases must be cleaned to remove sensitive data before or immediately after import to the development database. This includes database account passwords and deletion or modification of sensitive data such as personnel or financial, etc. information. The DBA must and will ensure that imported passwords and sensitive data are protected from view in a development database.

Privileges granted to application developers on shared production/development database systems to modify application code or application objects will be reviewed every three months to determine continued appropriateness.

- *(DG0017: CAT II) The DBA will ensure that software development on a production system is separated through the use of separate and uniquely identified data and application file storage partitions and processes/services.*

- *(DG0050: CAT IV) The IAO will ensure that software configuration management policies are implemented and strictly enforced to ensure untested software is not inadvertently loaded to production systems.*

- *(DG0075: CAT II) The DBA will ensure that no database links are defined between production and development databases.*

- *(DG0075: CAT III) The DBA will ensure that development applications do not access production databases unless justified and documented with the IAO.*

- *(DG0076: CAT II) The DBA will ensure that development databases created from production database exports have passwords changed from their production values.*

- *(DG0076: CAT II) The DBA will ensure that export data from a production database used to populate a development database has all sensitive data such as payroll data or personal information, etc., removed or modified prior to import to the development database.*

- *(DG0077: CAT II) The IAO will review privileges granted to developers on shared production/development database systems to modify application code or application objects every three months or more frequently.*

2.2.1 Shared Production/Development Systems – STIG Impacts

If software development is accomplished on a shared production/development system, the only deviations from this STIG aside from the separation of applications and database files, deal with the assignment of roles and privileges granted to developer database accounts. On shared production/development systems, an application developer account will not be given permission to create, alter, or drop schema objects. On shared production/development systems, at no time will the application developer account be given DBA roles within the database or on the operating system. Other deviations from this STIG will be handled on a case-by-case basis and will be fully justified and documented in accordance with this STIG. At no time will developers be granted system privileges within a production database. At no time will development occur within a production database.

- *(DG0077: CAT II) The DBA will ensure that developers are not granted system privileges within a production database.*

2.3 Ad Hoc Queries

Ad hoc queries allow the user to submit an untested request to the database. Such unrestricted access may have a negative impact on system performance or be used to exploit unknown vulnerabilities. Ad hoc queries should be disallowed on production systems. Instead, create and store frequently used queries in procedures. In some types of databases such as data warehouses, ad hoc queries may be required.

2.4 Multiple Services Host Systems

The installation of a DBMS on a host platform introduces additional vulnerabilities and resource requirements to the host. Additionally, the DBMS frequently offers services to clients that are members of a different audience than other services. Since it is a best security practice to separate or partition services offered to different audiences, any DBMS should be installed on a host system dedicated to its support and offering as few services as possible to other clients. For this reason, a DBMS should not be installed on a host system that also provides web services, directory services including a Windows primary domain or backup domain controller, directory naming services, etc.

2.5 Data Integrity

The goal of database security is to protect your critical and confidential data from unauthorized access. Access in this context means not only changing or deleting the data in your database, but also just reading or disclosing it. Database security should provide controlled, protected access to the contents of your database and, in the process, preserve the integrity, consistency, and overall quality of your data.

The integrity of all data within a DBMS is the prime concern of the DBA, IAO, and the application designers. Application design, development, and implementation are key factors in ensuring data integrity. Without the appropriate attention to data integrity enforcement from the DBA, IAO, and application designers, the entire application could be rendered unreliable to the customer. To ensure data integrity, it is important to perform database administration correctly, regularly, and reliably. Some of the duties that will be performed to promote data integrity are as follows:

- Correct and successful physical backup of all database data
- Correct and successful logical backup of all database data
- Recovery operations
- Database performance analysis
- Auditing enabling
- Audit data analysis

2.5.1 Database File Integrity

To protect the integrity of the database files, database COTS software authorizations will not be modified from installation defaults to be more permissive. All permissions on directories and files created as the result of an installation of the DBMS software will comply with the security evaluation specification (CC or TCSEC evaluated configuration) permissions if available or to the vendor recommended permissions if not. Permissions to change directory names, file permissions, or group information associated with the database software will be restricted to SAs and DBAs. Third party application software will be installed on a separate partition and all associated files and directories will be owned by the third-party application installation account.

- *(DO3613: CAT II) The SA will ensure that permissions to database software comply with security evaluation specifications. If unavailable, permissions to database software will be set to comply with vendor recommended permissions.*

- *(DO3613: CAT II) The SA will ensure that all directories created by the installation of the DBMS are protected in accordance with security evaluation specifications if available. If unavailable, DBMS directory permissions will be set to comply with vendor recommendations.*

- *(DO3613: CAT II) The SA will ensure that all file permissions created by the installation of a DBMS are modified as necessary to comply with security evaluation specifications if available. If unavailable, DBMS file permissions will be set to comply with vendor recommended permissions.*

- *(DO3613: CAT II) The SA will ensure that permissions to change directory names, file permissions, or group information associated with the database software are restricted to SAs and DBAs.*

- *(DG0012: CAT III) The SA will ensure that all third-party database application software is installed in a logical partition separate from the DBMS software and datafiles.*

- *(DG0019: CAT III) The SA will ensure that all DBMS and third-party database application software files and directories are owned by the application software installation account and are protected from access by more than the minimal number of users required.*

2.5.2 Database Software Baseline

Every DBMS will have a good backup of the database software, which is performed correctly and completely. An operating system backup of the database software will be performed every time the DBMS software is upgraded.

- *(DG0021: CAT II) The SA/DBA will backup the database software after every database software upgrade.*

2.5.3 Database File Backup and Recovery

A tested and verifiable backup strategy will be implemented for all databases. Backup and recovery procedures will be documented. Database backup and recovery procedures are left to the specific applications to design, test, and implement.

The site will have a contingency processing plan/disaster recovery plan that includes DBMS databases. The contingency plan will be periodically tested in accordance with DODI 8500.2 requirements.

The site will identify an off-site storage facility in accordance with DODI 8500.2 requirements. Off-site backups will be updated on a regular basis and the frequency will be documented in the contingency plan.

- *(DG0020: CAT II) The DBA will ensure that a tested and verifiable backup strategy is implemented on all DBMS databases*.

- *(DG0020: CAT II) The DBA will ensure that documented backup and recovery procedures exist.*

This page is intentionally left blank.

3 DISCRETIONARY ACCESS CONTROL

This section discusses the discretionary access controls and the identification and authentication criteria necessary to ensure that access to database resources are effectively managed and controlled for any database management system.

3.1 Database Account Controls

DOD directives require unique identification for each system user. A user is either an individual or an executing process/service that accesses a computer resource. Actions performed by a user within the database will be traceable to an individual database account.

DOD policy requires that permissions and privileges granted to individuals within an automated information system follow the concept of least privilege. Authorized database accounts will be granted access only to the resources needed to accomplish the mission. Authorized database accounts will have roles assigned that contain the minimum privileges necessary to perform each of their job functions. All database accounts will be protected by a certificate, password, or other approved authentication method.

The use of shared database accounts (accounts where multiple users are allowed to log on directly to the same account) denies individual accountability. If there is an absolute requirement for logging directly into a shared account, such as Oracle's SYS account, the IAO will obtain justification and documentation from the vendor that states the necessity. Software OS installation and maintenance accounts do not require justification; however, access to and use of these accounts should be protected and logged.

- *(DG0070: CAT II) The DBA will ensure that all database actions are traceable to an individual user.*

- *(DO3709: CAT III) The DBA will ensure that all database accounts are granted roles containing the minimum set of privileges required for the application.*

- *(DO3504: CAT II) The DBA will configure all database accounts to be protected by a password, certificate, or other approved authentication method.*

- *(DG0060: CAT II) The DBA will ensure that use of shared database accounts are justified and documented with the IAO.*

3.2 Authentication

Where possible, databases are encouraged to use Public Key Infrastructure (PKI) as the method of authentication in accordance with the *DODI 8520.2, Public Key Infrastructure (PKI) and Public Key (PK) Enabling, 1 April 2004*. Databases supporting web services must comply with appropriate policies as specified in the DISA *Web Server STIG*. Otherwise, use of PKI for authentication will be considered and a business case analysis will be submitted to the DOD component Chief Information Officer (CIO) for review and approval. (See *Section E3.4.1.3, DODI 8520.2*.) Where justified, use of PKI for authentication is not required.

Where use of PKI authentication is not available, the authentication method specified in the database security evaluation under Common Criteria (CC) or the Department of Defense's (DOD) Trusted Computer Security Evaluation Criteria (TCSEC) is preferred. For Oracle on Windows platforms and MS SQL Server, Windows authentication is preferred or required; for Oracle on UNIX platforms, Oracle authentication is preferred. Informix was evaluated using OS authentication and Sybase with database authentication. IBM's DB2 has not been evaluated by either criterion and accepts only OS authentication.

- *(DG0065: CAT III) The IAO will ensure that use of PKI for database authentication is compatible with DOD PKI specifications.*

3.2.1 Password Guidelines

Passwords must be protected from being accessed by unauthorized users. When an account is created for a user, that user will be given a temporary password. The administrator will brief the user on DOD password policy (DODI 8500.2 and *Chairman of the Joint Chiefs of Staff (CJCSM) 6510.01C*) and implementation of password protection.

All passwords will be stored in an encrypted format. The database account name and password will not be visible to the host or client operating system.

Where available, database account logons will be limited to three failed logons before they become locked. This requirement reduces the ability for password cracking programs to be used successfully. The DBA will set the duration of the lock time to a specific length as approved by the IAO for the application or site or require a manual reset. The duration should be set appropriately for the environment, keeping in mind that the longer the duration, the more protected the accounts will be from password cracking programs.

Default passwords for the database accounts created during installation of DBMS software, DBMS optional software, or other Commercial-Off-The-Shelf (COTS) database software will be changed immediately after installation.

- *(DG0066: CAT II) The DBA will assign a temporary database account password at database account creation.*

- *(DO3504, DO3485, DO348: CAT II) The DBA will ensure that database account passwords conform to DOD password policy and each user is briefed on the policy on receiving a temporary password.*

- *(DG0090, DG0067: CAT II) The DBA will ensure that database account passwords are stored in an encrypted format.*

- *(DG0068: CAT II) The DBA will ensure that database account names and database account passwords are not visible in clear text on the host command line.*

- *(DO3504: CAT II) The DBA will ensure that database account passwords are a minimum of eight alphanumeric characters in length and do contain a mix of upper case letters, lower case letters, numbers, and special characters.*

- *(DO3504: CAT II) The DBA will ensure that database account passwords do not contain personal information such as names, telephone numbers, account names, dictionary words, etc.*

- *(DO3504: CAT II) The DBA will ensure that database account passwords do not contain consecutively repeating characters.*

- *(DO3504: CAT II) Where possible, the DBA will ensure that new database account passwords differ from the previous password by at least four characters when a password is changed.*

- *(DO3485: CAT II) The DBA will ensure that database account passwords are changed every 90 days or more frequently.*

- *(DO3487: CAT II) The DBA will ensure that database account passwords are not reused within ten password changes.*

- *(DO3487: CAT II) Where available, the DBA will ensure that database account passwords are not reused for a period of one year or longer.*

- *(DO3485: CAT II) The DBA will ensure that application database account passwords are changed at least once a year and anytime an application administrator is reassigned.*

- *(DG0072: CAT II) The DBA will ensure that users are not allowed to change their database account passwords more than once every 24 hours without IAO approval.*

- *(DO3445: CAT I) The DBA will change all default database account passwords for database accounts created during DBMS installation and used by DBMS processes immediately after installation.*

- *(DO3537: CAT II) Where available, the DBA will limit database account logons to three failed logons before they become locked.*

- *(DG0073: CAT II) Where available, the DBA will set the length of database account lock time to a minimum as approved by the IAO for the application/site.*

3.2.2 Certificate Guidelines

Certificates used for authentication must be used in accordance with *DODI 8520.2, Public Key Infrastructure (PKI) and Public Key (PK) Enabling, 1 April 2004.*

3.3 Database Accounts

3.3.1 Administrative Database Accounts

DBA accounts are accounts designed and intended for use to administer the database storage architecture, grant privileges, and provide oversight management to all database objects within the database and, in some cases, to start, stop, and configure the database process. These privileged accounts should be used only for administration of the database and should not be used for application development, application testing, or application use. DBA accounts will only be used when performing administrative tasks. All database administration accounts will be password protected at all times. Default installation passwords will not remain on DBA database accounts. Shared and/or default DBA database accounts will not be used. Each DBA will use an individually assigned DBA-privileged database account for DBA activities. Operating system accounts granted DBA privileges within the database by means of assignment to an OS-defined group must be individually assigned. DBA database accounts will not be used for non-DBA activities. Instead, less-privileged database accounts will be used for non-DBA activities.

- *(DO3445: CAT I) The DBA will ensure that all default installation passwords do not remain on DBA database accounts.*

- *(DO0140: CAT II) The IAO will ensure that individual DBAs use their individually assigned database account to perform only DBA activities.*

3.3.2 Application Object Ownership/Schema Account

Object ownership grants full privileges to owned objects. All database objects will be owned by the database system, DBAs, or by an account created especially for application object ownership. It is recommended that for each application a custom database account is created and that this account is used to own all of the database objects accessed by the application. The application user will not own any database objects. Only the application owner will have the ability to grant object privileges to the application roles. At no time will an application user log on to the database account that is the application object owner. The application object owner account will be used only for update and maintenance of the application objects. To help protect this account, the custom application owner account will be disabled when not in use. The default DBMS database accounts will not be used as the owner of an application's objects or schema.

- *(DO0150, DM1759: CAT II) The DBA will ensure that all database objects are owned by the database system, DBAs, or by an account created especially for application object ownership.*

- *(DO0150, DM1759: CAT II) The DBA will ensure that application user database accounts do not own any database objects.*

- *(DO0150, DM1759: CAT II) The IAO will ensure that DBA accounts do not own application objects.*

- *(DO0160, DM0630: CAT II) The DBA will ensure that application user database accounts do not log on to the database account that is the application's object or schema owner.*

- *(DO0160, DM0630: CAT II) The DBA will ensure that the custom application object owner account is used only for update and maintenance of the application objects.*

- *(DO0160, DM0630: CAT II) The DBA will ensure that custom application owner accounts are disabled/locked when not in use.*

- *(DO0150: CAT II) The DBA will ensure that default DBMS database accounts other than the default administration account are not used as the owner of an application's objects or schema.*

3.3.3 Default Application Accounts

All default database accounts created during installation of an application including the database system itself will have their passwords changed immediately after installation. Since a database may be created multiple times, the passwords will be changed each time a database is created. Default database accounts created during application installations that are not required for daily operation of the application will be disabled.

- *(DO3445: CAT I) The DBA will change all default database account passwords after the application installation.*

- *(DO0160: CAT II) The DBA will disable default application accounts created during application installation that are not required for daily operation of the application.*

3.3.4 Application Non-interactive/Automated Processing Accounts

Accounts created for and used by non-interactive/automated processing are subject to special consideration. These database accounts may be used for a variety of functions such as activity log storage by remote or local devices, unattended database maintenance batch jobs, etc. These accounts will not be shared with interactive database users. The primary vulnerability associated with the use of non-interactive database accounts is a frequent requirement to store the account name and password within application code or external files and the possibility of exposure of this information during the logon process. The requirements for protecting the username and password vary by operating system and database system. Accounts used for automated processing should be further protected by restricting the account used to appropriate hours of access where possible. Additional policy for securing these accounts is located in specific database and OS-specific sections.

- *(DO3504, DO3487, DO3536, DO3537: CAT II) The DBA will ensure that non-interactive/automated processing accounts meet the same security requirements as database application user database accounts meet with the exception of password lifetime.*

- *(DO3485: CAT II) The DBA will restrict the password lifetime for non-interactive/automated processing accounts to a maximum of one year.*

- *(DG0060: CAT II) The DBA will ensure that use of non-interactive/automated processing accounts is documented and authorized by the IAO.*

- *(DG0067: CAT II) The DBA will ensure that database utilities and batch submissions do not contain or store unencrypted database account names and passwords.*

3.3.5 N-Tier Application Connection Accounts

Many databases require the use of a single account to support N-Tier application connections to a backend database. Such connections rely upon the degree of security applied to the network connection and the authentication method used between the middle-tier server and the database server. Also, the ability to audit at the individual user level may be lost at the database level. It becomes the responsibility in this scenario of the application level to audit individual activities as required. The access to the N-Tier connection account will be restricted by network configuration and authentication method to the connecting middle-tier server. Acceptance of risk for the limited auditing capability of the database in this configuration will be documented and filed with the IAO.

- *(DO0360: CAT II) The DBA will ensure that access to a shared database N-Tier connection account is restricted by network configuration and authentication method to the connecting middle-server.*

- *(DG0060: CAT II) The DBA will ensure that the acceptance of risk for the limited auditing capability of the database in a shared N-Tier connection account configuration is documented and filed with the IAO.*

- *(DO0360: CAT II) The DBA will configure connections between the database server and connecting middle tier system in accordance with policy as listed in Section 5.2, Network Connections to the Database.*

3.3.6 Application User Database Accounts

Application user database accounts are used to provide access to application database objects to perform a particular application function. Privileges granted to application user database accounts will follow the principle of least privilege and include only those privileges required to perform the assigned function. Privileges will not be granted directly to application user database accounts. Privileges will be granted to application user database accounts only through the use of database roles.

- *(DG0080: CAT II) The DBA will ensure that privileges granted to application user database accounts are restricted to those required to perform the specific application functions assigned.*

- *(DO3709: CAT III) The DBA will ensure that privileges are not directly granted to database application user database accounts.*

3.4 Database Authorizations

3.4.1 Database Object Access

Each database application user database account will be granted object access to the appropriate database objects through application specific roles. These roles will be based on the function being performed. Object privileges will not be assigned directly to individual application user database accounts. Object privileges will not be granted to PUBLIC (a DBMS defined mandatory all-user role) except those explicitly required by the DBMS vendor. Access to DBA views and tables, which contain access to all data dictionary object information, will be restricted to DBAs or batch processing accounts that have been documented with the IAO.

- *(DO3709: CAT III) The DBA will ensure that all object privileges granted to application users are granted through the use of application specific roles.*

- *(DO3709: CAT III) The DBA will ensure that object privileges are not assigned directly to individual application user database accounts.*

- *(DO3689: CAT II) The DBA will ensure that application object privileges are not granted to PUBLIC.*

- *(DO3475: CAT II) The DBA will ensure that DBMS installation default object privileges are not granted to PUBLIC except for those object privileges whose removal is not supported by the DBMS vendor.*

- *(DO0310: CAT II) The DBA will ensure that access to DBA views and tables is restricted to DBAs and batch processing accounts that have been documented with the IAO.*

3.4.2 Database Roles

A database role allows database privileges to be defined and assigned by application function. Individually required privileges and other database roles may be granted to a single database role thus allowing required privileges to be simultaneously granted to and revoked from database accounts. An example of such roles would be DBA roles, application administrator roles, and specific application user roles for financial applications, sales support applications, inventory applications, etc. All application user roles will be granted the most limited set of privileges that allows the user to accomplish the specific job function required of their position. No roles will be granted to PUBLIC. No permissions will be granted directly to application user database accounts with the following exceptions: 1) accounts created and maintained by default during the installation and maintenance of the database system, and 2) a single application user database account on a database with only one such account.

- *(DO3473, DM0690: CAT II) The DBA will ensure that all application user roles are granted the most limited set of privileges that allows the user to accomplish the specific job function required of their position.*

- *(DO0320, DM1714: CAT III) The DBA will ensure that roles are not granted to PUBLIC.*

- *(DO3709, DM1714: CAT III) The DBA will ensure that no permissions are granted directly to database accounts except those granted to database application owner accounts, a single application user database account on a database where only one such account is defined, and to accounts created and maintained by default during the installation and maintenance of the database system.*

3.4.2.1 DBA Role

The DBA role contains all database system privileges. System privileges include privileges to configure the database, enable the creation, modification, and deletion of database objects, maintain database accounts, and privileges that provide the ability to grant and revoke permissions to these objects. The DBA role also includes exclusive access to database views and tables that house privileged information about the database and database structure. In other words, the DBA has full access to the database’s data dictionary. The DBA role will only be granted to authorized DBAs. DBAs will be authorized by the IAO. In a production environment, the assignment of the DBA role will be restricted to authorized DBA accounts. In a development environment, the DBA role will be restricted to DBA accounts and authorized application developer accounts.

- *(DO3440: CAT II) The DBA will ensure that the DBA role is restricted to authorized DBA accounts in a production environment.*

- *(DO3440: CAT II) The DBA will ensure that the DBA role is restricted to DBA accounts and authorized application developer accounts in a development environment.*

- *(DO3440: CAT II) The IAO will authorize all DBA accounts.*

3.4.2.1 Application Developer Roles

The application developer role is used to assign required privileges to developer accounts on a development database. Restrictions on privileges assigned to application developer accounts differ for development databases housed on a host system that also supports a production database from developer accounts on a host system that supports only a development database.

Additional restrictions are necessary on a shared production/development database host to protect the production database and resources from application development. Production software, logs, datafiles, and other directories and files will not be accessible to application developer accounts. Application developer accounts will not be defined in production databases. Application developer accounts will not have DBA privileges within a development database on a shared production/development host.

Ideally, on all databases the DBA role remains distinct from the developer role. Also, each developer account and their objects are protected from other developer accounts. An application object owner is still created for each application in development and developers submit their locally tested objects before they are moved to the test application object owner account. Privileges should still be restricted as in a shared production/development environment, but, in cases where it is not practical to have separate DBA and developer accounts such as a development system that has only a single developer that must also perform DBA functions, then a single account may be used for both. However, the developer/DBA should be aware that application testing under this single privileged account might not accurately test the application under the target security environment of the production system.

- *(DO0300: CAT II) The DBA will ensure that on a shared production/development host system, an application developer database account is not granted permission to create, alter, or drop schema objects on the development database.*

- *(DO0300: CAT II) The DBA will ensure that on a shared production/development host system, an application developer database account is not granted database system privileges on the production database.*

- *(DG0077: CAT II) The SA/DBA will ensure that developer accounts on a shared production/development host system are not granted operating system privileges to production files, directories, or database components.*

- *(DO0350: CAT II) The DBA will ensure that privileges assigned to application developer database accounts are justified and authorized by the IAO.*

- *(DO0350: CAT II) The DBA will ensure that developers are not granted system privileges within a production database.*

3.4.2.2 Application Administrator Roles

Application administrator roles are roles used to assign application user database account maintenance responsibility to users other than DBAs. In some cases, such a capability is required by a specific application. Application administrator roles may have the privileges to create application user database accounts, assign specific application profiles to individual application user database accounts, and assign application roles to application user database accounts. The application administrator role should be activated only by a specified database application or by a password provided by a stored database application. The application administrator role will not be used as the default role for application user database accounts.

- *(DO0340: CAT III) The DBA will ensure that the application administrator role is used only when performing administration functions.*

- *(DO0340: CAT III) The DBA will ensure that the application administrator role is not used as the default role for application user database accounts.*

3.4.2.3 Application User Database Roles

Each application will create distinct roles that contain all privileges necessary for users of the application, including a separate role for application administrators. If an application contains multiple job functions, individual roles will be created containing the privileges necessary to perform each of the job functions.

All application user database accounts will be granted the appropriate application role upon creation. Any application roles granted will not be assigned as the default role. This will require the application to specifically enable the role. Application user database accounts may be assigned multiple application user roles based upon their job function. An application user database account will not be granted the privilege to alter any other database account. All DBA views and tables will be secured so application user database accounts or application user database roles do not have access to this information.

- *(DO3709: CAT III) The IAO will ensure that every application with two or more application user database accounts has distinct application user database roles containing any and all privileges necessary for application users and application administrators.*

- *(DO3709: CAT III) The DBA will ensure that all application user database accounts are granted the appropriate application user database role.*

- *(DO0350: CAT II) The DBA will ensure that application user database accounts are not granted the privilege to alter any other database account.*

3.5 Protection of Sensitive Data

Sensitive data or information, as defined in *DODI 5200.40, DOD Information Technology Security Certification and Accreditation Process (DITSCAP)* and *DODD 8500.1, Information Assurance (IA)* or as determined to be sensitive by the data owner, may be stored within data objects within the database and may require additional protection. Encryption of specific data items can protect data from unauthorized viewing. Sensitive data will be encrypted within the database when supported natively by the database.

- *(DG0090: CAT II) The IAO will ensure that sensitive data is identified by the data owner and configured for encryption by the DBA when supported natively by the database.*

3.6 Protection of Stored Applications

Application source code such as that found in procedures and functions stored within the database may contain information useful to malicious users. Application code provided by COTS software is already available publicly. However, custom applications and GOTS software may still be protected by non-disclosure. Therefore, custom application and GOTS application software source code objects will be encrypted within the database where available.

- *(DG0091: CAT III) The DBA will ensure that custom application and Government-Off-The-Shelf (GOTS) source code objects are encrypted within the database when supported natively by the database.*

3.7 Protection of Database Files

Database data files contain database objects and their contents. Review of the file contents may disclose sensitive data. An effective way to protect these files in addition to host system access control is to encrypt them. This may be accomplished on Windows hosts by using the encrypt file property. There is no similar function for UNIX systems and, at the time of this release of this STIG, no available NIAP-approved third-party tools. However, such tools for UNIX and Linux are available and may be employed if they use NIST-approved algorithms and receive approval from the appropriate office (DAA if freeware/shareware).

- In some cases, encryption of the database data files may have a prohibitively negative impact on database performance. Where performance permits, database data files should be encrypted on the host system.

This page is intentionally left blank.

4 DATABASE AUDITING

This section establishes the minimum amount of auditing required for database management systems.

4.1 Precautions to Auditing

Auditing can result in a great deal of information being collected on database activities. The level of auditing will have enough granularity to allow for the monitoring of intrusive activity.

4.2 Audit Data Requirements

Auditing will be configured and implemented on all systems. Auditing will be capable of capturing all database operations. This includes both events that occur within the database and affect modification to database parameters and resources as well as modifications to the database catalog (object creation, deletion, alteration) and events performed on or by the host system such as database shutdown and startup. Database audit data will be maintained for one year. The audit data is not required to be local to the database for a year, but will be available for historical analysis if needed. Audit data will only be readable by personnel authorized by the IAO.

- *(DO3413, DM0510, DM1703, DM5268: CAT II) The IAO will ensure that auditing is configured and implemented on all systems.*

- *(DO3413, DO3610, DO3692, DM0510, DM1703, DM5268: CAT II) The DBA will ensure that audit data is captured for all required database events (see Section 4.2.1, Minimum Required Audit Operations).*

- *(DO0241: CAT II) The SA/DBA will ensure that audit data is captured for database events that are auditable at the host system level including database process or service startup/shutdown and database authentication or access.*

- *(DG0030: CAT II) The IAO will ensure that database audit data is captured and maintained for one year.*

- *(DO3446: CAT II) The DBA will ensure that audit data is only readable by personnel authorized by the IAO.*

4.2.1 Minimum Required Audit Operations

The following minimum set of operations will be audited for successful and unsuccessful execution. In the event of intrusive or anomalous activity, more detailed auditing could be performed. (Reference the DBMS specific appendixes for specific minimum database auditing requirements.) The DBA will ensure that the following minimum audit requirements are met:

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that the creation, alteration, or deletion (drop) of database accounts is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that the creation, alteration, or deletion (drop) of any database system storage structure is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that the creation, alteration, or deletion (drop) of database objects is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that the creation, alteration, or deletion (drop) of database tables is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that the creation, alteration, or deletion (drop) of database indexes is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that enabling and disabling of audit functionality is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that granting and revoking of database system level privileges is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that any action that returns an error message because the object referenced does not exist is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that any action that renames a database object is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that any action that grants or revokes object privileges from a database role or database account is audited.*

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that all modifications to the data dictionary or database system configuration are audited.*

- *(DO3692, DM1703: CAT II) The DBA will ensure that all database connection failures are audited. Where possible, the DBA will ensure that both successful and unsuccessful connection attempts are audited.*

4.2.2 DBA Auditing

All connections performed to maintain or administer the database will be audited**.** All DBA operations will be audited where possible. At a minimum, the DBA connection will be audited and the following list of DBA activities will be reported.

- Database startup
- Database shutdown
- Database online backup
- Database archiving
- Database performance statistics collection

- *(DO3692, DM0510, DM5268: CAT II) The DBA will ensure that all database connections used to perform the above listed DBA actions are audited.*

4.2.3 Value Based Auditing

Value based auditing is performed on the individual data element. It provides a before and after look of changes to data values. Value based auditing is required for all classified data and all data stored in DBMSs on MAC I and MAC II systems. DODI 8500.2 requires that access and changes to this data be recorded in transaction logs. Most databases accommodate this basic requirement by default by recording transactions in log files available for restoration of data prior to an uncommitted transaction. Transaction logs will be reviewed weekly or more frequently or immediately when security events occur for classified data and data stored on MAC I and MAC II systems. Users will be notified of the last time and date of modification to classified data.

- *(DO3610: CAT II) The DBA will ensure that access and changes to classified data or data stored within a DBMS on a MAC I or MAC II system are stored in the DBMS transaction log.*

- *(DO3610: CAT II) The IAO will ensure that DBMS transaction logs are reviewed weekly or more frequently for suspicious or unauthorized changes to classified data or data stored in a MAC I or MAC II DBMS.*

- *(DG0031: CAT II) The IAO will ensure that processes or procedures are in place to notify users of the time and date of modifications to classified data stored in the database.*

4.2.4 Required Audit Operations on Audit Data

Unauthorized users often target auditing data in order to hide evidence of unauthorized activity. The audit data will be audited for the following types of operations.

- Update of audit information
- Deletion of audit information

This level of auditing applies to the actions of all users, including DBAs.

- *(DO3610: CAT II) The DBA will ensure that database audit trail information is audited for all update and deletion operations.*

4.3 Audit Data Backup

The audit data will be maintained for a period of one year. The audit data is not required to reside within the database or reside on local disk storage, but will be available on off-line storage if needed. If an application is not capable of maintaining this amount of audit information, the deficiency will be documented and an extension requested.

The audit data will be stored in a format readable by analysis programs or scripts. This will enable the IAO to perform a historical analysis if intrusive or anomalous behavior is discovered.

When the audit data is archived (backed up) to a historical format, the archived audit information will be deleted from the database or host system. The deletion of audit records may be the responsibility of the DBA or a security administrator and requires the generation of an audit record. (This means that after the audit data has been archived, only one record will remain in the active audit trail. This will be the record of deletion of audit information.) This allows the IAO and/or the DBA to monitor the audit information and verify that the audit data has been archived.

- *(DO3610: CAT II) The DBA/SA will ensure that all audit data deletion operations cause an audit record to be generated within the active audit trail.*

4.4 Audit Data Reviews

The collection of user account actions and process activities in the audit files is only part of the process of system monitoring. Collected data will be examined and analyzed at least daily to detect any compromise or attempted compromise of system security.

The database audit data will be reviewed regularly and within a scheduled time frame. This review process will check for any intrusive activity and any anomalous activity. The IAO will ensure that the audit data and/or system logs are reviewed daily for the following:

- Excessive logon attempt failures by single or multiple database accounts
- Logons at unusual/non-duty hours
- Failed attempts to access restricted system or datafiles indicating a possible pattern of deliberate browsing
- Unusual or unauthorized activity by System Administrators
- Command-line activity by a database account that should not have that capability
- System failures or errors
- Unusual or suspicious patterns of activity

At a minimum, the DBA or security administrator will do the following:

- Provide reports on current audit data
- Provide reports on historical audit data
- Provide a methodology to back up current audit data into a historical format
- Provide a means of archiving current audit data after a backup to a historical format

It is recommended that the DBA create processes to monitor and provide real-time alerts for high priority incidents.

- *(DG0052: CAT II) The IAO will ensure that the database audit data is reviewed daily.*

4.5 Audit Data Access

All access to database audit data will be limited to the DBA, security administrator and IAO. This limited access will be strictly enforced. Access to any DBA views that allow a database account to display audit information will be restricted to DBA or security auditors. Select, insert, delete, or update operations on audit information will be restricted to DBAs or security auditors. Privileges to disable auditing will be restricted to DBAs or security auditors.

- *(DO0310: CAT II) The DBA will ensure that access to any DBA views that allow a database account to display audit information is restricted to DBAs or security auditors.*

- *(DO3446: CAT II) The DBA will ensure that select, insert, delete, or update privileges on audit information is restricted to DBAs or security auditors.*

- *(DO0350, DM0530: CAT II) The DBA will ensure that privileges to disable auditing are restricted to DBAs or security auditors.*

4.6 Database Monitoring

In addition to reviewing audit data collections, unauthorized database activity may also be discovered by actively monitoring the status of database objects. The SA/DBA will review the process list and/or system job queues daily to ensure that no unauthorized batch jobs or scripts are being run against the database. The DBA will ensure that batch jobs and scripts do not store passwords in unencrypted format. The DBA will monitor the creation, reload, and compilation of database objects to ensure no unauthorized changes have been made. Where possible, DBAs will review what applications are being used to access the database to ensure that only authorized applications are allowed access to the database and to discover unauthorized database access attempts.

The DBA will regularly and routinely monitor database accounts for expiration and inactivity. Accounts not in use will be removed in accordance with site policy.

- *(DG0050: CAT IV) The DBA will monitor the database for unauthorized changes to database objects.*

- *(DG0051: CAT II) The DBA will monitor database batch and job queues to ensure that no unauthorized jobs are accessing the database.*

- *(DG0074: CAT II) The DBA will monitor database account expiration and inactivity and remove expired and inactive accounts in accordance with site policy.*

- *(DG0052: CAT IV) The DBA will monitor the database to discover access by unauthorized application software.*

5 NETWORK ACCESS AND REQUIREMENTS

This section describes the requirements for network security as it relates to all database management systems.

5.1 Protection of Database Identification Parameters

Database parameters such as instance identifiers, network addresses, and database host names may aid unauthorized users in finding and accessing databases. This information should never be published publicly such as via mailing lists or newsgroups and should be protected from unauthorized access where possible. Database information for specific databases will be restricted to authorized database users of that particular database and not disseminated to all users in the network or environment.

- *(DG0053: CAT III) The IAO will ensure that database client software includes only database identification parameters of databases to which that user is authorized access.*

5.2 Network Connections to the Database

When a database connection is requested via the network to a database server, the client will provide an individual account name and authentication credentials to access the database. The database account name and any password transmission from a client to a database server over a network will be encrypted.

- *(DO3698: CAT II) The IAO will ensure that the transmission of a database account name and password from a client to a database server over a network is encrypted.*

- *(DG0060: CAT II) The IAO will ensure that all network connections to a database require an individual database account and authentication credentials.*

5.2.1 Remote Administrative Database Access

Remote connections to the database by administrative users to perform administrative functions including database account password resets and account management will be encrypted. Without encryption, such activity performed during these connections could provide information useful to gain unauthorized access to the database.

- *(DG0093: CAT II) The DBA will ensure that remote administrative connections to the database are encrypted.*

5.2.2 Open Database Connectivity (ODBC)

ODBC, an application programming interface (API), provides another method besides native database methods to connect to a database and issue standard SQL commands. Many popular COTS applications provide connectivity to databases using ODBC drivers. These applications may be easily configured to access the database by a savvy user who can provide authentication information directly to the database. These types of connections emphasize the importance of defining database object access authorizations within the database itself and not solely within the application. ODBC tracing, a function used for debugging applications and connections, will be disabled or tightly controlled. This will prevent sensitive data from being stored in trace files on disk during debugging activities. Where its use is not justified, the ODBC tracing executable will be deleted from the system to ensure the function is unavailable. The removal of the executable may require periodic review as some upgrades or other application installations may re-install it. User account privileges defined at the DBMS level are the only privileges guaranteed in effect via ODBC connections. Database account passwords will not be stored in unencrypted format within ODBC connection definitions or data set names (DSN).

- *(DG0018: CAT III) The DBA will ensure that when not in use the ODBC tracing executable is deleted from the system to ensure the function is unavailable*.

- *(DG0067: CAT II) The IAO will ensure that unencrypted passwords are not stored in ODBC connection definitions.*

5.2.3 Java Database Connectivity (JDBC)

JDBC, another application programming interface (API), provides a method for connection to a database from a Java application. JDBC drivers may connect to a database by bridging to ODBC drivers or by using the database native SQL API such as Oracle's Net (Net8) or Microsoft's DB library. JDBC connection information, specifically database account passwords, will not be stored in unencrypted format.

- *(DG0094: CAT II) The IAO will ensure that JDBC connection data is not stored in unencrypted format.*

5.2.4 Web Server or Middle-Tier Connections to Databases

The availability requirements for the application served by the database and the sensitivity of the data being served determine the appropriate network architecture for web or middle-tier connections to DBMSs. Systems requiring greater assurance for availability (MAC I and MAC II) should have the DBMS located on a host server separate from the serving web or middle-tier server. This helps limit any security events to the compromised system. Encryption requirements for data transmitted between these systems is dependent on the sensitivity of the data being transmitted, the sensitivity level assigned to the network being traversed, and any differences in need-to-know between the data and the users on the network. Login credentials to the DBMS and web/middle-tier servers will always be encrypted. Sensitive data traversing a public network will always be encrypted. All encryption will use FIPS 140-2 validated cryptography.

A simple configuration to provide the required level of protection is shown in the figure below. This configuration shows the DBMS servers located on a dedicated screened subnet separate from the web/middle-tier server screened subnet. Connections by external users are restricted to web/middle-tier servers, but internal users on the private network may access the DBMS directly or via the web/middle-tier server. Access to DBMSs on the screened subnet is restricted to specific web/middle-tier servers and the users or servers on the private network. Other architectures or methods of separation may be used. This diagram depicts only one method to accomplish recommended security. Please consult the DISA *Enclave STIG* and DISA *Network Infrastructure STIG* for detailed networking requirements.

Connection pooling between web or middle-tier servers and databases that support individual identification and authentication of database users should be used if available. Currently, Oracle databases support this by means of the Oracle Call Interface (OCI).

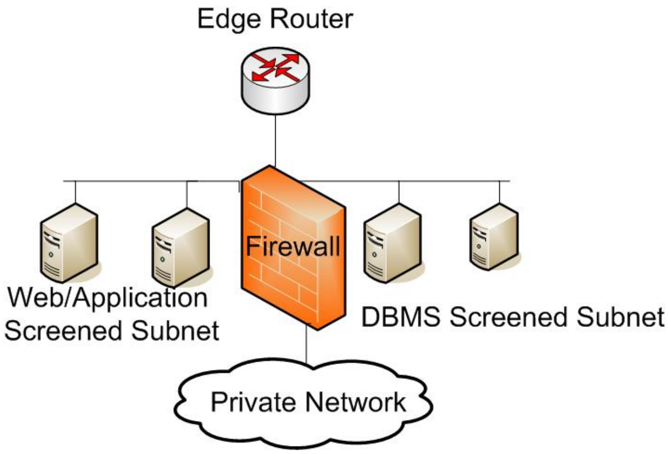

Figure 1. Network Architecture to Support External Users

5.2.5   Database Session Inactivity Time Out

Inactive database sessions are typically construed as an indication of unattended interactive account connections. While workstation and OS policy already require terminal lock requirements to protect unattended workstations from unauthorized access, an idle database session still uses database and host system resources and may lead to denial of service or session hijacking. Where available, database session inactivity time outs will be set to a system-wide (in this case, a database-wide) inactivity time out of 15 minutes with allowances for specific, authorized accounts being allowed a maximum inactivity of 60 minutes. Specific accounts requiring inactivity timeouts greater than 60 minutes will be justified and documented with the IAO.

- *(DO3536: CAT III) The DBA will set a default database session inactivity time out of 15 minutes or less unless an IAM approved letter of justification to exceed this limit is available.*

- *(DO3536: CAT III) The IAO will justify and approve database session inactivity timeouts for specific accounts that require a limit in excess of 60 minutes.*

- *(DO0350: CAT II) The DBA will deny to all database accounts the privilege to disable their time out unless the exception is justified and documented with the IAO.*

5.3 Database Replication

A distinct database account and password for the replication administrator and replication system database accounts will be used to secure replication procedures and facilities. The database account password will be encrypted when transmitted over a network. Access to replication procedures and facilities will be restricted to authorized DBAs and designated replication accounts.

Replication data may be stored temporarily in specific OS locations for retrieval by database replication partners. This data must be securely stored and access to it restricted to authorized replication components. This protection is controlled by OS file and directory permissions as well as database security controls. The structure and configuration of the replication architecture for the database should be clearly documented and include approval for dissemination of data as required.

- *(DO0210: CAT IV) The DBA will ensure that a distinct database account and password is used to secure the replication procedures and facilities.*

- *(DO0210, DM2133: CAT IV) The DBA will ensure that access to replication procedures and facilities is restricted to authorized DBAs and designated replication database accounts.*

5.4 Database Links

Databases may be configured to share data across remote database systems. Such linked databases may be part of federated or distributed database architectures. Authentication between linked databases may rely upon the credentials of the requesting database session or upon a statically defined username and password. Generally, it is best to use a current user’s network-based security credentials provided by means of a directory service to authenticate to the remote system, as this is the only way to maintain a clear auditable identity across all systems. When distributed databases are required to share or transmit data through database links via network connections, the database initiating the link will use the credentials of the current database session to connect to the remote database.

Database link access will be restricted to authorized users when possible. Applications will not create or use public database links unless justified and documented with the IAO with the exception of replication database links as required. It is possible to create a private database link and establish a public synonym to the database link. This protects the details about the database link, while providing the same capabilities as a public link.

To protect sensitive production database data, database links will not be defined between production and development databases. This restriction helps to prevent sensitive data from being access or downloaded for review by developers on a remote database.

- *(DO0250: CAT III) The DBA will ensure that database links use the credentials of the current database session to authenticate to the remote database system except as required by replication configurations.*

- *(DO0250, DM2142: CAT III) The DBA will ensure that public database links are not used by applications unless justified and documented with the IAO, with the exception of replication database links as required.*

- *(DG0075: CAT II) The DBA will ensure that no database links are defined between production and development databases.*

This page is intentionally left blank.

6 OPERATING SYSTEM

6.1 Database File Access

Database executable files, configuration files, and datafiles will be protected by the operating system from unauthorized access. Access to these files will be configured according to recommended security guidance provided by the DBMS software vendor. Local application user OS accounts will be granted access only to the DBMS application files required by the DBMS. Datafiles should be encrypted where possible to further protect the data stored within them from unauthorized disclosure. This can be accomplished using native OS file or device encryption. In some cases, the performance impact of the encryption prohibits its use.

- *(DO3613: CAT II; DM3769: CAT I) The DBA/SA will ensure that access to DBMS executable files, configuration files, and datafiles is defined in accordance with vendor security recommendations.*

- *(DO3613: CAT II; DM3769: CAT I) The DBA/SA will ensure that local application user OS accounts are granted access only to the DBMS application files required to access the DBMS.*

6.2 Local Database Accounts

Database applications that are installed on the DBMS host system and are accessed locally by application users may require a specific security configuration. Specific details for any OS account configuration requirements are located in the specific DBMS sections.

6.3 Database Administration Accounts

Database administration OS accounts required for operation and maintenance of the DBMS will be configured in accordance with the security requirements as specified by the specific DBMS. Details for the DBA account configurations are located in the specific DBMS appendix.

- *(DG0005: CAT II) The SA/DBA will ensure that database administration OS accounts required for operation and maintenance of the DBMS are configured in accordance with the security requirements specified for the specific DBMS.*

6.4 Database OS Groups

Typically, DBMS host configurations grant privileged access to DBAs by means of special OS group assignments. Membership in this group grants access privileges to DBMS directories and files. It also may grant special privileges within the database system itself. Only authorized DBAs will be granted membership to DBMS privileged OS groups. Some DBMS installations may require installation under an account created specifically for that purpose. Access to any shared DBMS software installation accounts will be restricted to authorized personnel only. Use of a shared DBMS software installation account will be logged and/or audited to indicate the identity of the person who accessed the account.

- *(DO0140: CAT I) The IAO will ensure that only authorized DBAs are granted membership to DBMS privileged OS groups.*

- *(DG0040: CAT III) The IAO will ensure that access to any shared DBMS software installation account is restricted to authorized personnel only.*

- *(DG0040: CAT III) The IAO will ensure that use of a shared DBMS software installation account is logged and/or audited to indicate the identity of the person who accessed the account.*

APPENDIX A RELATED PUBLICATIONS

Government Publications:

Department of Defense Directive 8500.1, “Information Assurance”, 24 October 2002.

Department of Defense Instruction 8500.2, “Information Assurance (IA) Implementation,” 6 February 2003.

Department of Defense Directive 5200.40, “DOD Information Technology Security and Accreditation Process (DITSCAP)” 30 December 1997.

Chairman of the Joint Chiefs of Staff (CJCS) Manual 6510.01, "Defense-in-Depth: Information Assurance (IA) and Computer Network Defense (CND)," 15 March 2002.

DOD Instruction 8520.2, "Public Key Infrastructure (PKI) and Public Key (PK) Enabling," 01 April 2004.

NSA Guide to the Secure Configuration and Administration of Oracle9i Database Server, 02 October 2003.

NSA Guide to Secure Configuration and Administration of Microsoft SQL Server 2000, 02 October 2003.

Defense Information Systems Agency, Enclave Security Technical Implementation Guide.

Defense Information Systems Agency, Network Infrastructure Security Technical Implementation Guide.

Defense Information Systems Agency, Windows NT/2000/XP Addendum.

Defense Information Systems Agency, UNIX Security Technical Implementation Guide.

Defense Information Systems Agency, OS/390 Security Technical Implementation Guide.

Defense Information Systems Agency, Web Server Security Technical Implementation Guide.

DBMS Vendor Publications:

Oracle 9i, Installation Guide, Release 2 (9.2.0.1.0) for UNIX Systems: AIX-Based Systems, Compaq Tru64 UNIX, HP9000 Series HP-UX, Linux Intel, and Sun Solaris, May 2002, Part No. A96167-01.

Oracle 9i, Database Installation Guide Release 2 (9.2.0.1.0) for Windows, May 2002, Part No. A95493-01.

Oracle 9i Enterprise Edition, Installation Guide, Release 1 (9.0.1) for OS/390, May 2001, Part No. A89900-01.

Oracle 9i Net Services, Reference Guide, Release 2 (9.2), October 2002, Part No. A96581-02.

Oracle 9i, Database Administrator’s Guide Release 1 (9.0.1) for Windows, June 2001, Part No. A90164-01.

Oracle 9i, Administrator’s Reference, Release 2 (9.2.0.1.0) for UNIX Systems: AIX-Based Systems, Compaq Tru64 UNIX, HP 9000 Series HP-UX, Linux Intel, and Sun Solaris, May 2002, A97297-01.

Oracle 9i Enterprise Edition, System Administration Guide Release 2 (9.2.0.1.0) for OS/390, May 2002, Part No. A97313-01.

Oracle 9i, Security Overview, Release 1 (9.0.1), June 2001, Part No. A90148-01.

Hack Proofing Oracle, Howard Smith, Oracle Corporation UK Limited, Paper presented at the Oracle Open World Conference, San Francisco, CA, October 2000.

A Security Checklist for Oracle 9i, An Oracle White Paper, March 2001, Author: Rajiv Sinha.

Microsoft SQL Server 2000, SQL Server Books Online, Version 8.00.00.

IBM DB2 Universal Database, Administration Guide: Performance, Version 8.

IBM DB2 Universal Database, Administration Guide: Implementation, Version 8.

IBM DB2 Universal Database, Installation and Configuration Supplement, Version 8.

IBM DB2 Universal Database, Federated Systems Guide, Version 8.

IBM DB2 Connect, Connect User's Guide, Version 8.

Other Publications:

The Center for Internet Security, Level Two Benchmark (Draft) SQL Server 2000, V0.6.

The Center for Internet Security, Oracle Security Benchmark, V1.1, May 2004.

SANS “Securing Oracle Step-by-Step”, January 2003.

This page is intentionally left blank.

APPENDIX B ORACLE SPECIFIC POLICY AND IMPLEMENTATION

B.1 Current Oracle Version

The information contained in this appendix is specific to Oracle Versions 8.1.x (8i), 9.2 (9i) and 10.1.0 (10g). When version-specific information is presented, it will be labeled with the version to which it specifically applies.

Oracle Support Services provides bulletins on Product Obsolescence and Desupport Notices on all platforms. These notices identify when Extended Maintenance Support (EMS), Extended Assistance Support (EAS), and Error Correction Support (ECS) end. ECS provides error correction for new problems. EAS and EMS provide telephone and electronic support and fixes for known problems. EAS and EMS do not provide error correction for new problems.

To protect your Oracle environment, the Oracle DBMS version will be an Oracle Support Services fully supported version. Oracle Support Services reported that Version 7.3.4 ECS support ended 31 December 2000. ECS support for Oracle Version 8.0.6, the last release of Oracle Version 8, ended 30 September 2001. ECS support for the terminal release of Oracle 8i, version 8.1.7, ends 31 December 2004 on most commonly used platforms. ECS support for Oracle 9i, version 9.0.1.x ended 31 December 2003. ECS support for Oracle 9i, version 9 Release 2 (9.2) is scheduled to end 31 December 2005. No end-of-support date has been published for Oracle 10g, version 10.1.0. The installed Oracle software will have all security patches applied.

- *(DO0100: CAT I) The IAO will ensure that unsupported DBMS software is removed or upgraded prior to a vendor dropping support.*

- *(DG0002: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(DO0100: CAT I) The IAO will ensure that the Oracle version has all patch sets applied.*

B.2 Oracle Security Evaluations

Oracle server versions have been evaluated against the TCSEC C2, ITSEC E3/F-C2 security criteria and the Common Criteria (CC) (ISO15408) EAL4 criteria. Following is a list of some of the evaluated versions and the criteria used:

- Oracle 9i, Release 9.2.0.1.0, CC EAL4, platform Solaris 8, NT 4.0
- Oracle 8i, Release 8.1.7, CC EAL4, platform Solaris 8, NT 4.0
- Oracle 8, Release 8.0.5, CC EAL4, platform NT 4.0
- Oracle 7, Release 7.3.4, ITSEC E3/F-C2, platform NT 4.0
- Oracle 7, Release 7.0.13.1, TCSEC C2 & B1, platform HP-UX BLS 8.0.4

B.3 Oracle Component Services

A default installation of Oracle includes the installation of several Oracle components. All of these components may not be required by your system. Review installed components and remove any components you do not need. For example, if you are not using JAVA, XML, InterMedia, Replication, etc., components, then remove them by using the Oracle installer. The Oracle components in the list below are required for standard Oracle database operation. Components other than those in the table below will be removed unless specifically required to support the operation of any database applications.

REQUIRED ORACLE DATABASE COMPONENTS
Assistant Common Files
Generic Connectivity Common Files
Generic Connectivity Using Open Database Connectivity (ODBC)
Oracle Net
Oracle Net Listener
Oracle Net Manager
Oracle Net Required Support Files
Oracle Net Services
Oracle Core Required Support Files
Oracle Call Interface
Oracle9i/8i/10g
Oracle9i/8i/10g Database
Oracle9i/8i/10g Development Kit
Oracle9i/8i/10g Windows Documentation (Windows only)
Parser Generator Required Support Files
Programming Language / Structured Query Language (PL/SQL)
PL/SQL Embedded Gateway
PL/SQL Required Support Files
Platform Required Support Files
Relational Database Management System (RDBMS) Required Support Files
Required Support Files

Table 2. REQUIRED ORACLE DATABASE COMPONENTS

B.4 Oracle Access Controls

Access controls of database objects are an integrated feature of the Oracle DBMS. Oracle defines two types of database accounts: administrative and non-administrative. Administrative database accounts are granted full privileges to manage the database structure, database objects, and other database accounts.

Privileges are divided into two types—system privileges and object privileges. System privileges permit the database account to exercise administrative functions. Object privileges permit the database account to read, modify, and delete data within already existing database objects or execute application program objects. Access to data objects may be further controlled by use of database views. Views define a subset of stored data that may be accessed by database accounts. Database accounts may also be further restricted to updates and inserts on specific columns or fields of data instead of entire records. Also, using Oracle's fine-grained access control, access policies may be defined and associated with tables and views that allow retrieval of subsets of records. This eliminates the requirement to grant database accounts access to all records in a table or view and is enforced at the database level as opposed to being controlled through the application. Oracle also provides the ability to define an application context. The application context activates access controls for only the specified application. The combination of fine-grained access and application context constitutes Oracle's Virtual Private Database (VPD).

Database account access to resources is controlled through use of Oracle profiles. Profiles define database account resource quotas and password management guidelines.

B.4.1 Oracle Identification and Authentication

Oracle supports the following authentication options:

- Oracle's own password-based authentication
- Host-based or OS authentication (by the underlying operating system)
- Global authentication and authorization (includes SSL and directory services)
- N-Tier authentication that supports proxy authentication to the database.

B.4.1.1 Database Authentication

Database authentication by the Oracle database requires use of a password. Oracle stores account passwords in encrypted format within the database. The table that stores the passwords is restricted from direct access by any database account. Oracle supports individual database accounts for each user. The logon process to Oracle database accounts is encrypted by default.

B.4.1.2 Operating System Authentication

Oracle supports authentication of database users by the host operating system. Users must have a corresponding account defined in the database that specifies external authentication by the host operating system. Host operating system authentication is NOT supported via the network unless specifically enabled (see REMOTE_OS_AUTHENT under INITIALIZATION PARAMETERS below). If enabled, Oracle assumes that any operating system that has authenticated the user is trusted and will allow access to a database account by the same name without authentication. Operating system authentication will NOT be enabled via the network. When operating system authentication is specified, users will be required to authenticate to the database host system prior to connecting to the database. Windows authentication allows Windows domain accounts to access the database as locally authenticated users. However, in order to increase security, Oracle will be configured to require use of the domain name to prevent users with the same account name in different domains to access the database as a shared account. Users authenticated externally by Windows will be identified with the domain name prefix.

- *(DO3538: CAT I) The DBA will disable operating system authentication via the network.*

- *(DO0370: CAT II) The DBA will configure the Oracle database to require the Windows domain prefix for database accounts authenticated externally by Windows.*

- *(DO0370: CAT II) The DBA will set the registry value OSAUTH_PREFIX_DOMAIN in HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOMEID to TRUE for database versions prior to 8.1.x using Windows authentication.*

B.4.1.3 Network Service Authentication

The Oracle Advanced Security option enables secure authentication by external third-party network services such as Kerberos, token cards, smartcards, and biometric devices. Many network authentication services also offer integration with directory services.

B.4.1.4 Global Authentication

Oracle defines global authentication as authentication via SSL and an external, centralized directory service. Database accounts are then defined as global users. These network authentication services provide a single sign-on capability that can authorize single user access to multiple systems and databases. Users authenticate themselves once to a central service, and may then connect to multiple applications or databases without providing additional credentials.

B.4.1.5 N-Tier or Proxy Authentication

Users accessing a database through a middle-tier or web application server can authenticate in one of the following ways:

1. Authenticate the user to the middle-tier server and have the middle-tier server authenticate to the database using a single logon for all users or an individual logon for each user using the individual credentials.

2. Authenticate to a global source from the middle-tier server that relays the user authentication credentials directly to the database. Authentication occurs via a directory service to the middle tier and database.

3. Authenticate directly to the database through the middle-tier server.

Although commonly in use, the first configuration listed above using a single logon for all users is the least secure because it does not allow for individual accountability within the database. To allow a middle-tier server to authenticate a user and preserve that user's individual identity and authorizations within the database, Oracle provides the Oracle Call Interface (OCI).

B.4.2 Oracle Connection Pooling

Connection pooling allows multiple users access to a database system across a shared connection. The use of a shared connection reduces the traditional amount of database server resources required to establish multiple individual connections. Oracle's connection pooling facility is additionally able to preserve individual user accountability across the pooled connection.

Third-party connection pooling mechanisms require use of a single, specific database account connection. Use of a single account precludes any ability for the database to manage authorizations or provide individual accountability for actions taken within the database. Where connection pooling is required, Oracle's connection pooling facility should be considered for use.

B.4.3 Secure Distributed Computing

Mutual authentication of databases enables secure distributed transactions between application servers, web servers, and database servers without compromising the user's credentials. Mutual database authentication and strong user authentication are accomplished by using directory service authentication. Directory service authentication ensures that the identity of any database account requesting data from a remote server will be traceable across audit trails on the accessed databases.

Oracle defines three different types of connections between databases (database links) — CURRENT_USER, connected user, and fixed user. CURRENT_USER database links use the authentication credentials of the global user, which is an account authenticated by a directory service. A "connected user" database link, which does not specify a database account, uses the credentials of the currently connected database account to connect to the remote database. A "fixed user" database link uses the database account name and password stored with the database link definition stored in the SYS.LINK$ table. Passwords stored with fixed user database link definitions are stored in clear text and are subject to unauthorized access. Fixed user database link connection definitions will not be used unless justified and documented with the IAO. Some replication configurations require use of fixed user database links. Access to the SYS.LINK$ table will be restricted to authorized DBAs.

- *(DO0250: CAT III) The DBA will deny use of fixed user database links unless they are justified and documented with the IAO.*

- *(DO3686: CAT I) The DBA will restrict access to the SYS.LINK$ table to authorized DBAs only.*

B.4.4 Oracle Administrative Connections

Oracle supports two different methods for authenticating administrative users to the database: OS and password file authentication. Once authenticated as an administrative user, privileges to manage the database including database startup, shutdown, backup, and recovery are granted by default.

OS administrative authentication requires membership in a specific OS group (see your specific Oracle OS installation guide for the OS group name) on Windows or UNIX hosts. On an OS/390 host, the granting of read authorization to the SAF-defined resource specified at installation (ssn.service.OPER, ssn.service.DBA) for an OSDI Oracle installation or a granting of privilege to a TSO account to MPM for Oracle MPM installations authorizes this authentication. Password file authentication allows remote users to access the database as an administrative user.

Prior to version 9i, Oracle allowed access to the SYS database account using the “connect internal” authentication. This access was granted to authenticated host accounts that were members of the host-specific operating system group (SYSDBA or ORADBA). Use of this account does not allow for individual accountability. Furthermore, actions performed as database account SYS or when administrators connect “AS SYDBA” or “AS SYSOPER” are not audited by Oracle’s auditing facility in Oracle versions earlier than 9.2.

Oracle administrative connections (SYS, “connect internal,” as SYSDBA/SYSOPER) will only be used to perform administrative functions available exclusively to an administrative connection. The Oracle administrative connection will not be used to perform everyday operations. Any use of an Oracle administrative connection will be documented for review by the IAO. By default, Oracle records administrative connections in the OS audit log.

Examples of appropriate operations requiring use of an administrative connection include installation, database creation, backup and recovery, database startup, and database shutdown. The use of administrative connections is not recommended for automated procedures or utilities that perform automated functions for the DBA. The ability to authenticate to the database with an administrative connection will be restricted to authorized DBAs.

A password file will not be used unless remote database administration is justified and required. In such cases, its use will be authorized and documented by the IAO. If remote administration is required, the password file will be used in exclusive mode. Exclusive mode requires individual account authentication and restricts assignment of database administrative privileges to accounts granted the SYSDBA privilege. Where remote administration is required, a password file will be used in exclusive mode.

- *(DO3440: CAT II) The DBA will ensure that Oracle administrative connections are used solely to perform administrative functions available only through an administrative connection.*

- *(DO3440: CAT II) The DBA will ensure that Oracle administrative connections are not used to perform everyday operations.*

- *(DO3440: CAT II) The DBA will ensure that the ability to authenticate to the database with an administrative connection is restricted to authorized DBAs.*

B.4.5 Oracle Administrative OS Groups

On UNIX and Windows systems that support Oracle administrative connections using OS authentication, OS group membership defines which OS accounts are granted the ability to use an administrative (SYSDBA/SYSOPER) connection.

- *(DO0140: CAT 1I) The SA will ensure that only authorized DBAs are granted membership to the Oracle administrative OS group.*

B.4.6 Default Oracle Accounts

Access to the default Oracle accounts and their associated passwords will be restricted to site-authorized DBAs only. Default accounts, particularly the SYS and SYSTEM accounts, will be used only for database update and maintenance. Default accounts will not be used for daily operations except for automated procedures that require use of the SYS or SYSTEM schema. Such automated procedures will be documented with the IAO. The SYS account is the owner of all Oracle data dictionary objects. Oracle default accounts will not be used as the owner of any non-default application schema. Administration personnel will not use Oracle default accounts for daily operations. No individual accountability is provided when multiple users use shared accounts. Individual accounts with administrator privileges will be maintained for each DBA for performing administrative functions on the database. Each DBA will log on using their individual account to perform everyday administrative functions on the database, thus providing an audit record of all activity. Oracle default accounts will be locked and expired when not required for daily operation of the database. Default accounts created for demonstration applications will be removed.

- *(DO0140: CAT 1I) The DBA will ensure that access to the default Oracle accounts and their associated passwords is restricted to site-authorized DBAs.*

- *(DO0140: CAT II) The DBA will ensure that default accounts are used only for database installation, update, and maintenance.*

- *(DO0140: CAT II) The DBA will ensure that the default accounts are not used as the owner of any application schema outside their default function.*

- *(DO0140: CAT II) The DBA will ensure that the Oracle default accounts are not used by DBAs for standard DBA operations.*

- *(DO0140: CAT II) The DBA will ensure that Oracle SYS or SYSTEM account access required by automated processes is documented with the IAO.*

- *(DO3440: CAT II) The DBA will ensure that an individual account with administrator privileges is maintained for each DBA performing everyday administrative functions on the database.*

- *(DO3445: CAT I) The DBA will lock and expire Oracle default accounts when they are not required for regular operation of the database.*

- *(DO0400: CAT II) The DBA will remove default accounts and objects created for demonstration applications.*

B.4.7 Default Oracle Passwords

The default passwords for accounts created during Oracle database creation will be changed after installation. A list of known default accounts is listed below. A more complete list may be found in *Appendix F, Oracle Database STIG Compliance Configuration.* A database may be created multiple times for the purposes of complete database reorganization. Default account passwords will be changed each time a database is created.

Following is a list of Oracle accounts reported by Oracle to not allow password modification if product operation is to be maintained:

- ODSCOMMON (present for Oracle Internet Directory (OID))
- AURORA$JIS$UTILITY$, AURORA$ORB$UNAUTHENTICATED
- OSE$HTTP$ADMIN, PORTAL30_SSO
- PORTAL30_SSO_PUBLIC

Please see Oracle Metalink Note 234712.1 for detailed information. At this time, use of default passwords for these accounts will be assigned a severity category code of 2 and extensions are required until such time that Oracle updates the software to allow for modification of the passwords.

ORACLE DEFAULT ACCOUNTS	
ADAMS	ORDSYS
AURORA$JIS$UTILITY$	OSE$HTTP$ADMIN
AURORA$ORB$UNAUTHENTICATED	OUTLN
BLAKE	PM
CLARK	QS
CTXSYS	QS_ADM
DBSNMP	QS_CB
HR	QS_CBADM
JONES	QS_CS
LBACSYS	QS_ES
MDSYS	QS_OS
OE	QS_WS
OLAPDBA	SCOTT
OLAPSVR	SH
OLAPSYS	SYS
ORDPLUGINS	SYSTEM

Table 3. ORACLE DEFAULT ACCOUNTS

- *(DO3445: CAT I) The DBA will change default Oracle account passwords immediately after installation.*

B.4.8 Oracle Password Management Requirements

Oracle Version 8 and later provides password management capability. The default password management settings in Oracle do not meet DOD requirements; therefore password settings must be manually configured to meet DOD policy and the password specifications laid out in *Section 3.2.1, Password Guidelines*. The password management settings will be applied to all database account profiles defined in the database.

- *(DO3485, DO3487, DO3504: CAT II) The DBA will configure the Oracle password management settings to adhere to DOD password policy.*

- *(DO3485, DO3487, DO3504: CAT II) The DBA will configure all Oracle database account profiles, including the default profile, to adhere to DOD password policy.*

B.5 Oracle Authorizations

In order to access a database object in Oracle, the database account must either own the object or have been granted access to the object. Oracle calls privileges assigned using a database role *indirect* assignment of privileges and privileges assigned to an account without use of a role *direct* assignment of privileges. When the account owns the object, then the account has full (and direct) privileges to the object, i.e., the account may alter the structure, the content, the existence, and access to other accounts to the object. When an account does not own the object, accounts may be granted access to the object by the owner of the object or by an account granted the authority (DBA or account granted WITH GRANT OPTION) to grant access to the object.

A database account may also be granted access to objects indirectly by means of a database role. A database role has access privileges granted directly to it or may have other roles with required permissions granted to it. The database account is then granted membership to the role. Database accounts may be granted roles by the DBA or by an account granted the role with the WITH ADMIN option.

Access to objects called from within a procedure must be granted either directly (cannot be assigned by use of a database role) to the owner of the procedure or to the account executing the procedure (directly or indirectly). By default, when a procedure, function, or package is created in Oracle, the package uses the privileges assigned to the owner to define the security. In other words, the procedure when executed will run in the security context of the procedure owner. Oracle calls this *definer's rights*.

The simplest and most secure method to assign privileges is to have a single object owner for all objects accessed by a single application. All procedures and functions supporting the application also have the same owner and use definer's rights. This negates the requirement to assign a long and potentially complicated list of privilege assignments to the objects. If database objects referenced in a procedure or function have multiple owners, then the owner of the procedure must have all object privileges assigned directly to them. Frequently, these privileges require the WITH GRANT OPTION in addition to the simple access assignment such as in the case where a view references tables owned by multiple users.

The alternative to procedures compiled using definer's rights is to specify use of invoker's rights. Invoker's rights use the rights assigned to the account executing the procedure or function. Both direct and indirect privileges are recognized. However, invoker's rights may become complicated when calls are made from one owner's procedure to procedures or functions owned by other accounts.

The privileges required for application users to successfully execute an application must be designed at application development time. If privilege assignment is not taken into account during the design phase, there may be no solution to privilege assignments at variance with the requirements of this STIG. These cannot be rectified at the database assignment level without disabling the application.

B.5.1 Oracle Predefined Roles

A role groups several privileges and roles together so that they can be granted and revoked simultaneously from database accounts. Oracle creates by default several predefined roles. These roles are created and maintained by Oracle and should not be used to assign privileges to custom database accounts with the exception of the DBA role. Oracle may at their discretion assign privileges to default roles that are not required by custom roles. It is the responsibility of the DBA to ensure that no unnecessary privileges are granted to custom database accounts. Therefore, assignment of Oracle predefined roles will be restricted to Oracle default accounts with the exception of the DBA role. Oracle predefined roles will not be granted to PUBLIC. Following is a list of Oracle predefined roles:

ORACLE PREDEFINED ROLES
AQ_ADMINISTRATOR_ROLE
AQ_USER_ROLE
CONNECT
CTXAPP
DELETE_CATALOG_ROLE
EXECUTE_CATALOG_ROLE
EXP_FULL_DATABASE
GLOBAL_AQ_USER_ROLE
HS_ADMIN_ROLE
IMP_FULL_DATABASE
JAVADEBUGPRIV
JAVAIDPRIV
JAVASYSPRIV
JAVAUSERPRIV
JAVA_ADMIN
JAVA_DEPLOY
OEM_MONITOR
OLAP_DBA
RECOVERY_CATALOG_OWNER
RESOURCE
SELECT_CATALOG_ROLE
WKADMIN
WKUSER
WM_ADMIN_ROLE

Table 4. ORACLE PREDEFINED ROLES

- *(DO0320: CAT II) The DBA will ensure that Oracle predefined roles are not granted to PUBLIC.*

- *(DO0170: CAT III) The DBA will restrict the assignment of Oracle predefined roles to Oracle default accounts with the exception of the DBA role.*

B.5.2 System Privileges

Oracle system privileges are used to grant elevated privileges within the database. The following list of Oracle system privileges will only be granted to DBAs, application object owner accounts, and predefined accounts as applied by default during installation, and application administration accounts in a production environment. In a development environment, application developer accounts may be granted any of the listed system privileges as required except privileges that allow them to access other schema objects (the "ANY" privileges). Application administrators may be granted the minimum privileges to create and maintain application user database accounts. Oracle system privileges will not be granted to PUBLIC. This list includes all system privileges available except CREATE SESSION.

ORACLE SYSTEM PRIVILEGES	
ADMINISTER DATABASE TRIGGER	CREATE TYPE
ADMINISTER RESOURCE MANAGER	CREATE USER
ADMINISTER SECURITY	CREATE VIEW
ALTER ANY CLUSTER	DEBUG ANY PROCEDURE
ALTER ANY DIMENSION	DEBUG CONNECT ANY
ALTER ANY INDEX	DEBUG CONNECT SESSION
ALTER ANY INDEXTYPE	DEBUG CONNECT USER
ALTER ANY LIBRARY	DELETE ANY TABLE
ALTER ANY OPERATOR	DEQUEUE ANY QUEUE
ALTER ANY OUTLINE	DROP ANY CLUSTER
ALTER ANY PROCEDURE	DROP ANY CONTEXT
ALTER ANY ROLE	DROP ANY DIMENSION
ALTER ANY SECURITY PROFILE	DROP ANY DIRECTORY
ALTER ANY SEQUENCE	DROP ANY INDEX
ALTER ANY SNAPSHOT	DROP ANY INDEXTYPE
ALTER ANY TABLE	DROP ANY LIBRARY
ALTER ANY TRIGGER	DROP ANY OPERATOR
ALTER ANY TYPE	DROP ANY OUTLINE
ALTER DATABASE	DROP ANY PROCEDURE
ALTER PROFILE	DROP ANY ROLE
ALTER RESOURCE COST	DROP ANY SECURITY PROFILE
ALTER ROLLBACK SEGMENT	DROP ANY SEQUENCE
ALTER SESSION	DROP ANY SNAPSHOT
ALTER SYSTEM	DROP ANY SYNONYM
ALTER TABLESPACE	DROP ANY TABLE
ALTER USER	DROP ANY TRIGGER
ANALYZE ANY	DROP ANY TYPE
AUDIT ANY	DROP ANY VIEW
AUDIT SYSTEM	DROP PROFILE
BACKUP ANY TABLE	DROP PUBLIC DATABASE LINK

ORACLE SYSTEM PRIVILEGES	
BECOME USER	DROP PUBLIC SYNONYM
COMMENT ANY TABLE	DROP ROLLBACK SEGMENT
CREATE ANY CLUSTER	DROP TABLESPACE
CREATE ANY CONTEXT	DROP USER
CREATE ANY DIMENSION	ENQUEUE ANY QUEUE
CREATE ANY DIRECTORY	EXECUTE ANY INDEXTYPE
CREATE ANY INDEX	EXECUTE ANY LIBRARY
CREATE ANY INDEXTYPE	EXECUTE ANY OPERATOR
CREATE ANY LIBRARY	EXECUTE ANY PROCEDURE
CREATE ANY OPERATOR	EXECUTE ANY TYPE
CREATE ANY OUTLINE	EXEMPT ACCESS POLICY
CREATE ANY PROCEDURE	FORCE ANY TRANSACTION
CREATE ANY SECURITY PROFILE	FORCE TRANSACTION
CREATE ANY SEQUENCE	GLOBAL QUERY REWRITE
CREATE ANY SNAPSHOT	GRANT ANY PRIVILEGE
CREATE ANY SYNONYM	GRANT ANY ROLE
CREATE ANY TABLE	INSERT ANY TABLE
CREATE ANY TRIGGER	LOCK ANY TABLE
CREATE ANY TYPE	MANAGE ANY QUEUE
CREATE ANY VIEW	MANAGE TABLESPACE
CREATE CLUSTER	ON COMMIT REFRESH
CREATE DATABASE LINK	QUERY REWRITE
CREATE DIMENSION	READUP
CREATE INDEXTYPE	READUP DBHIGH
CREATE LIBRARY	RESTRICTED SESSION
CREATE OPERATOR	RESUMABLE
CREATE PROCEDURE	SELECT ANY DICTIONARY
CREATE PROFILE	SELECT ANY SEQUENCE
CREATE PUBLIC DATABASE LINK	SELECT ANY TABLE
CREATE PUBLIC SYNONYM	SYSDBA
CREATE ROLE	SYSOPER
CREATE ROLLBACK SEGMENT	UNDER ANY TABLE
CREATE SECURITY PROFILE	UNDER ANY TYPE
CREATE SEQUENCE	UNDER ANY VIEW
CREATE SNAPSHOT	UNLIMITED TABLESPACE
CREATE SYNONYM	UPDATE ANY TABLE
CREATE TABLE	WRITEDOWN
CREATE TABLESPACE	WRITEDOWN DBLOW
CREATE TRIGGER	WRITEUP
	WRITEUP DBHIGH

Table 5. ORACLE SYSTEM PRIVILEGES

- *(DO03612: CAT II) The DBA will ensure that Oracle system privileges are not granted to PUBLIC.*

- *(DO0350: CAT II) The DBA will restrict the assignment of the Oracle system privileges listed above to DBAs, application installation accounts, application processing accounts, and predefined accounts in a production environment.*

- *(DO0350: CAT II) The DBA will restrict the assignment of the Oracle system privileges listed above to DBAs, application installation accounts, application processing accounts, predefined accounts, and application developer accounts in a development environment with the following exception:*

  - *The DBA will ensure that system privileges that allow administrative functions on objects other than those owned by the account (the "ANY" system privileges) are not granted to application developer accounts.*

B.5.3 Object Privileges

The alter, index, and references object privileges are the only object privileges that grant permissions to make system-wide modifications to database objects. The alter**,** index**,** and references object privileges will not be granted to any application user account, application administrator account or application role. No application object privileges will be granted to PUBLIC. Oracle recommends that unnecessary default privileges assigned to PUBLIC be revoked. However, no additional guidance has been provided by Oracle to assist in determining what object privilege assignments granted to PUBLIC are necessary. Until such time as more specific guidance is available, DBAs will not be required to revoke default object privileges from PUBLIC. However, it is recommended that all object privileges be revoked from PUBLIC and granted as necessary to custom application roles. Some object privilege grants expose specific vulnerabilities and will be revoked from PUBLIC. They are listed in a policy bullet below. Please note that revoking these privileges in Oracle database version 10.1 and later will disable the function of some default database applications. If site operations require use of these applications, then execute privileges on these packages must be granted directly to the Oracle default accounts requiring them and the invalidated Oracle packages must be recompiled. These accounts may include SYSMAN, MDSYS, WKSYS, and SYSTEM as well as others.

- *(DO3473: CAT III) The DBA will restrict assignment of the alter, index, and references object privileges to DBAs, object owners, and predefined roles unless justified and documented with the IAO.*

- *(DO3473: CAT III) The DBA will ensure that alter, index, and reference privileges are not granted to application developer accounts in a production database.*

- *(DO0320: CAT II) The DBA will ensure that application object privileges are not granted to PUBLIC.*

- *(DO3475: CAT II) The DBA will revoke the following object privileges assigned to PUBLIC during installation:*

  - *Execute on UTL_SMTP*
  - *Execute on UTL_TCP*
  - *Execute on UTL_HTTP*
  - *Execute on UTL_FILE*
  - *Execute on DBMS_RANDOM*
  - *Execute on DBMS_LOB*
  - *Execute on DBMS_ SQL*
  - *Execute on DBMS_JOB*
  - *Execute on DBMS_BACKUP_RESTORE*
  - *Execute on DBMS_OBFUSCATION_TOOLKIT*

B.5.4 Administration of Privileges

Application user database accounts, application administrator accounts, application developer accounts, or application roles will not have the administration option of any system privilege. Application user database accounts, application administrator accounts, application developer accounts, or application roles will not have the grant option of any object privilege. Application user database accounts, application administrator accounts, application developer accounts, or application roles will not have the administration option of any Oracle predefined role. The administration option of any role will be granted only to DBAs and application administrator accounts.

- *(DO3609: CAT II) The DBA will restrict the assignment of the administration option of any system privilege to DBAs.*

- *(DO3451: CAT II) The DBA will restrict the assignment of the grant option of any object privilege to DBAs and application object owner accounts.*

- *(DO3622: CAT II) The DBA will restrict assignment of the administration option of any Oracle predefined role with the exception of the AQ_ADMINISTRATOR_ROLE to DBAs. The administration option cannot be removed from this role.*

- *(DO3622: CAT II) The DBA will restrict assignment of the administration option of any application role to DBAs and application administrator accounts.*

B.6 Oracle Replication

Oracle replication configuration requires that a specific database account (usually named REPADMIN) be created and granted specific system privileges. This account is used to manage and administrate the replication functions. Other roles of the replication account include Propagator and Receiver. A single replication administration account will be used to perform all roles unless justified and documented with the IAO or IAM. The REPADMIN account may be directly granted the system privileges listed below on the master site database. A replication administration account must be created for each database. The privileges required by the REPADMIN to configure and administrator the entire replication environment account are automatically assigned via the DBMS_REPCAT_ADMIN.GRANT_ADMIN_ANY_SCHEMA. The REPADMIN account will not be granted system privileges other than those listed. The replication administration account must be protected from unauthorized access. Compromise of the REPADMIN account could lead to unauthorized propagations of database objects and content to unauthorized locations. Access to the REPADMIN account will be restricted to authorized DBAs. All passwords for the above accounts will follow the password requirements that are presented in *Section 3.2.1, Password Guidelines*.

REPADMIN SYSTEM PRIVILEGES	
ALTER ANY CLUSTER	CREATE DATABASE LINK
ALTER ANY INDEX	CREATE PUBLIC SYNONYM
ALTER ANY PROCEDURE	CREATE SESSION
ALTER ANY SEQUENCE	DELETE ANY TABLE
ALTER ANY SNAPSHOT	DROP ANY CLUSTER
ALTER ANY TABLE	DROP ANY INDEX
ALTER ANY TRIGGER	DROP ANY PROCEDURE
ALTER SESSION	DROP ANY SEQUENCE
COMMENT ANY TABLE	DROP ANY SNAPSHOT
CREATE ANY CLUSTER	DROP ANY SYNONYM
CREATE ANY INDEX	DROP ANY TABLE
CREATE ANY PROCEDURE	DROP ANY TRIGGER
CREATE ANY SEQUENCE	DROP ANY VIEW
CREATE ANY SNAPSHOT	DROP PUBLIC SYNONYM
CREATE ANY SYNONYM	INSERT ANY TABLE
CREATE ANY TABLE	SELECT ANY TABLE
CREATE ANY TRIGGER	UNLIMITED TABLESPACE
CREATE ANY VIEW	UPDATE ANY TABLE

Table 6. ORACLE REPADMIN SYSTEM PRIVILEGES

- *(DO0210: CAT III) The DBA will restrict access to the REPADMIN to authorized DBAs.*

- *(DO3485, DO3487, DO3504: CAT II) The DBA will configure the password for the REPADMIN account to adhere to DOD password policy as listed in Section 3.2.1, Password Guidelines.*

- *(DO0210: CAT III) The DBA will configure a single replication administration account to be used to perform all replication functions.*

B.7 Network Security

B.7.1 Encrypting Oracle Network Logins

Oracle password information in a connection request will be encrypted. This is automatically handled when transmitting over a network by Oracle version 9.02 and later. In versions earlier than 9.02, this is configured by setting the following parameters on each client and database server:

- Set the ORA_ENCRYPT_LOGIN environment variable to TRUE on the client machine.
- Set the DBLINK_ENCRYPT_LOGIN server initialization parameter to TRUE.

These parameters are set to FALSE by default and are not available for version 9.02 and later. If the ORA_ENCRYPT_LOGIN is not set to TRUE in the environment variables of the client, failed logon attempts will be retried and the password will be sent in clear text. Once these parameters have been set to TRUE, passwords will be encrypted in connection requests.

Please note that the setting or changing of passwords across the network is NOT encrypted without other encryption configuration such as Oracle Advanced Security.

- *(DO3698: CAT III) The DBA will ensure that Oracle passwords are encrypted when transmitted over a network connection.*

- *(DO3673: CAT III) The IAO will ensure that the Oracle environmental variable ORA_ENCRYPT_LOGIN is set to TRUE on Oracle client workstations.*

- *(DO3698: CAT III) The DBA will set the server initialization parameter DBLINK_ENCRYPT_LOGIN to TRUE.*

B.7.2 Protecting Database Network Communications

Where appropriate, integrity and confidentiality protections of network communication should be employed. As mentioned elsewhere in this document, administrative connections that may contain confidential information such as account password changes should be protected by encryption. In some database applications, the data itself including financial data such as credit card information, personnel data such as social security number and birth date, or other sensitive or classified information may require increased protection when being transmitted across the network. The sensitivity of the data should be reviewed and network communication protection configured if warranted. If local encryption is not implemented such as that provided by IPSec, then Oracle Advanced Security (OAS) should be used and configured to use Secure Socket Layer (SSL) if PKI is available or OAS native integrity/encryption if it is not. OAS should be configured in accordance with the Center for Internet Security (CIS) *Benchmark for Oracle Security* and/or the National Security Agency (NSA) Guide to the *Secure Configuration and Administration of Oracle9i Database Server*.

B.7.3 Oracle Listener Security

B.7.3.1 Listener Password

On UNIX, OS/390 systems supporting database connections over TCP/IP, and Windows hosts, Oracle establishes a listening process that accepts network requests to connect to the database. This process, the Oracle Listener, listens on the protocols specified in the listener.ora file. The listener itself may be accessed directly for administrative purposes using the LSNRCTL utility. Administrative access to the listener is controlled by two settings—the Security setting (the PASSWORDS_listener_name parameter in the listener.ora file), which indicates a password must be provided in order to perform administrative functions; and the ADMIN_RESTRICTIONS_*listener_name* listener.ora parameter that enables/disables the ability to modify the listener.ora file from the LSNRCTL utility while the listener is running.

No password is set on the listener by default. A listener password will be set after listener configuration. Failing to set a password on the listener could result in unauthorized users starting, stopping, and configuring the listener service. The password will be stored in encrypted format within the listener.ora file. This is accomplished by using the change_password function of the LSNRCTL utility.

- *(DO3630: CAT I) The DBA will set a password for the Oracle listener immediately after installation.*

- *(DO3630: CAT I) The DBA will ensure that the Oracle listener password is stored in encrypted format in the LISTENER.ORA file.*

B.7.3.2 Listener Administration Restrictions

The Oracle listener by default allows dynamic configuration via the LSNRCTL utility. Dynamic configuration leaves the listener vulnerable to unauthorized modification should the listener not be protected by a password or should the password be compromised. Dynamic configuration will be disabled by specifying the parameter ADMIN_RESTRICTIONS = ON in the listener.ora file for all listeners. This will require that any configuration changes be made to the listener through direct edits to the listener.ora file.

- *(DO6740: CAT II) The DBA will enable the ADMIN_RESTRICTIONS_listener_name parameter in the listener.ora file.*

B.7.3.3 Listener Access to External Procedures

Oracle provides access to executables run by the host OS by means of the Oracle EXTPROC component. The EXTPROC component has a known vulnerability that allows unauthenticated access via the Oracle Listener. If not required, the EXTPROC component will be disabled. This may be done through removal of the executable from the host system or by configuration of the listener. If required, a separate, dedicated listener will be created for exclusive use of the EXTPROC and network address restrictions to it will be strictly enforced. Use of the EXTPROC component will be justified and documented with the IAO.

- *(DO0280: CAT II) The DBA will disable the Oracle EXTPROC module if it is not required.*

- *(DO0280: CAT II) The DBA will configure a dedicated listener with appropriate address restrictions for the EXTPROC module if that component is required.*

- *(DO0280: CAT II) The DBA will configure TCP/IP address restrictions on systems that require use of the Oracle EXTPROC module.*

- *(DO0280: CAT II) The DBA will justify and document any use of external procedures with the IAO.*

B.7.3.4 Listener Network Address Restrictions

In a Windows and UNIX environment, access to the database from the network can be restricted based on TCP/IP network address. This restriction is defined in the SQLNET.ORA (PROTOCOL.ORA file for Oracle 8i) file. The parameter tcp.validnode checking=YES enables address restrictions. The parameter tcp.invited_nodes defines TCP/IP addresses that are allowed to connect and tcp.excluded_nodes defines TCP/IP addresses that are refused connections to the database. TCP/IP address restrictions will be defined on systems unless such restrictions are not feasible. A listener without defined address restrictions will be justified and documented with the IAO.

- *(DO0284: CAT III) The DBA will configure Oracle listeners to restrict access by network address unless justified and documented with the IAO.*

B.7.3.5 Encryption of Remote Administrative Access

Network connections made to databases using privileged database accounts including DBAs and other accounts with system privileges will be encrypted. The remote privileged users should configure a separate, dedicated Oracle listener to restrict and provide remote access to administrators. Refer to the NSA *Guide to the Secure Configuration and Administration of Oracle 9i Database Server* for more information on configuring encrypted network access to the database. An alternative for UNIX systems is to access the database server via SSH and access the database via a local session on the host.

- *(DG0093: CAT II) The DBA will ensure that Oracle database administrative connections across the network are encrypted.*

B.7.3.6 Listener Port Assignment

DOD requires standard port usage to better support firewall and intrusion detection monitoring. Therefore, Oracle default ports will be used to support Oracle network communications when traversing network firewalls.

By default, Oracle random ports may be assigned to individual network connections when shared server or multi-threaded server is used on UNIX platforms for Oracle versions earlier than 9i. Oracle always assigns random ports unless specifically disabled on both 8i and 9i versions of Oracle for Windows. Random port assignment will be disabled unless justified and documented with the IAO. This may be accomplished in the UNIX environment by either specifying DISPATCHERS=’’ in the init.ora file or by listing specific ports in the DISPATCHERS parameter in the init.ora file. In the Windows environment, random port assignment is disabled by adding the value HTLM\Software\Oracle\Home<ID#>\use_shared_socket=TRUE where ID# is the ID number assigned to a specific Oracle Home on the Oracle host system.

- *DO0285: CAT II) The DBA will ensure that random port assignment to network connections is disabled when traversing network firewalls.*

B.7.3.7 Listener Inbound Connection Timeout

The listener.ora parameter INBOUND_CONNECTION_TIMEOUT_*listener* in Oracle version 9i and later and CONNECT_TIMEOUT_*listener* in Oracle 8i and earlier, controls the amount of time the listener waits for a network client to complete the connection request. This limit protects the listener from consuming and holding resources for client connection requests that do not complete. A malicious user could use this to flood the listener with requests that result in a denial of service to authorized users. A connection timeout limit with the minimum appropriate for the application will be specified in the listener.ora and the database server sqlnet.ora (9i only) files. The expire_time sqlnet.ora parameter probes for dead connections and terminates them when found. This setting does cause a slight increase in network traffic. The sqlnet.ora expire_time will be set to greater than 0 unless justified and documented with the IAO.

- *(DO0286: CAT II) The DBA will configure the INBOUND_CONNECT_TIMEOUT_listener or CONNECT_TIMEOUT_listener parameter to be greater than 0 in the listener.ora and database server sqlnet.ora files.*

- *(DO0287: CAT II) The DBA will configure the sqlnet.expire_time parameter to be greater than 0 in the database server sqlnet.ora file.*

B.7.4 Oracle XML DB Protocol Server

The Oracle XML DB Protocol Server offers access to the Oracle XML DB resources using the standard Internet protocols FTP, HTTP, and WebDAV. This allows direct access to Oracle XML resources without the need for special or additional software. The Oracle XML DB Protocol Server is a specific type of Oracle shared server dispatcher and is specified in the Oracle database initialization parameter file for startup. Other Oracle XML DB Protocol Server configuration parameters are specified in the XML schema based XML resource named xdbconfig.xml. If access to the XML DB Protocol Server via the Internet protocols is not required, then they will be disabled. If access via the Internet protocols is required, logging will be enabled by setting the log-level for all enabled protocols to log a minimum of unsuccessful logins.

- *(DO0420: CAT II) The DBA will disable the Oracle XML DB Protocol Server if it is not required.*

- *(DO0421: CAT II) The DBA will enable logging for all protocols enabled on the XML DB Protocol Server.*

B.8 Oracle Intelligent Agent/Oracle Enterprise Manager (OEM)

The Oracle Intelligent Agent is used by the Oracle Enterprise Manager (OEM) to provide centralized database management both locally and remotely. Remote administration of databases is not prohibited, however, the enabling of remote administrative connections to the database introduces vulnerabilities to local databases and the host system itself. Additional protections are required for remote administrative action and some prohibitions do apply Administrative connections across the network are required to be encrypted (See B.7.3.5) in order to protect sensitive information such as passwords from being disclosed. The Oracle Intelligent Agent, because it offers administrative action on the local database and is available via the network, is vulnerable to attack. The Oracle Intelligent Agent will be disabled unless its use is justified and documented with the IAO. Network communications to the Oracle Intelligent Agent may include sensitive data and will be encrypted. Due to the increased vulnerability of databases and systems available to Internet users, the Oracle Intelligent Agent will not be installed or enabled on Oracle database hosts accessible to the Internet. Remote administration may still be performed using allowed protected local sessions such as VPN or protected dial-up connections to a local host account that has DBA privileges to t he database.

- *(DO0430: CAT II) The DBA will disable the Oracle Intelligent Agent on databases accessible to the Internet.*

- *(DO0430: CAT II) The DBA will ensure that the Oracle Intelligent Agent is disabled on databases accessible to the Internet unless justified and documented with the IAO.*

- *(DG0093: CAT II) Network communications to the Oracle Intelligent Agent will be encrypted.*

B.9 Oracle Account Protections

In addition to the password management controls specified above, Oracle accounts will be configured in accordance with the following security protections.

B.9.1 Default/Temporary Tablespaces and Tablespace Quotas

System tablespace will be restricted to use by the Oracle database system operation and maintenance. Oracle database accounts, with the exception of Oracle default accounts, will not specify the system tablespace as their default or temporary tablespace. In Oracle versions 9i and later, a default temporary tablespace will be defined and used. Such a default temporary tablespace ensures that objects identified as temporary objects are automatically removed from the database upon database session termination. Application user and application administration database accounts will have all tablespace quotas set to 0.

- *(DO0155: CAT III) The Oracle DBA will ensure that non-default Oracle database accounts do not specify the SYSTEM tablespace as the default or temporary tablespace.*

- *(DO0155: CAT III) The Oracle DBA will configure a default temporary tablespace in Oracle databases version 9i and later.*

- *(DO0157: CAT III) The DBA will ensure that application user and application administrator accounts have all quotas on all tablespaces set to 0.*

B.9.2 Idle Time in Oracle

In Oracle, the session inactivity time will be set through profiles by specifically setting the idle time within the profile (refer to *Section 5.2.5, Database Session Inactivity Time Out,* for time out requirements). This means that every Oracle database account will be assigned to a profile and the idle time setting within each profile will be set to 15 minutes or less.

Oracle automatically creates the default profile and assigns all Oracle database accounts to this profile, unless they are manually assigned to another profile at creation. The default profile will be modified so the idle time setting is set to 15 minutes or less. If any other profiles may beare created, they, too, will be set to 15 minutes or less unless required for operation of specific functions. Specific functions requiring an extended idle time will be justified and documented with the IAO. The ALTER PROFILE SQL statement will be used to change the idle_time resource under the default profile.

The initialization parameter, RESOURCE_LIMIT, will be set to TRUE, and the database restarted, before the resource values for any profiles will affect any and all database accounts.

- *(DO3536: CAT III) The DBA will configure an idle time limit for all database accounts through the use of profiles.*

- *(DO3536: CAT III) The DBA will configure all idle time limits to conform to established DOD or best practice/vendor-recommended policy (currently 15 minutes for general use database accounts).*

- *(DO3536: CAT III) The DBA will configure the Oracle DEFAULT profile for the required idle time.*

- *(DO3536: CAT III) The DBA will ensure that all database accounts requiring an idle time limit greater than 60 minutes are justified and documented with the IAO.*

- *(DO3536: CAT III) The DBA will ensure that all database accounts such as N-Tier connection accounts, connection pooling accounts, and non-interactive or batch processing accounts that require an unlimited idle time are justified and documented with the IAO.*

B.9.3 SESSIONS_PER_USER in Oracle

Current policy does not require a DBA to limit the number of sessions that a database account has open at any time. The SESSIONS_PER_USER setting can be a valuable way to control the number of connections that any particular database account has open at any one time. This parameter should be set in the profile of all application user database accounts and modified in the default profile.

Setting this parameter too low for a database application that requires multiple sessions can prevent it from functioning properly. In the case of an application that allows multiple users to connect with a single database account, too low a value for SESSIONS_PER_USER could prevent subsequent application users from connecting.

B.10 Oracle ARCHIVELOG Mode

The Oracle ARCHIVELOG mode allows databases to be recovered after failure to a specific point in time by archiving redo log files. If ARCHIVELOG mode is not enabled, then recovery of Oracle databases may only be recovered to the time when the database was last backed up. ARCHIVELOG mode is appropriate for dynamic or transaction oriented database systems, but not necessarily so for databases that store relatively static data. The DBA should enable ARCHVELOG mode based on the appropriateness for the database.

B.11 Securing SQLPlus Commands

The Oracle SQLPlus application provides a means to enter SQL statements directly to the Oracle database. Accounts with access to this application may connect to the database and exercise their full privileges as granted to their database account. Oracle offers a way to limit access to SQL commands entered in the SQLPlus application by establishing the Product User Profile. The SYSTEM account will create the PRODUCT_USER_PROFILE table by running the pupbld.sq**l** script. At a minimum, the SQLPlus HOST command that allows access to the database host system commands will be restricted to authorized DBAs.

- *(DO0410: CAT II) The DBA will use the SYSTEM database account to create the PRODUCT_USER_PROFILE table by running the pupbld.sql script.*

- *(DO0410: CAT II) The DBA will restrict access to the SQL*Plus HOST command to authorized DBAs.*

B.12 Protection of Database Stored Procedures

Oracle provides the WRAP Utility to encrypt stored PL/SQL procedures and functions. However, the utility does not encrypt string or numerical literals, variable names, or table and column names. It does not encrypt passwords stored in these procedures or functions. The WRAP utility will be used to encrypt custom and GOTS application code stored in the database. Variable values may be obscured by using concatenated values. Account names and passwords and other sensitive data should not be hard coded in the PL/SQL code. Instead, store usernames and passwords in encrypted form within a protected database table.

- *(DG0091: CAT II) The DBA will ensure that custom and GOTS application code objects of type FUNCTION, PROCEDURE, and PACKAGE BODY stored in the database are encrypted using the Oracle WRAP Utility.*

B.13 Oracle Trace Utility

The Oracle Trace Utility, otrace, is used to collect performance and resource utilization data. Such data collection can have a negative impact on database performance and disk space usage. If not in use, the Oracle Trace Utility should be disabled. This can be accomplished by the deletion of the process.dat, collect.dat, and regid.dat files found in the $ORACLE_HOME/otrace/admin directory. Backups of these files should be confirmed before any deletion is done in order to provide the trace capability in future if required.

B.14 Auditing in Oracle

This section describes how to enable auditing on an individual database. The minimum level of auditing will be enabled for every database on all machines.

B.14.1 Oracle Audit Monitoring

The following Oracle audit and error log files will be reviewed in accordance with the audit monitoring requirements listed in *Section 4.4, Audit Data Review*. In addition, it is recommended that the information found in the v_$resource_limit table be monitored regularly to prevent exhaustion of configured resources.

1. Alert log file (location specified in BACKGROUND_DUMP_DEST initialization parameter).

2. Listener log file (location in $ORACLE_HOME/network/log directory)

B.14.2 Database Auditing

Oracle provides the capability to capture audit information and to store this audit information within an Oracle database table or external datafiles. When stored in the database, all audit information is stored in a database table named AUD$. This table is owned by the SYS account by default. Privileges to delete, update, or insert directly into the audit table will be restricted to auditors and DBAs. Access to this table will be restricted to DBAs or security auditors. The DBA or security auditor may delete information from this table after performing maintenance backups. When stored externally to datafiles, the datafiles will be protected by the operating system. Access to the external audit files will be granted in accordance with the OS STIG.

- *(DO3446: CAT II) The DBA will restrict access to the AUD$ table to DBAs and/or security auditors.*

- *(DO0234: CAT II) The DBA/SA will restrict access to the external audit files to security auditors, SAs, and DBAs.*

B.14.3 AUD$ Table Location and Ownership

By default, the AUD$ table resides inside the SYSTEM tablespace and is owned by SYS. Although not supported by Oracle, moving the AUD$ table to a dedicated tablespace and onto a separate disk can improve Oracle performance. If you decide to move the AUD$ audit data table, please do so as prescribed in Oracle Document ID 72460.1. It may be necessary to move the audit table back to the SYSTEM tablespace to support upgrades or backup and recovery operations. Ownership of the AUD$ table will be restricted to a protected database account such as SYS or SYSTEM. Access to the protected account will have the same restrictions as the SYS and SYSTEM accounts.

- *(DO0190: CAT II) The DBA will configure the AUD$ table to be owned by a protected database account.*

B.14.4 Enabling Auditing

To enable Oracle auditing, the initialization parameter AUDIT_TRAIL will be set to TRUE, DB, or OS. Setting the initialization parameter to TRUE or DB will store audit records in the AUD$ table within the database. Setting the initialization parameter to OS will store records in an operating system audit file. The directory name for this OS file can be specified in the AUDIT_FILE_DEST initialization parameter on host systems other than Windows. On Windows hosts, the audit trail data directed to the OS is stored in the Windows event logs.

There are events that are audited by default regardless of the setting of the AUDIT_TRAIL parameter. Oracle will always audit certain database-related actions regardless of whether database auditing is enabled. These events include the following:

- Instance startup
- Instance shutdown
- Connections to the database with administrator privileges (SYS, SYSDBA, SYSOPER)

- *(DO3413: CAT II) The DBA will set the Oracle audit trail parameter to AUDIT_TRAIL=TRUE or AUDIT_TRAIL=DB or AUDIT_TRAIL=OS.*

B.14.5 AUDIT/NOAUDIT Statements

To activate a particular audit option, a DBA uses the AUDIT SQL statement. To disable auditing, a DBA uses the NOAUDIT SQL statement. The forms of the AUDIT SQL statement are shown below.

- AUDIT system-privilege
- AUDIT system-privilege WHENEVER SUCCESSFUL
- AUDIT system-privilege WHENEVER NOT SUCCESSFUL

By default, an audit privilege statement audits both successful and unsuccessful attempts to use system privileges. If only successful or unsuccessful attempts are desired, then the preference can be specified by using the WHENEVER SUCCESSFUL and WHENEVER NOT SUCCESSFUL indicators.

The NOAUDIT SQL statement is used to disable auditing for the specific option desired. A DBA will perform the NOAUDIT statement to disable auditing.

- NOAUDIT system-privilege
- NOAUDIT system-privilege WHENEVER SUCCESSFUL
- NOAUDIT system-privilege WHENEVER NOT SUCCESSFUL

Conversely, the NOAUDIT SQL command disables all auditing on use of a specific system privilege when neither WHENEVER SUCCESSFUL or NOT SUCCESSFUL is not specified. If NOAUDIT is specified with WHENEVER SUCESSFUL, then only successful use of the specified system privilege is not audited. If the system privilege had previously been audited for both successful and unsuccessful accesses, then auditing would still be in effect for unsuccessful uses. Similarly, the NOAUDIT WHENEVER NOT SUCCESSFUL disables auditing on a system privilege only for unsuccessful uses.

The following statements show the usage for the AUDIT and NOAUDIT SQL statements for object privileges:

- AUDIT object-privilege ON object-name [WHENEVER [NOT] SUCCESSFUL]
- NOAUDIT object-privilege ON object-name [WHENEVER [NOT] SUCCESSFUL]

B.14.6 Mandatory Auditing

Once auditing is enabled at the database level, the following system privilege and object auditing will be enabled. All auditing will be done BY ACCESS unless justified and documented with the IAO. Access auditing means that audit records are generated each time an audited privilege is used or an audited statement is issued. Auditing by access may generate more audit records than audit by session; however, it does not impact database performance as much since each audit event does not trigger a search through the audit trail for the same action as does audit by session. The mandatory auditing requirements are meant to capture only changes to the data dictionary or database structure or privilege assignment events. Please note that in production systems few actions that generate audit records should occur. Database objects and structure are typically static. Data access and modification auditing should be decided during application design and implemented as an application requirement.

- *(DO3610, DO3692: CAT II) The DBA will configure all auditing to be recorded BY ACCESS unless justified and documented with the IAO.*

B.14.6.1 Statement Auditing

Statement auditing is a basic group of statements that, if performed, will generate an audit record. The statement auditing options specified in *Table 2* will be specified as a minimum. These audit options may be enabled by issuing the following SQL audit statements:

```
AUDIT ALL;
AUDIT ALL PRIVILEGES;
AUDIT SYSDBA;
AUDIT SYSOPER;
AUDIT ALTER SEQUENCE;
AUDIT ALTER TABLE;
AUDIT COMMENT TABLE;
AUDIT GRANT DIRECTORY;
AUDIT GRANT PROCEDURE;
AUDIT GRANT SEQUENCE;
AUDIT GRANT TABLE;
AUDIT GRANT TYPE;
```

The following SQL statements will disable audits set by the commands above that are not required:

```
NOAUDIT EXECUTE ANY LIBRARY;
NOAUDIT EXECUTE ANY PROCEDURE;
NOAUDIT EXECUTE ANY TYPE;
NOAUDIT EXECUTE LIBRARY;
NOAUDIT LOCK ANY TABLE;
NOAUDIT SELECT ANY SEQUENCE;
NOAUDIT SELECT ANY TABLE;
NOAUDIT UPDATE ANY TABLE;
NOAUDIT DELETE ANY TABLE;
NOAUDIT EXECUTE ANY INDEXTYPE;
NOAUDIT EXECUTE ANY OPERATOR;
NOAUDIT INSERT ANY TABLE;
NOAUDIT NETWORK;
NOAUDIT DELETE TABLE;
NOAUDIT INSERT TABLE;
NOAUDIT UPDATE TABLE;
NOAUDIT EXECUTE PROCEDURE;
NOAUDIT SELECT TABLE;
NOAUDIT SELECT SEQUENCE;
```

ORACLE STATEMENT AUDIT REQUIREMENTS	
ALTER ANY CLUSTER	DIMENSION
ALTER ANY DIMENSION	DIRECTORY
ALTER ANY INDEX	DROP ANY CLUSTER
ALTER ANY LIBRARY	DROP ANY DIMENSION
ALTER ANY OUTLINE	DROP ANY DIRECTORY
ALTER ANY PROCEDURE	DROP ANY INDEX
ALTER ANY ROLE	DROP ANY LIBRARY
ALTER ANY SEQUENCE*	DROP ANY OUTLINE
ALTER ANY SNAPSHOT*	DROP ANY PROCEDURE
ALTER ANY TABLE	DROP ANY ROLE
ALTER ANY TRIGGER	DROP ANY SEQUENCE
ALTER ANY TYPE	DROP ANY SNAPSHOT
ALTER DATABASE	DROP ANY SYNONYM
ALTER PROFILE	DROP ANY TABLE
ALTER RESOURCE COST	DROP ANY TRIGGER
ALTER ROLLBACK SEGMENT	DROP ANY TYPE
ALTER SEQUENCE	DROP ANY VIEW
ALTER SESSION	DROP PROFILE
ALTER SYSTEM	DROP PUBLIC DATABASE LINK
ALTER TABLE	DROP PUBLIC SYNONYM
ALTER TABLESPACE	DROP ROLLBACK SEGMENT
ALTER USER	DROP TABLESPACE
ANALYZE ANY	DROP USER
AUDIT ANY	ENQUEUE ANY QUEUE
BACKUP ANY TABLE	FORCE ANY TRANSACTION
BECOME USER	FORCE TRANSACTION
CLUSTER	GLOBAL QUERY REWRITE
COMMENT ANY TABLE	GRANT ANY PRIVILEGE
COMMENT TABLE	GRANT ANY ROLE
CONTEXT	GRANT DIRECTORY
CREATE ANY CLUSTER	GRANT PROCEDURE
CREATE ANY DIMENSION	GRANT SEQUENCE
CREATE ANY DIRECTORY	GRANT TABLE
CREATE ANY INDEX	GRANT TYPE
CREATE ANY LIBRARY	INDEX
CREATE ANY OUTLINE	MANAGE ANY QUEUE
CREATE ANY PROCEDURE	MANAGE TABLESPACE
CREATE ANY SEQUENCE	NOT EXISTS

ORACLE STATEMENT AUDIT REQUIREMENTS	
CREATE ANY SNAPSHOT	PROCEDURE
CREATE ANY SYNONYM	PROFILE
CREATE ANY TABLE	PUBLIC DATABASE LINK
CREATE ANY TRIGGER	PUBLIC SYNONYM
CREATE ANY TYPE	QUERY REWRITE
CREATE ANY VIEW	RESTRICTED SESSION
CREATE CLUSTER	ROLE
CREATE DATABASE LINK	ROLLBACK SEGMENT
CREATE DIMENSION	SEQUENCE
CREATE LIBRARY	SYNONYM
CREATE PROCEDURE	SYSDBA
CREATE PROFILE	SYSOPER
CREATE PUBLIC DATABASE LINK	SYSTEM AUDIT
CREATE PUBLIC SYNONYM	SYSTEM GRANT
CREATE ROLE	TABLE
CREATE ROLLBACK SEGMENT	TABLESPACE
CREATE SEQUENCE	TRIGGER
CREATE SESSION	TYPE
CREATE SNAPSHOT	UNLIMITED TABLESPACE
CREATE SYNONYM	USER
CREATE TABLE	VIEW
CREATE TABLESPACE	
CREATE TRIGGER	
CREATE TYPE	
CREATE USER	
CREATE VIEW	
DATABASE LINK	
DEQUEUE ANY QUEUE	

Table 7. ORACLE STATEMENT AUDITING REQUIREMENTS

*This is a system privilege audit option in Oracle 8i.

- *(DO3692: CAT II) The DBA will enable the statement auditing options presented above.*

B.14.6.2 Object Auditing

Object auditing audits events related to a specific object. The following auditing option applies to specific objects and will be enabled by a DBA or security administrator. Some applications may require additional object auditing option as specified by the application.

In addition to auditing for specific events, Oracle provides the capability of setting a default audit option for all objects. All objects created after a default has been defined, will be audited for the default audit event. All application objects will be audited for RENAME. The RENAME audit option will be set as the default for all objects. The RENAME audit option will be set for all existing application objects.

- *(DO3610: CAT II) The DBA will enable the object auditing option RENAME for all application objects.*

- *(DO3610: CAT II) The DBA will enable the object auditing option RENAME by default for objects.*

B.14.6.3 System Privilege Auditing

Oracle provides the capability to audit individual use of Oracle system privileges within the database. The list of available system privileges is 127 for Oracle 8i and 141 for Oracle 9i. These privileges grant the permission to issue data definition language (DDL) statements that modify the data dictionary and effect system-wide changes. The system privileges listed below in *Table 3* below will be audited.

SYSTEM PRIVILEGE AUDIT REQUIREMENTS
ADMINISTER DATABASE TRIGGER
ADMINISTER RESOURCE MANAGER
ALTER ANY CLUSTER
ALTER ANY DIMENSION
ALTER ANY INDEX
ALTER ANY INDEXTYPE
ALTER ANY LIBRARY
ALTER ANY OUTLINE
ALTER ANY PROCEDURE
ALTER ANY ROLE
ALTER ANY SEQUENCE
ALTER ANY SNAPSHOT
ALTER ANY TABLE
ALTER ANY TRIGGER
ALTER ANY TYPE
ALTER DATABASE
ALTER OPERATOR
ALTER PROFILE
ALTER RESOURCE COST
ALTER ROLLBACK SEGMENT

SYSTEM PRIVILEGE AUDIT REQUIREMENTS
ALTER SESSION
ALTER SYSTEM
ALTER TABLESPACE
ALTER USER
ANALYZE ANY
AUDIT ANY
AUDIT SYSTEM
BACKUP ANY TABLE
BECOME USER
COMMENT ANY TABLE
CREATE ANY CLUSTER
CREATE ANY CONTEXT
CREATE ANY DIMENSION
CREATE ANY DIRECTORY
CREATE ANY INDEX
CREATE ANY INDEXTYPE
CREATE ANY LIBRARY
CREATE ANY OPERATOR
CREATE ANY OUTLINE
CREATE ANY PROCEDURE
CREATE ANY SEQUENCE
CREATE ANY SNAPSHOT
CREATE ANY SYNONYM
CREATE ANY TABLE
CREATE ANY TRIGGER
CREATE ANY TYPE
CREATE ANY VIEW
CREATE CLUSTER
CREATE DATABASE LINK
CREATE DIMENSION
CREATE INDEXTYPE
CREATE LIBRARY
CREATE OPERATOR
CREATE PROCEDURE
CREATE PROFILE
CREATE PUBLIC DATABASE LINK
CREATE PUBLIC SYNONYM
CREATE ROLE

SYSTEM PRIVILEGE AUDIT REQUIREMENTS
CREATE ROLLBACK SEGMENT
CREATE SEQUENCE
CREATE SESSION
CREATE SNAPSHOT
CREATE SYNONYM
CREATE TABLE
CREATE TABLESPACE
CREATE TRIGGER
CREATE TYPE
CREATE USER
CREATE VIEW
DEQUEUE ANY QUEUE
DROP ANY CLUSTER
DROP ANY CONTEXT
DROP ANY DIMENSION
DROP ANY DIRECTORY
DROP ANY INDEX
DROP ANY INDEXTYPE
DROP ANY LIBRARY
DROP ANY OPERATOR
DROP ANY OUTLINE
DROP ANY PROCEDURE
DROP ANY ROLE
DROP ANY SEQUENCE
DROP ANY SNAPSHOT
DROP ANY SYNONYM
DROP ANY TABLE
DROP ANY TRIGGER
DROP ANY TYPE
DROP ANY VIEW
DROP PROFILE
DROP PUBLIC DATABASE LINK
DROP PUBLIC SYNONYM
DROP ROLLBACK SEGMENT
DROP TABLESPACE
DROP USER
ENQUEUE ANY QUEUE
EXTENDS ANY TYPE*

SYSTEM PRIVILEGE AUDIT REQUIREMENTS
FORCE ANY TRANSACTION
FORCE TRANSACTION
GLOBAL QUERY REWRITE
GRANT ANY PRIVILEGE
GRANT ANY ROLE
MANAGE ANY QUEUE
MANAGE TABLESPACE
QUERY REWRITE
RESTRICTED SESSION
UNLIMITED TABLESPACE

Table 8. ORACLE REQUIRED SYSTEM PRIVILEGE AUDITS

* This is not an audit option in Oracle9i.

- *(DO3692: CAT II) The DBA will ensure that the system privileges listed in the above table are audited.*

B.14.7 Fine-Grained Auditing

Oracle fine-grained auditing, the ability to audit changes to data, may be implemented within an Oracle database in the form of SQL predicates defined on table-object access conditions. This allows audit records to be generated by the Oracle database instead of requiring the audit capability to be written into each application accessing the data.

B.14.8 Audit Trail Maintenance

The Oracle audit trail will be maintained periodically. The audit data will be exported and then purged routinely. The SYSTEM tablespace where the audit trail is located will be checked routinely to ensure that free space is available for the audit trail table to grow. If space is not available (the AUD$ table has consumed all of it), then all activity upon the DBMS will stop until space is made available by the DBA. The sizing of the audit trail tablespace and the maintenance of the audit trail will be specific to each application.

- *(DG0030: CAT III) The DBA will ensure that audit data is maintained for a minimum of one year.*

B.15 Oracle File and Directory Operating System Permissions

(This section and its policies do not apply to the OS/390 environment.)

B.15.1 Oracle File and Directory Ownership

All files stored in the $ORACLE_HOME/bin directory will be owned by the Oracle software installation account.

- *(DG0019: CAT III) The SA/DBA will ensure that all files stored in the $ORACLE_HOME/bin directory are owned by the Oracle software installation account.*

B.15.2 Oracle File and Directory Permissions

All permissions to files and directories that are created as the result of an installation of Oracle or stored in the $ORACLE_HOME directory will be set to the recommended security settings of the Oracle installation guide or more restrictive. Accounts other than the Oracle software owner account and the DBA group should be denied access except to executables under the $ORACLE_HOME/bin directory as specifically required. These files and directories will be secured by using access control methods native to the operating system.

- *(DO3613: CAT II) The SA/DBA will set all directories created by the installation of Oracle to Oracle's recommended security settings or more restrictive. See File Permissions under specific OS policy*.

- *(DO3613, DM3769: CAT II) The SA/DBA will ensure that access to executables or files under the $ORACLE_HOME/bin directory or subdirectories is restricted to the Oracle software owner account, DBAs, and particular accounts as specifically required.*

B.15.3 Initialization Parameter Files

Access to the Oracle initialization parameter files including INIT.ORA, INIT<SID>.ORA, and/or SPFILE.ORA will be restricted to the Oracle owner and DBAs.

- *(DO0276: CAT II) The SA/DBA will restrict access to the Oracle initialization parameter file to the Oracle owner account and DBAs.*

B.15.4 Remote Logon Password File Permissions

Oracle stores the internal SYS password and the password of accounts granted the SYSDBA or SYSOPER role in the **orapw<SID>** file. Although the passwords are encrypted, access to this file will be restricted to authorized DBAs. *Read* access to this file could allow someone to determine the internal or SYS password and would allow an unauthorized user access to Oracle.

- *(DO3845: CAT II) The SA/DBA will restrict access to the ORAPW<SID> file to the Oracle owner account and authorized DBAs.*

B.15.5 Listener.ora File Permissions

The permissions of the operating system file listener.ora, which houses listener configuration parameters and the listener password, will be restricted to the Oracle software owner or the DBA OS user group. *Read* access to this file could allow someone to determine the listener service password and would allow an unauthorized user to start, stop, and configure the listener service. If a password has been set, the following entry will be found in the file:

```
PASSWORDS_listener_name =
```

- *(DO3623: CAT I) The SA/DBA will restrict access to the listener.ora file to the Oracle owner account, the Oracle TNSLISTENER service/process account, and authorized DBAs.*

B.15.6 DBSNMP_RW.ORA and DBSNMP_RO.ORA File Permissions

The SNMP_RW.ORA file contains the password for the DBSNMP database account in cleartext. The SNMP_RO.ORA file contains configuration information for the Oracle Intelligent Agent. Access to these files will be restricted to the Oracle software owner account and DBAs.

The DBSNMP_RW.ORA and DBSNMP_RO.ORA files are used to configure the Oracle Intelligent Agent process. The DBSNMP_RO.ORA file contains configuration information for the Intelligent Agent. The DBSNMP_RW.ORA file includes the username and clear text password that the Intelligent Agent process uses to connect to the database. *Read* access to this file would provide information for unauthorized access to the Oracle Intelligent Agent database account (DBSNMP). If the default password has been changed, the following entries will be found in the file:

```
SNMP.CONNECT.<service_name>.NAME=
SNMP.CONNECT.<service_name>.PASSWORD=
```

- *(DO3642: CAT I) The SA/DBA will restrict access to the dbsnmp_rw.ora and dbsnmp_ro.ora files to the Oracle software owner account and authorized DBAs.*

B.15.7 SQLNET.ORA File Permissions

The SQLNET.ORA file contains network configuration information for the host database and listener. Unauthorized access to this file could result in compromised access to the database and/or the listener. Access to the database server SQLNET.ORA file will be restricted to the Oracle software owner account and DBAs.

- *(DG0090: CAT II) The SA/DBA will ensure that access to the SQLNET.ORA file is restricted to the Oracle software owner account and DBAs.*

B.15.8 Network Log and Trace File Protections

SQLNet and Listener log and trace files may contain information useful for accomplishing unauthorized database access. Access to these files will be restricted to the Oracle software owner account and DBAs.

The SQLNET.ORA parameters log_directory_client, log_directory_server, trace_directory_server, and trace_directory_client will be set to a valid, protected directories.

The LISTENER.ORA parameters log_file_listener and trace_directory_<listener name> will be set to valid, protected directories. The LISTENER.ORA parameter logging_listener will be set to the value ON. Access to the file designated in the LISTENER.ORA parameter trace_file_<listener name>_n will restricted to the Oracle software owner account and DBAs.

- *(DG0090: CAT II) The SA/DBA will ensure that access to the SQLNet and Listener log files is restricted to the Oracle software owner account and DBAs.*

B.16 Oracle Critical File Management

Access to the Oracle critical files listed below will be restricted to the Oracle owner account and DBAs.

- *(DG0090: CAT II) The SA/DBA will ensure that access to the Oracle critical files will be restricted to the Oracle owner account and DBAs.*

B.16.1 Control Files

All Oracle databases have control files defined that are used to record database dynamic information including the database name, the name and location of Oracle database files and online redo log files, the timestamp of the current database creation, current log sequence number, and checkpoint information. In addition to managing ongoing activity it is used for database recovery. Because the control file function is critical, a minimum of two control files will be defined for each database and located on separate disks. Placing the files on separate disks will prevent total loss in the event of a disk failure.

- *(DO0260: CAT IV) The DBA will configure a minimum of two Oracle control files for each Oracle database.*

- *(DO0260: CAT IV) The DBA will locate the Oracle control files on separate physical or RAID 1 or 5 disks.*

B.16.2 Redo Log Files

Online redo log files contain records of all changes made to the database as they occur. They are critical to the recovery of a failed database instance. In order to protect online redo log files from disk failures, redo log files will be multiplexed by defining a minimum of two redo log file groups configured with a minimum of two file members each. The file members for each redo log file group will be located on separate physical disks. For example, the first file member from each group will be placed on one physical disk and the second file from each group will be placed on a separate physical disk.

- *(DO0270: CAT IV) The DBA will configure a minimum of two Oracle redo log file groups on separate physical disks or RAID 5 or 1 disks, with a minimum of two members each, for each database.*

B.16.3 Database Files

Datafiles contain the actual database data. Placing datafiles on separate physical disks isolates disk contention between applications as well as allows for discrete OS file protections. Additionally, they should be placed on separate physical disks from redo log files to allow for recovery in the event of disk failure. However, when redo logs are multiplexed, this is not necessary.

B.17 Optimal Flexible Architecture (OFA)

File location and host system file structure is configured by default at Oracle installation to comply with Oracle’s Optimal Flexible Architecture (OFA). The OFA defines naming conventions and file structures for housing the three types of Oracle files—database files (datafiles, control files, and redo log files), product or software files (database executables), and administrative files (configuration files, export files, script files, etc.). The OFA structure provides a foundation for a more manageable, more efficient, better performing, and more reliable database system. It is recommended that all Oracle installations conform to the OFA.

Oracle does not provide OFA guidelines for Oracle installations on the OS/390 platform; however, instance naming standards, database file naming standards, and tablespace naming standards should still be followed.

B.17.1 Instance Naming Standards

The system identifier of a database using the Oracle DBMS is referred to as the Oracle SID. To allow for portability of the applications across multiple operating systems, Oracle recommends that the instance name be between four and eight characters long. If third-party or COTS software recommends a certain naming standard, follow their recommendation, however, when possible change the default SID of third party software from the default which may be well known.

Production database instance names or Oracle SIDs will not include a version number, Oracle-related or otherwise. Production database instance names or Oracle SIDs will not use the default of ORCL.

- *(DO0220: CAT IV) The DBA will not include a version number, Oracle-related or otherwise, in production database instance names or Oracle SIDs.*

- *(DO0220: CAT IV) The DBA will not use the default name of ORCL for production database instance names or Oracle SIDs.*

- *(DO0221: CAT III) The DBA will not use the default SID of third party applications unless it cannot be changed.*

B.17.2 Tablespaces

The use of dedicated tablespaces for different applications reduces the possibility of contention of disk storage. The USER tablespace or similar tablespace should be designated as the default tablespace for DBA's and developers. Application objects will be located in separate, dedicated tablespaces created for each application.

- *(DO0231: CAT IV) The DBA will locate the application segment(s) in separate, dedicated tablespace(s).*

B.17.3 Oracle UNIX Specific OFA Standards

B.17.3.1 Oracle UNIX Directory Structure Standards

Oracle files may be categorized into three types of files—administrative, application, and database files. These categories of files require specific storage and access requirements on the host system. Administrative files include database initialization and configuration files, log files, export files, and other types of database output files. Application files include database server executables and network administrative configuration files. Database files include the database control, redo log, and datafiles. On UNIX systems, Oracle recommends that a minimum of four mount points be defined. One mount point is for the administration and application software and three are for database files. Database file mount points are used to provide redundancy for control and redo log files as well as to separate datafiles based on I/O contention, backup requirements, and lifespan considerations. Mount point names should follow the format of */pm* where *p* is a fixed string constant and *m* is a unique identifier such as sequential numbering. No Oracle files should be located on the same partition as the operating system.

The ORACLE_BASE directory, defined by a UNIX account environment variable, is used as the parent directory for all Oracle product installations. The OFA-compliant ORACLE_HOME directory naming follows the format */mount-point/standard directory name*/oracle*/version* where the standard mount-point is as described above and the standard directory name is a UNIX standard directory name like "app." The ORACLE_HOME directory is used to store specific database version software. On OFA-compliant installations, it is a subdirectory under the ORACLE_BASE directory. The OFA-compliant ORACLE_HOME directory naming follows the format $ORACLE_BASE/product*/version* where product denotes the database server product and version denotes the Oracle Database version. Full pathnames should only be referenced in the files meant to store them such as the /etc/passwd and Oracle oratab files. Database-specific administrative files in an OFA-compliant system are stored under the $ORACLE_BASE/admin/*database name* directory.

OFA guidelines currently are not supported in the Defense Information Infrastructure Common Operating Environments (DII COE) guidelines for UNIX. The directories to support the OFA guidelines can be established as soft links. The system can meet both OFA and DII COE requirements by placing a symbolic link or file stub in the default location that points to the OFA file locations. Site-specific (application specific) data should be located on separate mount points.

B.17.3.2 Oracle UNIX Datafile Location Standards

All database files associated with a database instance should reside in a directory structure, similar to the examples provided below. Site-specific (application-specific) datafiles should be located on a separate mount point and these mount points should be located on separate disk drives.

Example:

Datafiles associated with the client tablespaces for the application ACME supporting the location FB:

/u01/oradata/acmefb/client01.dbf

Example:

Datafiles associated with the index for the client tablespace for the application ACME supporting the location FB:

/u01/oradata/acmefb/client01.idx

B.17.4 Windows Specific OFA Standards

OFA standards for Oracle on Windows and UNIX have the same main subdirectory structure and filenames. They differ in the root directory level names and in the method for defining variables. The ORACLE_BASE directory is located by default directly under the root of a disk partition and named X:\ORACLE. The ORACLE_HOME directory, which supports a specific version or release of Oracle, is located directly under the ORACLE_BASE directory, for example, ORACLE_BASE\ora90. Under the ORACLE_BASE directory are the ADMIN, ORACLE_HOME, and ORADATA directories. No Oracle files should be stored on the same partition as the Windows operating system.

On Windows, Oracle variables are defined in the registry rather than in environment variables. Also, Windows does not support symbolic links that allow UNIX systems to support an apparent single directory structure even though files may be on different physical disks.

B.17.5 Oracle OS/390 Specific Naming Standards

B.17.5.1 OSDI Subsystem Naming Standard

Under Oracle’s OSDI architecture, a single subsystem may support multiple instances. The OSDI subsystem must have a unique one to four-character name. The subsystem name will be used as the command prefix for OSDI commands. Oracle recommends that the subsystem name also be used as the OSDI command prefix.

B.17.5.2 OSDI Service Naming Standards

OSDI service names are used as the OS/390 jobname for the service unless otherwise specified. To ensure that the OSDI services are run under JES and not the master subsystem, OSDI service names will be unique from any OS/390 subsystem names. OSDI jobnames will use the OSDI service name. OSDI database service names will be used as the Oracle system identifiers (SID) and must follow requirements for Oracle instance naming. A single OSDI network service may support all connections to and from all from all database services on the host system. The network service name should indicate that it is a network and not a database service.

- *(DO0455: CAT III) The SA/DBA will configure the OSDI service names to be unique from any OS/390 subsystem names.*

- *(DO0456: CAT III) The SA/DBA will configure OSDI jobnames to use the OSDI service name.*

- *(DO0457: CAT III) The SA/DBA will use the OSDI database service names as the Oracle system identifiers (SID) and will follow requirements for Oracle instance naming.*

B.18 Initialization Parameters

This section covers Oracle Initialization parameters that have security impacts and the parameters that need to be set for security to operate. This section is not intended to cover all Oracle Initialization parameters. All initialization parameters listed below must be specified for all Oracle instances.

B.18.1 AUDIT_TRAIL

The AUDIT_TRAIL parameter specifies where the Oracle database writes the audit trail information. The valid values are TRUE, DB, and OS. This parameter may be placed anywhere in the parameter file after the db_name parameter.

- *(DO3413: CAT II) The DBA will set the AUDIT_TRAIL parameter audit_trail=TRUE, audit_trail=DB, or audit_trail=OS.*

B.18.2 RESOURCE_LIMIT

The RESOURCE_LIMIT parameter specifies whether or not enforcement of resource limits is enabled. If not enabled, the required idle time limits would not be enforced. The default value for this parameter is FALSE, the required value for this parameter is TRUE.

- *(DO3696: CAT II) The DBA will set the RESOURCE_LIMIT parameter to TRUE.*

B.18.3 REMOTE_OS_AUTHENT

The parameter REMOTE_OS_AUTHENT, when set to TRUE, allows the authentication of remote clients by the host operating system. The default value for this parameter is FALSE. This parameter will remain set to FALSE because of the risk of an impersonation attack (impersonating a valid OS node), otherwise known as spoofing. The required value for this parameter is FALSE.

- *(DO3538: CAT I) The DBA will set the REMOTE_OS_AUTHENT parameter to FALSE.*

B.18.4 REMOTE_OS_ROLES

The parameter REMOTE_OS_ROLES, when set to TRUE, allows operating system roles to be used from remote clients. The required value for this parameter is FALSE.

- *(DO3539: CAT I) The DBA will set the REMOTE_OS_ROLES parameter to FALSE.*

B.18.5 OS_ROLES

The parameter OS_ROLES, when set to TRUE, allows operating system roles to be used. The required value for this parameter is FALSE.

- *(DO0240: CAT II) The DBA will set the OS_ROLES parameter to FALSE.*

B.18.6 DBLINK_ENCRYPT_LOGIN

The parameter DBLINK_ENCRYPT_LOGIN, when set to TRUE, prevents unencrypted passwords from being sent to remote servers. This parameter is supported only for backwards compatibility to Oracle database versions 6 and 7. All attempts between later versions are always encrypted. This parameter has been desupported as of Version 9, Release 2 (9.2). The default value for this parameter is FALSE, the required value for this parameter is TRUE.

- *(DO3698: CAT II) The DBA will set the DBLINK_ENCRYPT_LOGIN parameter to TRUE for database versions 9.0.1 and earlier.*

B.18.7 SQL92_SECURITY

The initialization parameter SQL92_SECURITY when enabled or set to TRUE, specifies that SELECT privileges are required during an UPDATE or DELETE function when a where clause specifying column values is present. When set to false, UPDATE and DELETE privileges allow SELECT actions in such cases. Distinct SELECT privileges will be required on all tables where select statements are performed. The SQL92_SECURITY parameter will be set to TRUE.

- *(DO3540: CAT III) The DBA will set the SQL92_Security parameter to TRUE.*

B.18.8 UTL_FILE_DIR

The parameter UTL_FILE_DIR was added to support Oracle packages that allow the reading and writing of text files to an operating system file. This parameter, if used, will be set to a specific operating system directory where application procedures/programs can read and write files. This means the directory will exist and have the permissions correctly set to allow Oracle background processes to write to the directory. Errors will result if this initialization parameter is used on a directory to which Oracle cannot *read/write*. If this parameter is set to "*," then all directories are allowed *read/write* access. The UTL_FILE_DIR parameter will not be set to "*."

Recognize that whatever directory the UTL_FILE_DIR parameter is set to will allow the Oracle database account to access this directory. Also, the UTL_FILE_DIR parameter may be set to multiple values. All directories in the UTL_FILE_DIR parameter are available to be read or written to. The UTL_FILE package will not be granted to PUBLIC.

- *(DO3547: CAT I) The DBA will set the UTL_FILE_DIR parameter to a specific operating system directory or directories. This directory will exist and have the permissions set so Oracle background processes may read/write to this directory.*

- *(DO3547: CAT I) The DBA will ensure the UTL_FILE_DIR parameter is not set to "*".*

- *(DO3475: CAT II) The DBA will deny access to the UTL_FILE package to PUBLIC.*

B.18.9 07_DICTIONARY_ACCESSIBILITY

The 07_DICTIONARY_ACCESSIBILITY parameter controls restrictions on SYSTEM privileges. If the parameter is set to TRUE, access to objects in the SYS schema is allowed. If this parameter is set to FALSE, SYSTEM privileges that allow access to objects in other schemas do not allow access to objects in the dictionary or SYS schema.

When the 07_DICTIONARY_ACCESSIBILITY=FALSE, then the SELECT ANY TABLE privilege will allow access to views or tables in any schema except the SYS schema. The system privilege EXECUTE ANY PROCEDURE would allow access to procedures in any schema except the SYS schema. If you need to access objects in the SYS schema, then you must be granted the explicit object privilege. The following roles that can be granted to the DBA also allow access to dictionary objects:

- SELECT_CATALOG_ROLE
- EXECUTE_CATALOG_ROLE
- DELETE_CATALOG_ROLE

- *(DO3685: CAT III) The DBA will set the 07_DICTIONARY_ACCESSIBILITY parameter to FALSE.*

B.18.10 REMOTE_LOGIN_PASSWORDFILE

The REMOTE_LOGIN_PASSWORDFILE initialization parameter specifies whether Oracle uses a password file and, if in use, how many databases can use the password file. Setting the parameter to NONE signifies that Oracle should ignore any password file meaning that administrative access is granted by virtue of membership in the specified operating system Oracle DBA group. Setting the parameter to EXCLUSIVE signifies that the password file can be used by only one database. The password file requires remote DBAs to use their own individual DBA accounts to authenticate to the database for administrative database operations. Setting the parameter to SHARED allows more than one database to use the password file, however, the only account recognized by the password file is the SYS account. The REMOTE_LOGIN_PASSWORDFILE will be set to NONE unless remote administration is required. If remote database administration is required then the REMOTE_LOGIN_PASSWORDFILE parameter will be set to EXCLUSIVE. Remote database administration requires encrypted communications to the database and a dedicated administrative listener is recommended.

You can create a password file using the password file creation utility ORAPWD, or for selected operating systems you can create this file as part of your standard installation. You can also reference your operating system-specific Oracle documentation for information on using the installer utility to install the password file. The types of filenames allowed for the password file are operating system specific. Some platforms require the password file to be a specific format and located in a specific directory. Other platforms allow the use of environment variables to specify the name and location of the password file. See your operating system-specific Oracle documentation for the names and locations allowed on your platform.

If you are running multiple instances of Oracle using the Oracle 9i Real Application Clusters (Oracle Parallel Server), the environment variable for each instance should point to the same password file.

It is critically important to the security of your system that you protect your password file and the environment variables that identify the location of the password file. Any account with access to this file could potentially compromise the security of the connection.

- *(DO3546: CAT III) The DBA will set the REMOTE_LOGIN_PASSWORDFILE parameter to EXCLUSIVE or NONE unless justified and documented with the IAO.*

- *(DO0291: CAT II) The DBA will protect the environment variable identifying the location of the password file.*

B.18.11 AUDIT_SYS_OPERATIONS

The AUDIT_SYS_OPERATIONS initialization parameter introduced with Oracle version 9.2 enables auditing of actions performed by the SYS, SYSDBA, or SYSOPER accounts. When set to TRUE, actions performed as SYS or with a SYSDBA or SYSOPER connection are audited regardless of the AUDIT_TRAIL setting. The audit records generated are stored in the OS audit file in the $ORACLE_HOME/rdbms/admin directory or in the Windows event log. The AUDIT_SYS_OPERATIONS parameter will be set to TRUE.

- *(DO0241: CAT II) The DBA will set the AUDIT_SYS_OPERATIONS parameter to TRUE on Oracle database versions 9.2 or later.*

B.18.12 GLOBAL_NAMES

The global names parameter value of TRUE requires that database links be defined with the same name as the database to which they connect. This prevents inadvertent connections to the wrong database and simplifies management of database links.

- *(DO0242: CAT III) The DBA will set the GLOBAL_NAMES parameter to TRUE.*

B.18.13 _TRACE_FILES_PUBLIC

(This parameter is an undocumented parameter.) The setting of TRACE_FILES_PUBLIC = TRUE allows all database accounts access to trace files. TRACE_FILES_PUBLIC will be set to FALSE.

- *(DO0243: CAT II) The DBA will set the _TRACE_FILES_PUBLIC parameter to FALSE.*

B.18.14 MAX_ENABLED_ROLES

The MAX_ENABLED_ROLES parameter specifies the number of roles that may be active for a single database session at one time. Setting this parameter may provide additional assurance that application roles are being enabled and disabled in accordance with design. The default value of this parameter is 30. Consider adjusting this to a lower value if it is appropriate for your application or environment.

B.18.15 REMOTE_LISTENER

The REMOTE_LISTENER parameter causes the database to register with a listener located on a separate host machine. The configuration and management of the remote listener would be outside the security domain of the database host system. Remote listeners should not be used unless required.

B.18.16 AUDIT_FILE_DEST (UNIX Only)

The AUDIT_FILE_DEST parameter specifies the directory where the Oracle database audit trail will be written on the host system. The AUDIT_TRAIL=OS must be specified for this parameter to take effect.

- *(DO0234: CAT II) The DBA will specify a valid and protected directory for the AUDIT_FILE_DEST.*

B.18.17 USER_DUMP_DEST

The USER_DUMP_DEST parameter specifies the host directory where database session trace files are written. The USER_DUMP_DEST parameter will be set to a valid and protected directory.

- *(DO0235: CAT II) The DBA will specify a valid and protected directory for the USER_DUMP_DEST.*

B.18.18 BACKGROUND_DUMP_DEST

The BACKGROUND_DUMP_DEST parameter specifies the host directory where the Oracle alert log and trace files for the Oracle background processes are written. The BACKGROUND_DUMP_DEST parameter will be set to a valid and protected directory.

- *(DO0236: CAT II) The DBA will specify a valid and protected directory for the BACKGROUND_DUMP_DEST.*

B.18.19 CORE_DUMP_DEST

The CORE_DUMP_DEST parameter specifies the host directory where the Oracle core files are written. This parameter is applicable only for UNIX systems. The CORE_DUMP_DEST parameter will be set to a valid and protected directory.

- *(DO0237: CAT II) The DBA will specify a valid and protected directory for the CORE_DUMP_DEST.*

B.18.20 LOG_ARCHIVE_START

The LOG_ARCHIVE_START parameter when enabled starts redo log archiving at the time of instance startup. The database must be in archive log mode for this parameter to take effect.

B.18.21 LOG_ARCHIVE_DEST

This parameter requires that ARCHIVELOG mode be enabled on the database. If applicable, the DBA will set this parameter to a valid and protected directory.

- *(DO0238: CAT II) If ARCHIVELOG mode is enabled, the DBA will set the LOG_ARCHIVE_DEST and LOG_ARCHIVE_DEST parameters to a valid and protected directory.*

B.18.22 LOG_ARCHIVE_DUPLEX_DEST / LOG_ARCHIVE_DEST_n

This parameter requires that ARCHIVELOG mode be enabled on the database. If applicable, the DBA will set this parameter to a valid and protected directory.

- *(DO0238: CAT II) If ARCHIVELOG mode is enabled, the DBA will set the LOG_ARCHIVE_DUPLEX and LOG_ARCHIVE_DEST parameters to a valid and protected directory.*

B.18.23 OS_AUTHENT_PREFIX

The OS_AUTHENT_PREFIX by default is set to the value ‘OPS$’. If OPS$ is used as the OS_AUTHENT_PREFIX, then accounts created with the IDENTIFIED BY clause may authenticate to the database using either OS authentication (connect /) or using database authentication (connect username/password). Setting the OS_AUTHENT_PREFIX to a value other than ‘OPS$’ prevents an OS account from being able to access a database account by the same name without providing a password. The OS_AUTHENT_PREFIX parameter will set to a value other than ‘OPS$’.

- *(DO3447: CAT II) The DBA will set the* OS_AUTHENT_PREFIX *to a value to other than “OPS$”.*

B.19 Oracle Operating System Security Requirements

The Oracle software is developed with a generic Oracle Kernel without regard for the target operating system. This ideology allows Oracle to be the same across all platforms with the only uniqueness being in the operating system specific shell code to provide access to the operating system.

This section describes any and all deviations from published OS STIGs as well as OS-specific Oracle security guidance. All OS STIG issues will not be addressed initially, but will be added as discovered in testing and implementation of all operating systems.

B.19.1 Oracle UNIX Specific Information

This section describes Oracle required deviations from the published DISA *UNIX STIG*. Reference the DISA *UNIX STIG* for additional guidance.

B.19.1.1 Oracle Operating System Software Owner Account

The installation of Oracle on a UNIX host requires that a unique UNIX userid be created and configured. This account becomes the owner of all Oracle application and datafiles and should be used only for the update and maintenance of the Oracle software. This account should be locked when not in use if feasible. The Oracle OS installation account will not be used in the performance of standard DBA activities. The individual DBAs will use their individually assigned OS accounts. This allows auditing of all operations from an OS perspective and allows the Oracle auditing to audit actions performed with the correct OS account. Access to the Oracle OS account will be restricted to site-authorized DBAs only. The Oracle software installation account will not be a member of the root group.

- *(DG0040: CAT III) The IAO will ensure that the UNIX Oracle OS installation account is not used when performing daily DBA activities.*

- *(DG0040: CAT III) The IAO will ensure that the UNIX Oracle OS installation account is only used when performing software installation and upgrades.*

- *(DG0040: CAT III) The IAO will restrict access to the UNIX Oracle OS installation account to site-authorized DBAs only.*

- *(DO3616: CAT II) The SA/DBA will configure all Oracle installation files and directories ownership to belong to the Oracle OS installation account.*

- *(DO0120: CAT II) The SA/DBA will ensure that the Oracle OS installation account is not a member of the root group.*

B.19.1.2 Oracle Process Owner Accounts

Individual accounts created for the individual Oracle component processes allows for the separation of security controls on directories and files and accountability for events. Separate accounts will be created for and used by the Oracle Listener, Intelligent Agent (if required), and the database processes.

- *(DO0121: CAT II) The SA/DBA will create and use individual UNIX OS accounts for the Oracle database processes, the Oracle Listener, and the Oracle Intelligent Agent account (if required).*

B.19.1.3 Oracle UNIX Profile Requirements

Every database user's **.**profile file may be owned by the individual's OS userid. The DISA *UNIX STIG* specifies this as an option. For database accounts, it may be advisable that the **.**profile be owned by the individual's OS userid to allow the OS user to customize the DBMS required environment variables and umask setting.

B.19.1.4 Non-Interactive/Automated Processing Database Accounts

Access to database accounts used for non-interactive and automated processing frequently requires that the username and password be stored and transmitted. Any storage of database account passwords on any system will be encrypted. On UNIX systems, the preferred method for encryption of a password within a file is by means of an application call such as the C language *crypt* function. Use of the UNIX crypt command to encrypt a file is not considered sufficiently secure, as the means to decrypt these files is widely known. If use of the crypt application function is not possible, then the UNIX crypt command may be made more secure by compressing the file and encrypting it several times.

In all cases, storage of the encryption key to the file and the application or batch file that encrypts the data will be protected from unauthorized access by the operating system. Access to these files and keys will be restricted to System Administrators and DBAs.

Passwords will not be stored unencrypted in UNIX environment variables.

If possible, non-interactive/automated processing accounts will use Oracle "identified externally" accounts (accounts that are authenticated by the host operating system) to connect to the database. These accounts eliminate the requirement to provide a separate username and password to authenticate to the database. These accounts may not be used for remote connections to the database.

- *(DG0067: CAT II) The DBA will ensure that passwords for non-interactive/automated processing database accounts are stored in encrypted format using a programming a FIPS 140-2 compliant encryption function.*

- *(DG0067: CAT II) The DBA will ensure that passwords for non-interactive/automated processing database accounts are stored in encrypted format. An application encryption function using FIPS 140-2 approved encryption will be used. If an application cannot be obtained, then file compression and multiple file encryptions using UNIX's crypt command may be used.*

- *(DG0067: CAT II) The SA/DBA will ensure that passwords are not stored unencrypted in UNIX environment variables.*

- *(DO0133: CAT II) The SA/DBA will restrict access to files containing logon credentials and encryption keys to SAs and DBAs.*

B.19.1.5 Oracle UNIX Groups

The Oracle software owner will have a umask setting of 022. System Administrators/DBAs will change their umask settings to 022 when performing database operations. This allows for all actions performed by the DBA to inherit the correct umask setting for the underlying DBMS. The DISA *UNIX STIG* specifies the umask to be set at 077. System Administrators are not required to follow this umask setting of 077 when performing database operations. Application user database accounts, application administrator accounts, or application developer accounts will not be members of the UNIX DBA group(s).

- *(DO0279: CAT II) The SA/DBA will configure the Oracle software owner to have a umask setting of 022.*

- *(DO0279: CAT II) System Administrators/DBAs will change their umask settings to 022 when performing database operations.*

- *(DO0145: CAT I) The SA/DBA will not make application user database accounts, application administrator accounts, or application developer accounts members of the UNIX DBA group(s).*

B.19.1.6 Oracle Files

Some Oracle database files require that the *suid* bit be set. The *suid* bit must be set on the following files as required by the successful operation of Oracle. This requirement is in accordance with the DISA *UNIX STIG*.

FILE NAME	***OWNER***	***GROUP***
dbsnmp	root	dba
oidldapd	oracle	dba
oracle	oracle	dba

Table 9. ORACLE SUID FILES

Oracle files will 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 OS account executing the file. Enabling the setgid bit on a directory causes all files created within the directory to be created with the group of the directory.

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 OS account executing the file and can cause the creation of unsecured files and unauthorized access. Oracle files with the setuid bit enabled will be restricted to administrator usage only. Only the following executable files are required by Oracle to have the setuid bit enabled—dbsnmp, oidldapd, and oracle.

- *(DO3615: CAT II) The SA/DBA will disable the setgid bit on all Oracle files.*

- *(DO3614: CAT II) The SA/DBA will restrict use of Oracle files with the setuid bit enabled to administrator usage only.*

- *(DO3614: CAT II) The SA/DBA will restrict the enabling of the setuid bit to the following Oracle executable files—dbsnmp, oidldapd, and oracle.*

B.19.2 Microsoft Windows Settings

This section describes Oracle required deviations from the *Windows NT/2000/NT Addendum* and Windows-specific Oracle security guidance. Reference that document for additional guidance.

B.19.2.1 File Permissions

The Oracle Windows Services are configured upon installation to use the Windows local SYSTEM account. Oracle recommends that Full Control access permissions to Oracle data and file directories be granted only to the Windows service account (SYSTEM). However, System Administrators, and DBAs may also require additional access to Oracle database files and directories for maintenance and update. Therefore, Full Control permissions may be granted to System Administrators and DBAs on Oracle directories and files. Application user OS accounts may be granted *read* access permissions to required Oracle application executables. The Everyone group will not be granted access permissions to any Oracle database files or directories.

- *(DO3613: CAT II) The DBA will restrict Full Control permissions to Oracle database files and directories to the Oracle service account, System Administrator accounts, and DBA accounts.*

- *(DO3613: CAT II) The DBA will remove all permissions on any Oracle database files or directories from the Everyone group.*

B.19.2.2 Registry Permissions

The Oracle Windows service account (SYSTEM) will be granted Full Control to the Oracle registry keys under HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE. The Everyone group will not be granted access to the Oracle registry keys. *Write* permissions to the Oracle registry keys will be restricted to DBAs, System Administrators, and the Oracle service account.

- *(DO3677: CAT II) The DBA will restrict Full Control permissions to Oracle registry values and keys to the Oracle service account, System Administrators, DBAs, and the Oracle installation account.*

- *(DO3677: CAT II) The DBA will restrict write permissions to Oracle registry values and keys to the Oracle service account, DBAs, System Administrators, and the Oracle installation account.*

- *(DO3677: CAT II) The DBA will remove all permissions granted to the Everyone group to all Oracle registry values and keys.*

B.19.2.3 Oracle Installation Log File

If the database is created through the Oracle Database Assistant client application and the typical option is chosen, the Server Manager reports all activities to a log file called SPOOLMAIN.LOG. This log file contains the passwords in plain text and has file permissions that allow anyone to view it. This file will be deleted after a database install.

- *(DO3847: CAT III) The SA/DBA will delete the log file SPOOLMAIN.LOG after database creation.*

B.19.2.4 Oracle Services for Windows

Oracle installs two main Windows services that support Oracle database operation. The OracleServiceSID, where SID is the Oracle instance name, is the Oracle database service. The OracleORACLE_HOMEService, where ORACLE_HOME is the value for the ORACLE_HOME variable, supports remote connections to the database. These two services use the same Oracle Windows SYSTEM security account for successful operation by default. The services may also be configured to use a Windows local administrator account. This security account for Oracle services must be provided the file and registry permissions as listed above.

B.19.2.5 Non-Interactive/Automated Processing Database Accounts

Access to database accounts used for non-interactive and automated processing frequently requires that the username and password be stored. Any storage of database account passwords on any system will be encrypted. On Windows systems, passwords for these accounts may be stored in text or application files or in the Windows registry. These passwords stored in host system files of any type or the Windows registry must be stored in encrypted format using a DES or stronger encryption algorithm.

In all cases, the encryption key, the encrypted password file or registry key, and the application or batch file that encrypts the data will be protected from unauthorized access by the operating system. Access to these files and keys will be restricted to System Administrators and DBAs.

Passwords will not be stored unencrypted in Windows environment variables.

If possible, non-interactive/automated processing accounts will use Oracle “identified externally” accounts (accounts that are authenticated by the host operating system) to connect to the database. These accounts eliminate the requirement to provide a separate username and password to authenticate to the database. These accounts may not be used for remote connections to the database.

- *(DG0067: CAT II) The DBA will ensure that passwords for non-interactive/automated processing database accounts are stored in encrypted format using a programming FIPS 140-2 compliant encryption function when possible.*

- *(DG0067: CAT II) The SA/DBA will ensure that database account passwords are not stored unencrypted in Windows environment variables.*

- *(DO0133: CAT II) The SA/DBA will restrict access to files and keys containing passwords and encryption keys to SAs and DBAs.*

B.19.3 OS/390 Specific Information

This section describes Oracle required deviations from the published DISA *OS/390 STIG* as well as OS/390-specific security requirements. Reference the DISA *OS/390 STIG* for additional guidance.

B.19.3.1 Oracle Library Security

Access to Oracle libraries will be restricted to three types of users—the Oracle software installer who is typically a Systems Programmer; Oracle DBAs who are OS/390 user accounts with special privileges to Oracle resources; and general database accounts for OS accounts that require access to specific Oracle utility executables. Following is the list of the minimum access restrictions. Oracle Corporation provides a host based Resource Access Control Facility (RACF) access level recommendation for each PDS used by the Oracle subsystem. Equivalent ACP authorization levels should be set for non-RACF installations.

DATA SET	*DATABASE ACCOUNT*	*DBA*	*SYSTEMS PROGRAMMER*
oran.orav.AUTHLOAD	None	Execute	Alter/Execute
oran.orav.CMDLOAD	Execute	Execute	Alter/Execute
oran.orav.DOC	None	Read	Alter/Update
oran.orav.DBA	None	Read	Alter/Update
oran.orav.INSTLIB	None	Read	Alter/Update
oran.orav.ISPCLIB	None	Read	Alter/Update
oran.orav.ISPMLIB	None	Read	Alter/Update
oran.orav.ISPPLIB	None	Read	Alter/Update
oran.orav.ISPSLIB	None	Read	Alter/Update
oran.orav.ISPTLIB	None	Read	Alter/Update
oran.orav.MACLIB	Read	Read	Alter/Update
oran.orav.PARMLIB	None	Alter/Update	Alter/Update
oran.orav.SRCLIB	Read	Alter/Update	Alter/Update
oran.orav.SQL	Read	Read	Alter/Update
oran.orav.SQLLIB	Read	Read	Alter/Update

Table 10. ORACLE OS/390 LIBRARY SECURITY

NOTE: oran.orav represents the high-level qualifier and second-level qualifier where the library is installed.

- *(DO3613: CAT II) The SA/DBA will restrict access to Oracle libraries according to the list above at a minimum.*

B.19.3.2 Oracle VSAM File Security

The systems programmer will own the VSAM system data sets used by the Oracle subsystem. These data sets contain the control files, database files, and redo log files. Other access to these files will be restricted to the Oracle database service userid.

- *(DO3613: CAT II) The SA/DBA will restrict access to the Oracle VSAM files to the Oracle database service userid and the systems programmer owner.*

B.19.3.3 Security for Oracle MPM Installations

B.19.3.3.1 Oracle MPM Restricted Commands

Oracle commands reside in the Oracle oran.orav.CMDLOAD library where oran is the high-level qualifier and orav is the secondary qualifier for the installed Oracle product version. Access to the following MPM commands will be restricted to System Programmers and DBAs:

- oran.oranv.CMDLOAD(CRTCNV) – Utility to convert CRT files into a load module format that can be used as input to the linkage editor.

- oran.oranv.CMDLOAD(MPMCMD) – Controls the operation of the Oracle subsystem. Every MPM parameter or command has an associated authority, which is SYS or ALL.

- oran.oranv.CMDLOAD(SVRMGRL) – This is a mpmparm privuser, which allows startup, shutdown, display, and connects internal actions against the database.

To restrict access to the above listed procedures, remove them from the CMDLOAD library and move them to a restricted library or use program control to restrict access to the command procedures in the CMDLOAD library.

- *(DO0380: CAT II) The SA/DBA will restrict access to command procedures CRTCNV, MPMCMD, or SVRMGRL systems programmers and DBAs.*

B.19.3.3.2 Oracle MPM Exits

The Oracle OS/390 managed exits provided for the OS/390 clients are Logon User Exit Point and User Role Exit Point. An Oracle for OS/390 client is defined as a user connecting to Oracle from batch, CICS, Oracle Access Manager from CICS, IMS/TM, Oracle Access Manager from IMS/TM, Open Edition MVS, TSO, and Oracle SQL*Net.

B.19.3.3.2.1 Oracle MPM User Logon Exit Point

The Oracle user logon exit point uses the operating system to authenticate users to the Oracle database. When configured, the Oracle server invokes the user logon exit point each time a user attempts to log on to Oracle. Protection of authentication credentials (username and password) by exit logon modules requires verification with encryption.

- *(DO0440: CAT II) The DBA will deny use of User Logon Exit Points for authentication to the Oracle database unless they have been reviewed and verified to protect authentication credentials with encryption.*

B.19.3.3.2.2 Oracle MPM User Role Exit Point

Current Oracle database security guidelines as set forth in this document prohibit the assignment of database role membership outside of the database. Thus the DBA retains the sole authority and responsibility for assigning database authorizations. In accordance with this, user role exit points will not be used by Oracle databases.

- *(DO0441: CAT II) The DBA will deny use of Oracle database user role exit points.*

B.19.3.4 Security for Oracle OSDI Installations

Oracle integration with OS/390 security features includes use of OS/390 resource profiles, Program Properties and APF authorizations, and association of OS/390 account identification with OSDI-defined services. Resource profiles for Oracle bind and administrative access protections will be defined in a dedicated resource class named *ORAB*. Resource profiles for Oracle OSDI command access protections will be defined in a separate dedicated resource class named *ORAO*.

- *(DO0442: CAT II) The SA/DBA will create a dedicated resource class named ORAO for Oracle OSDI command access protection.*

- *(DO0442: CAT II) The SA/DBA will create a dedicated resource class named ORAB for Oracle BIND and administrative access (SYSDBA, SYSOPER) protection.*

B.19.3.4.1 Access to the OSDI Subsystem

Access to OSDI Subsystem commands is controlled by resource profiles defined during Oracle installation. Oracle resource profiles for OSDI subsystem commands will be defined the ORAB dedicated resource class. Access to OSDI subsystem commands will be restricted to system programmers (or consoles), DBAs, and other IAO-authorized accounts by these resource profiles. The level of authorizations for OSDI commands will be set as shown in the following table where *ssn* is the Oracle OSDI subsystem name selected at installation:

COMMAND	***AUTHORIZATION LEVEL***
ssn.DEFINE	Update
ssn.ALTER	Control
ss*n*.SHOW	Read
ssn.START	Read
ss*n*.DISPLAY	Read
ssn.DRAIN	Read
ssn.RESUME	Read
ssn.STOP	Read

Table 11. ORACLE OSDI SUBSYSTEM AUTHORIZATIONS

- *(DO0443: CAT II) The SA/DBA will define the resource profiles for the Oracle OSDI commands as listed above.*

- *(DO0443: CAT II) The SA/DBA will restrict access to Oracle OSDI commands to system programmers, DBAs, and other IAO-authorized accounts.*

- *(DO0443: CAT II) The SA/DBA will assign levels of authorization to OSDI subsystem commands as listed above.*

B.19.3.4.2 Access to OSDI Services

Two services are defined within the Oracle OSDI architecture: the database service or instance and the Oracle network service. Connections to these services are accomplished using OSDI bind processing that performs an authorization check when the connection is requested. Authorization permission is verified by checking the OSDI bind resource profiles. If resource profiles have not been defined for a service, then all binds from all address spaces are allowed. Resource profiles will be defined for all Oracle OSDI services to a SAF-compliant security server. Two profiles will be created for each service. One profile if managed binds (used by CICS, IMS, and other Oracle services) and the other is for binds by normal applications (TSO, batch Oracle tools or Oracle applications). The name format for these profiles will be *ssn.service.UBIND and ssn.service.ABIND* where *ssn* is the OSDI subsystem name defined at installation, *service* is the Oracle database or Oracle Net service name, and the constant *UBIND* indicates application binds and the constant *ABIND* indicates managed binds. Resource profiles that protect binds to Oracle services will be created in the dedicated Oracle resource class.

- *(DO0444: CAT II) The SA/DBA will configure all binds to Oracle OSDI services to be authorized by a SAF-compliant security server.*

- *(DO0444: CAT II) The SA/DBA will create resource profiles that protect binds to Oracle OSDI services in the ORAB resource class.*

- *(DO0444: CAT II) The SA/DBA will create an ssn.service.UBIND and an ssn.service.ABIND profile for each Oracle database service.*

- *(DO0445: CAT II) The SA/DBA will restrict read authorization to the UBIND resource profiles to authorized Oracle database accounts.*

- *(DO0445: CAT II) The SA/DBA will restrict read authorization to the ABIND resource profiles to authorized Oracle database managed services.*

B.19.3.4.3 Access to SYSDBA and SYSOPER Privileges

Access to the Oracle database with database administration privileges SYSDBA and SYSOPER is controlled by SAF-defined resources. OS accounts granted *Read* authorizations to these resources are granted the privilege to connect to the Oracle database with administrative privileges. *Read* access to the SYSDBA and SYSOPER resources will be restricted to authorized DBAs and systems programmers. If the SYSDBA and SYSOPER resource profiles are not defined, then any userid may connect to the database service with these privileges. The SYSDBA and SYSOPER resource profiles will be defined in the dedicated Oracle resource profile. These profiles are named *ssn.service*.OPER and *ssn.service*.DBA where *ssn* is the OSDI subsystem name and *service* is the database service name.

- *(DO0446: CAT II) The SA/DBA will define the resource profiles ssn.service.DBA and ssn.service.OPER in the ORAB resource class.*

- *(DO0446: CAT II) The SA/DBA will restrict read access to the ssn.service.DBA and ssn.service.OPER to system programmers and authorized DBAs.*

B.19.3.4.4 Database Service Actions Subject to OS/390 Authorization

Some functions of the database services require OS/390 authorization to function including data set operations, OSDI bind operations, and access to UNIX System Services (USS). These functions operate under the OS/390 userid of the database service. Thus, the userid associated with the database service must be granted the authority to perform these functions. See the *Oracle9i Enterprise Edition Installation Guide* for instructions on associating OS/390 userids with OSDI services.

B.19.3.4.4.1 Data Set Creation and Deletion

The userid associated with the database service must be granted authorization to create and delete Oracle VSAM linear data set (LDS) files. Oracle invokes the OS/390 IDCAMS utility to perform data set creation and deletion operations. Authorization to create and delete Oracle data sets will be restricted to the associated Oracle service name userid, DBAs, and systems programmers.

- *(DO0447: CAT II) The SA/DBA will restrict authorization to create and delete Oracle data sets to the Oracle service userid and systems programmers.*

B.19.3.4.4.2 Data Set Open

The Oracle service requires *update-type* access to the VSAM LDS datafiles and *write* access to the database alert and diagnostic logs. It also requires read access to files including database parameter and SQL files. Access to Oracle data sets will be restricted to the Oracle service userid and systems programmers.

- *(DO0448: CAT II) The SA/DBA will restrict access to Oracle data sets to the Oracle service userid and systems programmers.*

B.19.3.4.4.3 OSDI Bind Authorization

OSDI binds initiated by OSDI services are used to establish database links. Database links are used to connect one database instance to another or to connect the Oracle Net service to an instance. OSDI bind authorizations are used to restrict which address spaces may connect to the local database service. OSDI bind authorization between OSDI databases and network services will be restricted to authorized database and network services.

- *(DO0445: CAT II) The SA/DBA will restrict read authorization to the ABIND resource profiles by other database and network services to authorized database and network services.*

B.19.3.4.4.4 UNIX System Service Access

Some database service functions require access to OS/390 UNIX System Services (USS). Access to USS by the Oracle database service must be granted to the database service userid for these functions to work. An OMVS segment must also be defined for the database service userid to support functions. The OMVS segment must be defined in accordance with OS/390 USS security guidance.

B.19.3.4.5 External Data Mover Actions Subject to OS/390 Authorizations

The Oracle backup and recovery utility and the External Data Mover (EDM) services require authorization to operate. If in use, the EDM services userid must be granted authorization to open sequential backup data sets for backup (output) and recovery (input). It may also be granted authorization to delete backup datasets for maintenance operations. The EDM services userid does not require access to Oracle VSAM LDS files to function.

B.19.3.4.6 Oracle Net Actions Subject to OS/390 Authorizations

In order to grant authorization to Oracle Net OSDI services, an OS/390 userid must be associated with the Oracle Net service. See the *Oracle9i Enterprise Edition Installation Guide* for instructions on associating OS/390 userids with OSDI services.

B.19.3.4.7 Authorizing Oracle Logon

Oracle user connections to the database may be authorized by the host operating system. Host authorization may be used to authenticate users accessing the database from a local host session or a remote host session. Oracle provides host OS authentication by use of a built-in SAF check or an external logon exit module. If the customer selects host authentication, the built-in Oracle SAF check will be used. Logon exit modules will not be used. The LOGIN_AUTH OSDI service parameter will be set to NONE or SAF. This setting applies only to database accounts that are IDENTIFIED EXTERNALLY.

- *(DO0449: CAT II) The DBA will set the LOGIN_AUTH OSDI service parameter to NONE or SAF.*

B.19.3.5 Oracle Access Manager

Oracle Access Manager allows applications running under CICS or IMS/TM to issue industry standard SQL statements against an Oracle server. No coding for proprietary application program interfaces (APIs) is required. Mainframe programs can execute distributed transactions spanning both OS/390 and Oracle data. Distributed transactions can be coordinated using CICS or IMS/TM to ensure the integrity of the entire transaction.

To prevent static storage of passwords in CICS and IMS/TM Oracle connect strings, CICS and IMS/TM connections to the database will use Oracle database accounts authenticated by the OS/390 host system. This requires that the LOGIN_AUTH initialization parameter be set to SAF and the CICS application id username be created within Oracle as IDENTIFIED EXTERNALLY.

- *(DO0450: CAT II) The DBA will configure CICS and IMS/TM connections to an Oracle database to be authenticated by the OS/390 host system.*

B.19.3.6 System Management Facility

The IBM System Management Facility (SMF) provides a facility for users to monitor, collect, and record a variety of system and job related information. The Oracle subsystem uses the standard SMF interface to write user records and the Oracle audit trail if specified to the SMF data sets. User records may contain Oracle subsystem accounting data and other Oracle information allowing Oracle installation sites to charge individual users for the resources they use.

Audit trail data must be protected in accordance with security policy. Access to any SMF data set storing Oracle audit data will be restricted to systems programmers and database security officers.

- *(DO0451: CAT II) The SA/DBA will restrict access to Oracle audit data stored in SMF data sets to systems programmers and database security officers.*

This page is intentionally left blank.

APPENDIX C MICROSOFT SQL SERVER SPECIFIC POLICY AND IMPLEMENTATION

C.1 Current SQL Server Version

The information contained in this appendix is specific to Microsoft (MS) SQL Server Versions 7 and 8. MS SQL Server Version 8 is marketed as SQL Server 2000. When version-specific information is presented, it will be labeled with the version to which it specifically applies.

SQL Server versions may be updated with hotfixes and service packs to address product bugs as well as to provide fixes for published security bulletins.

MS Support Services has defined a “product support lifecycle.” When a product is selected for support expiration, Microsoft provides a six-month notice. Microsoft publishes the list of hotfix availability end dates on its web site. Products that have been expired under standard support no longer receive published updates from Microsoft to address discovered security problems.

To protect your SQL Server environment, the DBMS version will not be an expired product. The SQL Server software service pack will be no earlier than the current service pack version minus one.

- *(DM0590: CAT I) The IAO will ensure that unsupported DBMS software is removed or upgraded prior to a vendor dropping support.*

- *(DG0002: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(DM1769, DM5408, DM0710, DM5145: CAT I) The IAO will ensure that the SQL Server version has all patches and hotfixes applied.*

C.2 SQL Server Meets C2 Security Requirements

SQL Server Version 8 (SQL Server 2000) is capable of being fully C2 compliant. The policies and information in this appendix incorporate most configuration settings that set SQL Server to meet the C2 specifications.

C.2.1 SQL Server Meets DAC Requirements

SQL Server meets DAC requirements in the following ways.

- SQL Server requires individual user logons.
- Database objects are owned by individual database accounts.
- The owner of an object by default is the only account with object privileges that allow access to a database object.

- Only authorized users or the owner can grant access to an object to other database accounts, roles, or to PUBLIC (this includes DBAs).

C.2.2 SQL Server Meets Identification and Authentication Requirements

SQL Server supports two choices for user authentication:

- SQL Server -based (by password),
- Host-based (by the underlying operating system / enterprise)

Of these two options, only the host-based authentication method meets C2 requirements. Windows and Windows Active Directory provide a Windows security identifier (SID) to SQL Server that provide the ability to audit activity by individual database accounts.

C.2.3 Secure Distributed Computing

Mutual authentication of databases enables secure distributed transactions between application servers, web servers, and database servers without compromising the user’s credentials. Mutual database authentication and strong user authentication may be accomplished by using industry-standard X.509 certificates or passwords. Microsoft refers to these systems as Linked Servers that specify an OLE DB provider and an OLE DB data source.

- *(DM3566: CAT II) The DBA will configure SQL Server to use Windows authentication only.*

C.2.4 SQL Server Meets Object Reuse Requirements

SQL Server meets the requirements to prevent Object Reuse in the following ways.

- All resources are cleared upon allocation.
- Files managed by SQL Server are cleared according to Windows object reuse policy.

C.2.5 SQL Server Meets Auditing Requirements

When SQL Server’s auditing is enabled, the auditing requirement of C2 is met in the following ways:

- Audit records for identification and authentication are generated.
- Audit records for access to protected objects are generated.
- Audit records for deletion of objects are generated.
- Audit records for administrative actions are generated.
- Audit records for other security relevant events are generated.
- Audit information is recorded and protected by DAC permissions.
- Audit records include the time, account, event type, and success or failure.
- Identification and authentication audit records include the computer name originating the event.
- Administrators may select and filter the review of audit data.
- The audit subsystem can be configured to prevent the loss of audit records.

C.3 SQL Server Access Controls

Access controls of database objects are an integrated feature of the SQL Server DBMS. Access permissions fall into three categories:

- Statement permissions
- Object permissions
- Implied permissions

Permissions may be granted directly to database accounts or to roles. Statement permissions grant database accounts the ability to create and configure the database and its items. Object permissions grant the ability to manipulate data and execute procedures within the database. Implied permissions are those privileges granted through membership in fixed server roles or through object ownership. Database object owners have the implied permissions to perform all activities on the objects they own.

C.3.1 sa Connection

The SQL Server **sa** pseudo database account will not be used. Only trusted connections will be used to access the SQL Server database.

- *(DM3566: CAT II) The DBA will deny use of the SQL Server sa pseudo database account.*

- *(DM3566: CAT II) The DBA will configure SQL Server to use trusted connections only.*

C.3.2 OS DBA Group

The installation of MS SQL Server does not create any Windows OS groups. A DBA OS group will be created and the authorized DBA accounts assigned as members to this group.

- *(DM0920: CAT II) The SA/DBA will create a DBA Windows OS group.*

- *(DM0921: CAT I) The SA/DBA will assign only IAO-authorized DBA Windows accounts to the DBA OS group.*

C.3.3 SYSADMIN Role

All DBA activities will be performed as an account within the **SYSADMIN** role. The **SYSADMIN** fixed server role provides full system privileges as well as allows the account to perform all database administration functions including:

- Instance startup, mount, and database open
- Instance shutdown, dismount, and database close
- Alter database backup, transaction log, and recover

- *(DM0500: CAT I) The DBA will assign only authorized DBAs the SYSADMIN role.*

- *(DM0500: CAT I) The DBA will deny the Windows BUILTIN\Administrators group the assignment to SYSADMIN role.*

- *(DM0922: CAT II) The DBA will ensure that DBA accounts are only used to support DBA activities.*

C.3.4 Default SQL Server Passwords

Any default passwords for database accounts created during an SQL Server installation will be changed after installation.

C.3.5 Default sa Password

The default **sa** password, used to connect as administrator, will be changed from the default installation value. Leaving the default password unchanged could result in unauthorized accounts accessing the server as sa, which provides them full database administration privileges.

- *(DM1459: CAT I) The DBA will password protect the SQL Server sa pseudo database account.*

- *(DM1459: CAT I) The DBA will change the SQL Server sa pseudo database account default password.*

C.3.6 SQL Server Agents Service Account

The MS SQL Server Agent services, MSSQLServer or MSSQL$Instancename for a named instance and SQLServerAgent, will not be run under the administrator or system accounts. A service account will be defined and will be a local Windows account unless a Windows domain account is required to support replication, remote procedure calls, or SQLMail. The SQL Server Agent services will use the same account. The service account will not be a member of the local or domain administrators group. The service account will be denied the interactive logon right. The service account must be added to the SQL Server SYSADMIN role. The SQL Server Agent service account requires the following rights:

- Act as part of the operating system
- Increase quotas
- Replace a process-level token
- Log on as a service
- Access this computer from network
- May require the logon as a batch job right

- *(DM0901: CAT II) The SA/DBA will configure the SQL Server services to run under accounts other than a Windows administrator or system account.*

- *(DM0901: CAT II) The SA/DBA will configure the SQL Server services to run under a single local Windows account or a Windows domain account if required.*

- *(DM0901: CAT II) The SA/DBA will configure the SQL Server services account so that it is not a member of a local or domain administrators group.*

- *(DM0901: CAT II) The SA/DBA will deny the SQL Server services account the interactive logon right.*

C.3.7 SQL Server Database Accounts and Windows OS Accounts

Matching SQL Server database accounts and Windows OS accounts contribute to a secure environment by limiting confusion over user identity. Limiting confusion of identity lessens the risks of improper permission assignments and aids in the monitoring of auditing logs. It is recommended that the DBA maintain SQL Server security account names in accordance with Windows account names. Windows groups may also be used to assign SQL Server roles and thereby eliminate the maintenance requirements for individual SQL Server account names altogether.

C.3.8 Guest Account

The SQL Server guest account allows Windows accounts without direct SQL Server authorization that have been authenticated to the Windows OS to access the database. It cannot be removed from the master and tempdb databases. The guest account will be deleted from all databases except the master and tempdb databases.

- *(DM1709 : CAT II) The DBA will delete the database guest account from all databases except the master and tempdb databases.*

C.3.9 SQL Server Non-Interactive/Automated Processing Accounts

C.3.10 Linked or Remote Servers

Linked or remote servers will only be configured to use Windows authentication. The capability to preserve a user’s identification, and, therefore, maintain DAC integrity, is currently available only in a Windows 2000 or later environment where the connections can be protected with Kerberos and account delegation can be used. When linking SQL Server databases, the connection will be authenticated using the current user's identification and passwords or certificates.

- *(DM3566: CAT II) The DBA will configure linked servers to use the user's current authentication to access the remote database.*

C.3.11 SQL Server Account Password Requirements

SQL Server authentication is dependent upon the underlying Windows platform. Password requirements will be implemented according to the *Windows NT/2000/XP Addendum.*

C.3.12 Predefined Roles

SQL Server contains two different default role types—fixed server roles and fixed database roles.

Fixed server roles are used for database administration and are applied system wide. The fixed server roles exist within the database and are granted to Windows local or global groups or to individual Windows accounts. These roles will only be used to support DBA activities.

Fixed database roles apply only within a particular database. A database account assigned a fixed database role has those privileges only in the database where they have been assigned. Fixed server and database roles will not be granted to application user database accounts, application administrator accounts, or application roles. Fixed server and database roles will not be granted to PUBLIC. Fixed server and database roles will not be granted to GUEST. The BUILTIN\Administrators group should be removed from the SYSADMIN role. The DBA OS group will be added to the SYSADMIN role.

Fixed server roles:
- sysadmin
- serveradmin
- setupadmin
- securityadmin
- processadmin
- dbcreator
- diskadmin

Fixed database roles:
- db_owner
- db_datareader
- db_accessadmin
- db_securityadmin
- db_ddladmin
- db_backupoperator
- db_datawriter
- db_denydatareader
- db_denydatawriter

- *(DM0530: CAT II) The DBA will ensure that SQL Server fixed server roles are only used to support DBA activities.*

- *(DM0530: CAT II) The DBA will not grant SQL Server predefined roles to PUBLIC or GUEST.*

- *(DM0530: CAT II) The DBA will not grant the above-listed SQL Server predefined roles to application user database accounts, application administrator accounts, application developer accounts, or application roles.*

- *(DM0500: CAT I) The DBA will ensure that the SYSADMIN role is not granted to the BUILTIN\Administrators OS group.*

- *(DM0500: CAT II) The DBA will grant the SYSADMIN role to the DBA OS group.*

C.3.13 SQL Server Privileges

SQL Server access privileges may be granted to database accounts, Windows groups, or database roles. Data may be protected down to the column level. Privileges to fixed roles cannot be modified.

C.3.13.1 Statement Privileges

SQL Server statement privileges will not be granted to PUBLIC or GUEST. The following list of SQL Server statement privileges will not be granted, directly or indirectly through the use of roles, to any application user, application administrator, application developer, or application role.
STATEMENT PRIVILEGES
BACKUP DATABASE
BACKUP LOG
CREATE DATABASE
CREATE DEFAULT
CREATE FUNCTION
CREATE PROCEDURE
CREATE RULE
CREATE TABLE
CREATE VIEW

Table 12. SQL SERVER STATEMENT PRIVILEGES

- *(DM1760: CAT II) The DBA will ensure that SQL Server statement privileges are not granted to PUBLIC or GUEST.*

- *(DM1760: CAT II) The DBA will ensure that the SQL Server statement privileges listed above are not granted to individual application user database accounts, application administrator accounts, application developer accounts, or application roles.*

C.3.13.2 Object Privileges

SQL Server object privileges include SELECT, INSERT, UPDATE, DELETE, REFERENCES, and EXECUTE. The references object privilege will not be granted to any application user, application administrator, or application role. No object privileges will be granted to PUBLIC or GUEST. Access to system tables is granted by default to public in each database. Careful attention must be paid to ensure that these permissions have been removed from PUBLIC.

- *(DM1715: CAT II) The DBA will ensure that application user database accounts, application administrator administrators, and application roles are not granted the references object privilege.*

- *(DM1715: CAT II) The DBA will ensure that object privileges are not granted to PUBLIC or GUEST.*

C.3.13.3 Job System Privileges

Jobs can be used to automate administrative procedures as well as T-SQL procedures. CmdExec and Active Scripting job steps issue or can issue operating system commands and will be restricted to use by DBAs. Access to the host operating system poses a security risk.

- *(DM3763: CAT I) The DBA will restrict use of CmdExec and Active Scripting job steps to DBAs.*

C.3.13.4 Grant Object Privilege

Application user database accounts, application administrator accounts, application developer accounts, or application roles will not have the grant option of any object privilege. PUBLIC and GUEST will not have the grant option of any object privilege.

- *(DM5144: CAT II) The DBA will ensure that application user database accounts, application administrator accounts, application developer accounts, and application roles do not have the administration option of any object privilege.*

- *(DM5144: CAT II) The DBA will deny PUBLIC and GUEST the grant option of any object privilege.*

C.3.14 Configuring Net Libraries

SQL Server automatically listens on all net-libraries installed on the server. Installed libraries should be restricted to only those necessary to provide access to the intended client base. The Multi-Protocol net-library is the most secure protocol without the availability of Kerberos in a Windows 2000 Active Directory environment as it allows for encryption of all network communications.

C.3.15 Windows System Permissions

SQL Server executables are created with specific system permissions. These permissions are critical to the correct operation of the software. Likewise, file ownership is designed to run as it is installed and should not be modified.

C.3.15.1 SQL Server Directories

All directories that are created as the result of an installation of SQL Server, including file permissions and groups, will not be modified to be more permissive from the initial installation. The SQL Server software owner will own all directories and files created by the installation of SQL Server. These files are located in the home directory on the server and include executable, parameter, and datafiles. Changing the permission or ownership values may impact the operation of the SQL Server software. These files should be secured by using access control methods native to the operating system.

- *(DM3769: CAT II) The SA/DBA will restrict access to all directories created by the installation of SQL Server to full control permissions granted to the SQL Server service account, the DBA OS group, the Administrators group, and the local SYSTEM accounts.*

- *(DM3769: CAT II) The SA/DBA will restrict access to all files created by the installation of SQL Server to full control permissions granted to the SQL Server service account, the DBA OS group, the Administrators group, and the local SYSTEM accounts.*

C.3.15.2 SQL Server Registry Permissions

Windows Registry keys pertaining to the SQL Server will be protected in accordance with the Windows NT/2000/XP Addendum and according to the following specifications:

- *(DM3775: CAT II) The SA/DBA will restrict access to the Windows registry keys under the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer (for a default instance) or HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MS SQL Server\InstanceName (for a named instance) to full control permissions granted to the DBA OS group, the Administrators group, the local SYSTEM account, and the SQL Server service account.*

- *(DM3775: CAT II) The SA/DBA will restrict read and write permissions to the HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\MSSQLServer and HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Perflib registry keys to the SQL Server service account and the DBA OS group.*

C.4 Auditing In SQL Server

Logon audit data is stored in both the Windows event log as well as the SQL Server 2000 error log. Logon auditing is configured at the SQL Server configuration level. The SQL Server logon audit level will be set to Failure or ALL to record unsuccessful logon attempts at a minimum. Event auditing may be turned on by setting the C2 trace flag or by defining a trace using SQL Profiler and configuring it for autostart. The C2 flag setting ensures that all required event auditing is set and that event auditing begins at database startup. This section describes how to configure custom event auditing on an individual database. Remember that at least the minimum level of auditing will be enabled for every database on all machines.

- *(DM1703: CAT II) The DBA will configure the SQL Server audit level to Failure or ALL.*

- *(DM0510, DM5432: CAT II) The DBA will enable auditing on the SQL Server database.*

C.4.1 Database Audit Trail Location

If appropriate, the SQL Server auditing should be configured to direct audit trail data to protected trace files and not to the system event logs. Unlike event log data, which can be viewed by System Administrators, trace file data is stored in binary format and can only be read by DBAs using SQL Server stored procedures. Audit data is also stored in the SQL Server error logs.

C.4.2 Database Audit Trail Protection

SQL Server audit data will be protected from loss. The audit log rollover capability will be configured for all audit traces on the database. This capability prevents audit data from being overwritten and halts the database when there is a failure in creating a new rollover file. SQL Server and SQL Server Agent error logs should be prevented from being overwritten before they have been backed up. The default number of error logs is six. Use of the error logs should be monitored and the number of logs adjusted as necessary to prevent data loss.

- *(DM5267: CAT II) The DBA will enable the file rollover capability on SQL Server audit traces.*

- *(DM5432: CAT II) The DBA will configure SQL Server to halt if a failure in audit file rollover occurs.*

C.4.3 Mandatory Auditing

Once auditing is enabled at the database level, specific auditing instructions will be issued from a DBA account. To activate required audit options, a DBA will enable the C2 audit option or define a trace to be run upon server startup configured with selected audit events as follows:

- db_owner
- Audit Add DB User Event
- Audit Add Logon to Server Role Event
- Audit Add Member to DB Role Event
- Audit Add Role Event
- Audit Add Logon Event
- Audit App Role Change Password Event
- Audit Backup/Restore Event
- Audit Change Audit Event
- Audit DBCC Event
- Audit Logon Event
- Audit Logon Change Password Event
- Audit Logon Change Property Event
- Audit Logon Failed Event
- Audit Logon GDR Event
- Audit Logout Event
- Audit Object Derived Permission Event
- Audit Object GDR Event
- Audit Object Permission Event
- Audit Server Starts and Stops Event
- Audit Statement GDR Event
- Audit Statement Permission Event

- *(DM5432, DM0510, DM5268: CAT II) The DBA will enable all auditing options presented above.*

C.4.4 Value Based Auditing

SQL Server does not supply a value based auditing capability. If value based auditing is required, the application must include this in its design. All audit data collected should be protected by access controls and one-year maintenance.

C.4.5 Audit Trail Maintenance

The SQL Server audit trail requires periodic maintenance. The audit trail data will be backed up and then purged routinely. The disk where the audit trail is located must be checked routinely to ensure that free space is available for the audit trail table to grow. If space is not available, then all activity upon the DBMS will stop until space is made available by the System Administrator. The sizing of the audit trail disk space and the maintenance of the audit trail are specific to each application. Audit trail data will be maintained for a minimum of one year.

- *(DG0030: CAT III) The DBA will ensure that the audit trail data is backed up.*

- *(DG0030: CAT III) The DBA will ensure that the audit trail data is maintained for a minimum of one year.*

C.5 Encrypting SQL Server

SQL Server will not maintain passwords. The Windows OS platform will maintain passwords in accordance with the *Windows NT/2000/XP Addendum*. Logon session information is encrypted through the standard Windows logon process. This process also applies to connections for distributed/replicated SQL servers.

C.6 SQL Server Replication

Replication allows for partial or full copies of the Publisher database to Subscriber databases that are maintained on remote servers. Snapshot replication updates the subscriber databases to defined points in time by applying an image of the Publisher database. Transaction replication updates the subscriber databases by applying transactions made on the Publisher database to subscriber databases. Merge replication allows updates to the subscriber databases that are applied in coordination with updates to the Publisher database.

The replication agents (Snapshot, Merge, Distribution, Log Reader, Queue Reader, and others) run under the SQL Server Agent service account. For replication purposes, this account must be a Windows domain account and, as specified earlier in this document, will not be a local or domain administrator or system account. The replication agents connect to one or more remote servers or instances run under the security context of the SQL Server Agent service account when using trusted connections.

C.6.1 Replication Role Requirements

Permissions to support and administrate replication functions are automatically assigned to the sysadmin, db_owner, or replmonitor SQL Server roles at installation time. Replication administration permissions will not be modified from their default assignments. Replication administration permissions will be granted only to authorized application administrators and DBAs.

- *(DM2133: CAT II) The DBA will ensure that replication administration permissions are not modified from their default assignments.*

- *(DM2133: CAT II) The DBA will grant replication administration permissions only to authorized application administrators and DBAs.*

C.6.2 Securing the Distributor Database

The distributor database may reside locally on the SQL Server publisher database system or on a remote server. Both publisher and subscriber databases require connections to the distributor database. All database connections for replication agents will use Windows authentication logons. This requirement is not compatible with some replication configurations.

- *(DM3566: CAT II) The DBA will configure all database connections for replication agents to use Windows authentication logons.*

C.6.3 Snapshot Folder Security

Snapshot folders must be stored on a protected network share. The network share may be located on the distributor database system or on another server. The snapshot folder will be located on an explicit share and not a Windows administrative share. The snapshot folder permissions will be set to system, administrator, OS DBA group Full Control, SQL Server Agent domain account Read, Write. Subscribers will also be granted read permissions on the snapshot folder in a replication pull configuration.

- *(DM 2133: CAT II) The DBA will configure the snapshot folder location on an explicit share and not on a Windows administrative share*

- *(DM 2133: CAT II) The DBA will set snapshot folder permissions to SYSTEM and ADMINISTRATOR Full Control, SQL Server Agent domain account read and write*.

C.6.4 Publication Access Lists

Permissions to publications for replication are maintained in the Publication Access List (PAL). All replication agent logon accounts must have entries in the PAL in order to participate in the replication process. Therefore, the SQL Server Agent service domain account must be entered in the PAL.

C.6.5 Agent Logon Security

All replication agents, as mentioned previously, will use Windows authentication and run under the security context of the SQL Server Agent service Windows domain account.

C.6.6 Security and Replication over the Internet

Virtual Private Network (VPN) offers the most secure connection for establishing replication between databases over the Internet. Replication may also occur via a proxy server. The configuration of replication over the Internet must be in compliance with the DISA *Enclave Security STIG*. Care should be taken to protect the data by employing replication filters and locating the distribution database and snapshot files in protected and audited locations as appropriate.

C.7 Naming Conventions And File Locations

By default, SQL Server installs software and datafiles to subdirectories under \Program Files\Microsoft SQL Server\Mssql. Instance names, introduced in SQL Server 2000, must be unique to the servers running multiple instances and are limited to 16 characters.

C.7.1 Instance Naming Standards

The multiple instance capability was introduced in SQL Server 2000. This allows the naming of an instance that is reflected in the SQL Server service name, associated file directories, and registry hives. Unlike Oracle DBMS architecture, SQL Server allows only one instance for one or more databases, but allows many databases per instance.

- *(DM0660: CAT III) The DBA will not include a version number, SQL Server-related or otherwise, in the SQL Server production database instance names.*

C.7.2 SQL Server Datafile Naming Standards

There should be a separate datafile for each database within the instance. Datafiles associated with a database will contain the database name or prefix identifying the database and a sequence number representing each additional datafile needed to support the database.

Following are the SQL Server datafile file types and their recommended naming conventions:

FILE NAME	***DATABASE***	***SEQUENCE NUMBER***	***EXT.***
MASTER.MDF	Master	01 or none for the primary datafile	.MDF
MYDATA01.MDF	Client tablespace for ACME application	01 or none for the primary datafile	.MDF
MYDATA02.MDF	mydata	02 for 1st secondary datafile for same database, 03 for next secondary datafile for same database, etc.	.MDF

FILE NAME	***DATABASE***	***SEQUENCE NUMBER***	***EXT.***
TEMPDB.MDF	tempdb	01or none for the primary datafile	.MDF
PUBS.MDF	pubs	01 for the 1st datafile, 02 for 2nd datafile for same tablespace, etc.	.MDF
ERRORLOG	N/A external to database	Extension number updated sequentially on system restart	.n
MASTLOG	N/A external to database	01 or none for the 1st datafile, 02 for 2nd datafile for same database, etc.	.LDF
AUDITTRACE_date	N/A external to database	Date identifies copy of audit trace file	.TRC

Table 13. SQL SERVER FILE TYPES

C.7.3 Databases

Databases should be named with eight or fewer characters using a name descriptive enough to identify the function of the data contained within the database. Databases should be created to store tables and indexes to support the application hosted in the database.

The system database MASTER will be located in a separate database and should be named MASTER. The MASTER database will reside within in its own unique datafile(s). The system database TEMPDB will be located in a separate database and should be named TEMPDB. The TEMPDB database will reside within its own unique datafile(s). The system database MODEL will be located in a separate database and will reside within in its own unique datafile(s). The system database MSDB will be located in a separate database and will reside within in its own unique datafile(s). Application databases will be located in separate databases and will reside within their own datafiles. When applicable, the database will have Backup Transaction Log selected. NOTE: This is dependent on the database’s functionality.

- *(DM0550: CAT II) The DBA will locate the system database **SYSTEM** in a separate database that resides within its own unique datafile(s).*

- *(DM0550: CAT II) The DBA will locate the system database **TEMPDB** in a separate database that resides within its own unique datafile(s).*

- *(DM0550: CAT II) The DBA will locate the system database **MSDB** in a separate database that resides within its own unique datafile(s).*

- *(DM0550: CAT II) The DBA will locate the miscellaneous system database **MODEL** in a separate database that resides within its own unique datafile(s).*

- *(DM0550: CAT II) The DBA will locate the application databases in separate databases that reside within their own unique datafile(s).*

C.8 Initialization Parameters

This section covers SQL Server configuration options that have a security impact on the database and how these parameters are required to be set. This section is not intended to cover all SQL Server configuration options.

C.8.1 ALLOW UPDATES

The ALLOW UPDATES parameter specifies whether direct updates may be made to the system tables. When “allow updates” is disabled, database accounts cannot make updates to the system tables.

- *(DM1757: CAT II) The DBA will disable or set to 0 the ALLOW UPDATES parameter.*

C.8.2 C2 AUDIT MODE (SQL Server 2000 only)

The C2 AUDIT MODE parameter specifies whether or not automatic auditing of security events is enabled. The C2 AUDIT MODE will be used if a custom audit trace that meets this STIG's audit requirements is not defined and enabled.

- *(DM0510: CAT II) The DBA will enable the C2 AUDIT MODE (set to 1) if no custom defined audit trace is enabled.*

C.8.3 REMOTE ACCESS

The parameter REMOTE ACCESS, when set to 1, allows logons from remote servers running SQL Server. It is used with remote stored procedures and replication. For a more secure database environment, this parameter will be set to 0 unless replication is in use on the database or the requirement is fully justified and documented with the IAO.

- *(DM2142: CAT III) The DBA will disable the REMOTE ACCESS parameter (set to 0) unless replication is in use on the database or the requirement is fully justified and documented with the IAO.*

C.8.4 SCAN FOR STARTUP PROCS

The parameter SCAN FOR STARTUP PROCS, when set to 1, sets SQL Server to scan for startup procedures at startup time. Startup procedures may be planted by intruders to take effect at the next startup time. The SCAN FOR STARTUP PROCS parameter will be set to 0 unless fully justified and documented with the IAO.

- *(DM1761: CAT II) The DBA will disable the SCAN FOR STARTUP PROCS parameter (set to 0) unless fully justified and documented with the IAO.*

C.9 Stored Procedures

Access to system-defined stored procedures, like all other database objects, should be restricted to authorized users only. Access to object linking and embedding (OLE) stored procedures (sp_OA<name>) and registry access procedures (xp_REG<name>) will be restricted to DBAs unless fully justified and documented with the IAO.

User-defined stored procedures will be stored in encrypted format for additional protection. It is recommended that system-defined stored procedures be reviewed on a regular basis to discover if any unauthorized modifications have been made. Modification of system stored procedures is one method used by Trojan horses.

- *(DM2095: CAT II, DM2119: CAT II) The DBA will restrict access to object linking and embedding (OLE) stored procedures (sp_OA<name>) and registry access procedures (xp_REG<name>) to DBAs or remove them from the system unless fully justified and documented with the IAO.*

- *(DM1803: CAT II) The DBA will ensure that all user-defined stored procedures are stored in encrypted format.*

C.10 Extended Stored Procedures

Extended stored procedures run under the host operating system with full security rights. This allows for the potential for unauthorized users to gain access to the operating system. User-defined extended stored procedures will not be used. System-defined extended stored procedures will be limited to use by authorized DBAs only, unless fully justified and documented with the IAO. Extended stored procedures that are not required will be removed from the system. The XP_CMDSHELL extended stored procedure will be removed from the system unless fully justified and documented with the IAO. This extended stored procedure provides direct access to the host operating system.

- *(DM1762: CAT III) The DBA will prevent creation and use of user-defined extended stored procedures.*

- *(DM1762: CAT III) The DBA will restrict use of system-defined extended stored procedures to authorized DBAs only unless fully justified and documented with the IAO.*

- *(DM1762: CAT III) The DBA will remove all extended stored procedures that are not required from the database and host system.*

- *(DM1758: CAT I) The DBA will remove the XP_CMDSHELL extended stored procedure from the system unless fully justified and documented with the IAO.*

C.11 Object Encryption

SQL Server provides the capability to encrypt stored procedures, triggers, and views. Encrypting these objects provides an extra layer for security by preventing the viewing of the source for these objects and, therefore, the underlying structure of data objects they access. While experienced hackers easily defeat the added protection of object encryption, it does prevent the revealing of unnecessary information by less sophisticated attempts. Custom and GOTS stored procedures in the database will be encrypted.

- *(DM1803: CAT II) The DBA will ensure that GOTS or custom application stored procedures are stored in encrypted format.*

C.12 Database Backup File and Restore Protection

To ensure backup file protection, database backup and recovery activities should not be performed across the network. Access permissions to backup files will be restricted to System Administrators. Restore permissions on databases will be restricted to DBAs and database owners.

- *(DG0064: CAT II) The DBA will restrict access permissions to database backup files to System Administrators.*

- *(DG0063: CAT II) The DBA will restrict restore permissions on databases to DBAs and/or the database owners.*

C.13 SQL Server Installed Services

All services provide a mechanism for potential security vulnerabilities. Installed services will be limited to those required for the operation of the SQL Server database and to those services required to support the application. By default, SQL Server requires the SQL Server database engine (MSSQLServer or MSSQL$InstanceName) and the SQL Server Agent (SQLServerAgent service). Optional SQL Server services required by the application will be justified and documented with the IAO.

- *(DM0900: CAT II) The DBA will restrict use of SQL Server services to those required to support database operation and application support.*

- *(DM0901: CAT III) The DBA will justify and document optional SQL Server Services with the IAO.*

C.14 SQL Mail

SQL Server provides interaction to e-mail in two ways. SQL Mail is the process used by the MSSQLServer Service. This process uses a MAPI connection to a mail host.

SQL Mail can introduce many vulnerabilities into a system. Incoming mail commands can contain malicious code or viruses.

- *(DM0900: CAT II) The DBA will ensure that SQL Mail is not implemented.*

The SQLServerAgent uses its own mail that is configured and controlled separately from the SQL Mail.

- *(DM0901: CAT III) The DBA will ensure that use of the SQLServer Agent e-mail notification is documented with the IAO.*

C.15 Microsoft Data Engine/Microsoft SQL Server Desktop Engine (MSDE)

MSDE (Microsoft SQL Server Desktop Engine) is a fully functional version of Microsoft SQL Server. MSDE 2000 is included on the Microsoft Office 2000 CD. An earlier version of MSDE Version 1.0 was named Microsoft Database Engine. References in this document to MSDE include both versions.

MSDE contains all the functionality of SQL Server limited only by the number of user connections and the size of user databases. This robust functionality has the potential to increase vulnerabilities on a user’s workstation.

For purposes of securing MSDE, the use of MSDE is broken into two categories, a standalone version of MSDE and a shared version of MSDE. Standalone MSDE is defined as an installation of MSDE on a non-networked workstation or a networked workstation in which the Server Service has been disabled. Shared MSDE is defined as installation of MSDE installed on a server or on a networked workstation that does not have the Server Service disabled.

In general, all guidance listed in *Appendix C, Microsoft SQL Server Specific Policy and Implementation*, of this *Database Security Technical Implementation Guide*, will be applied to the MSDE environment. The following sections include MSDE specific issues and security policy changes to the Database for MSDE.

In cases where the MSDE supports distributed clients, the IAO will ensure that the distributed MSDE installations are configured in compliance with the policies outlined. Otherwise, the local DBA is responsible for these configurations unless stated specifically otherwise.

C.15.1 Current MSDE Version

MSDE may be updated with hot fixes and service packs to address product bugs as well as to provide fixes for published security bulletins.

To protect your environment, the MSDE version will not be an expired product.

- *(DM0590: CAT I) The IAO will ensure that MSDE versions in use are not a Microsoft expired version.*

C.15.2 MSDE Security Bulletins and Hotfixes

Periodically, security vulnerabilities may be discovered within the MSDE code. These vulnerabilities may be published and modifications to remove the vulnerability may be provided. DBAs will review security bulletins and install appropriate hotfixes as soon as possible.

- *(DM1769: CAT II) The DBA will apply all Microsoft security hotfixes.*

C.15.3 User Authentication

MSDE supports two choices for user authentication:

- SQL Server -based (by password),
- Host-based (by the underlying operating system/enterprise)

This STIG requires host-based authentication for SQL Server. There may be cases in a distributed application environment where host based security is not desirable. A shared MSDE environment must use host-based authentication. A standalone installation MSDE may use SQL Server based authentication if documented by the IAO. If this documentation is complete, there is also no need to maintain a DBA Windows OS Group. SQL logons should be created and added to the appropriate role. The built in SA logon will be password protected and not used for routine DBA duties. It is the responsibility of the IAO to ensure that the MSDE environment for each installation is clearly understood and documented and that the DBA responsibilities for each installation of MSDE are assigned. In some cases, configuration of the MSDE is the responsibility of a workstation/application configuration support personnel. In these cases, DBA configuration responsibilities belong to that person.

- *(DM3566: CAT II) The IAO will ensure that use of SQL Server Authentication in a standalone MSDE environment is documented.*

- *(DM3566: CAT II) The IAO will ensure that host-based (Windows) authentication is used for shared MSDE environments.*

- *(DM3566: CAT II) The DBA will deny use of the SA logon for routine DBA duties.*

- *(DM1459: CAT II) The DBA will password protect the SA account.*

- *(DM1459: CAT II) The DBA will change the SA logon password from its default value.*

C.15.4 Auditing

Logon audit data is stored in both the Windows event log as well as the MSDE error log.

- *(DM1703: CAT II) The DBA will set the audit level to Failure or ALL.*

- *(DM0510: CAT II) The DBA will enable auditing on the MSDE database.*

The following policy issues, which are required otherwise in this STIG, are optional within a standalone MSDE environment if documented with the IAO:

- *(DM5267: CAT II) The DBA will ensure that MSDE audit traces have the file rollover capability enabled unless documented with the IAO.*

- *(DM5432: CAT II) The DBA will ensure that MSDE is configured to halt if a failure in audit file rollover occurs unless documented with the IAO.*

- *(DM0510, DM5432: CAT II) The DBA will ensure that the auditing options presented in Section D.4.3, Mandatory Auditing, or the C2 audit option are configured for MSDE unless documented with the IAO.*

- *(DG0030: CAT II) The DBA will ensure that audit trail data is backed up unless documented with the IAO.*

- *(DG0030: CAT II) The DBA will ensure that audit trail data is maintained for a minimum of one year unless documented with the IAO.*

C.16 Sample Databases

Microsoft SQL Server ships with two sample databases: Northwind and Pubs. These databases contain many default permissions that do not conform to policy. Additionally, sample items can be used as an entry point into systems.

- *(DM0923: CAT II) The DBA will ensure that the sample Databases (Northwind and Pubs) are removed.*

C.17 SQL Server Service Components To Be Researched

Following is a list of Microsoft SQL Server components that were not researched in time for inclusion in this document. Any security considerations for these components will be reviewed and included in an updated version of this document.

- Full-text search
- Distributed Transaction Coordinator
- User-installed extended stored procedures
- Microsoft Meta Data Services
- Microsoft Analysis Services
- Microsoft English Query

This page is intentionally left blank.

APPENDIX D IBM DB2 UNIVERSAL DATABASE SPECIFIC POLICY

The information contained in this appendix is specific to DB2 Universal Database (UDB) Version 8 installations on Windows and UNIX servers. This guidance does not apply to DB2 version 8 on the z/OS platform or other DB2 versions on any other platforms and should not be used to evaluate the security of other platforms/versions. Please look for guidance specific to the z/OS platform in future versions of this STIG.

D.1 Current DB2 Version

D.1.1 UNIX/Linux/Windows

The latest supported version of DB2 UDB is 8.1, which was introduced November 2002 for UNIX and Windows platforms. The only other supported version at the time of this STIG release is Version 7.2. Versions 5 and 6 (with the exception on the OS/390 platform through June 2005) are no longer supported.

Support for version 7.1 (versions prior to Fixpak 3), expired 30 June 2003. The latest Fixpaks for the supported versions and the dates of their release are listed below. The build level for the installed version is listed in the DB2 Control Center / Help /About window. This information may be viewed at http://www-306.ibm.com/software/data/db2/udb/support/downloadv8.html.

Version	FixPak	Build Level	Release Date
7	12	s040510	May 2004
	11	s031208	December 2003
	10a	s030813	September 2003
8	6	s040616	February 2004
	5	s040212	February 2004
	4a	n040122	November 2003

Table 14. DB2 UDB FIXPAKS

- *(DG0001: CAT I) The IAO will ensure that unsupported DBMS software is removed or upgraded prior to a vendor dropping support.*

- *(DG0002: CAT II) The IAO will ensure that the site has a formal migration plan for removing or upgrading DBMS systems prior to the date the vendor drops security patch support.*

- *(DG0003: CAT II) The DBA will ensure that the latest FixPak has been installed for the installed version.*

D.2 DB2 Security Evaluations

DB2 UDB for Linux, UNIX, and Windows Version 8.1 was submitted for Common Criteria evaluation and NIAP approval in August 2003 against EAL4+ criteria. No prior versions of DB2 have been certified under Common Criteria or TCSEC.

D.3 DB2 Component Services

Component Services are optional components that provide additional capability to administer or operate the database. Component services are listed below along with any required security configurations necessary to protect them and their access to the database. Coverage in this section applies only to DB2 UDB on UNIX/Windows/Linux platforms. Coverage for components on the z/OS platform will be covered in a future release.

D.3.1 DB2 Administration Server (DAS)

The DAS provides job scheduling, remote administration of DB2 instances and databases, DB2 Discovery services, and alert notification via email and pagers to selected administrators.

The DAS runs as a service/daemon under a dedicated OS account name on the database server. This account requires DB2 SYSADM privileges and the OS privileges listed under the DB2 Operating System Accounts section in this appendix.

Privileges to administer the DAS are granted to members of the DAS administrative OS group. The default name for this group is dasadm1, but may be customized. The group name is specified to the DAS in the DAS configuration parameter dasadm_group. DAS administrative access will be restricted to authorized DBAs.

- *(N/A: CAT I) The IAO will ensure that only authorized DBAs are assigned the DAS administrative privilege.*

D.3.2 Data Links Support

DB2 Data Links provides DBMS access and some DBMS management capabilities to files stored locally on the database sever. Access to files registered with Data Links by users via the DB2 database is controlled by data links authorizations. Access to data links files may be controlled by data links authorizations. The data links security setting will be set to ON to enable access control to data links files. Access to data links files via the database server OS will be restricted to authorized users. Accounts granted access to data links directories are granted access to all files in the directory. The data links services or processes will be granted only the privileges required by other DB2 processes as defined elsewhere in this appendix.

- *(N/A: CAT II) The DBA will ensure that the Data Links link security setting is set to ON.*

- *(N/A: CAT II) The SA/DBA will ensure that access to Data Link file directories is restricted to SAs, DBAs, the DB2 installation account, and the DB2 service/daemon accounts.*

- *(N/A: CAT II) The SA/DBA will ensure that the Data Links services/process are granted the minimum privileges to operate.*

D.4 DB2 Access Controls

This section discusses DB2 authentication and authorization methods. DB2 uses OS authentication services to identify and validate users requesting access to the DB2 databases. DB2 also uses OS group membership to provide DB2 internal role-based assignment of database privileges.

D.4.1 DB2 Identification and Authentication

DB2 does not provide an independent authentication facility. Instead, it depends upon the database host operating system to provide user authentication. Access to a DB2 database is controlled by one of the following authentication methods:

> SERVER - This is the DB2 default authentication method. When the authentication mode selected is SERVER, DB2 passes the userid and password to the database host operating system for validation. If the operating system approves the authentication, the user is allowed to access the database. If the user is connecting to the database from other than the local server, the userid and password are sent in clear text across the network using this method.
>
> SERVER_ENCRYPT – This authentication method allows for a remote user to connect to the database by passing an encrypted userid and password. If the client specifies SERVER authentication, however, DB2 will accept an unencrypted userid and password to authenticate the user.
>
> CLIENT – This authentication method allows the user requesting connection to the database to authenticate to a remote operating system other than the database host. The database configuration parameters TRUST_ALLCLNTS and TRUST_CLNTAUTH can be set to refine the selection of remote host systems that are trusted to authenticate the remote user. These parameters can force authentication of the SERVER type depending on settings. If a userid and password are sent by the client with the database connection request, they are sent in clear text. The userid and password are ignored in cases where the server is not the authentication authority (both the client is trusted and the trust_clntauth and authentication modes are set to CLIENT).
>
> KERBEROS – This authentication mode is available only in environments where both the DB2 database server and the clients are running on a platform that supports Kerberos authentication. Current supported Kerberos platforms are Windows 2000, Windows XP, and Windows .NET. Kerberos authentication uses a secret key to authenticate the user to the database server operating system. It does not require the user to provide the username and password.

KRB_SERVER_ENCRYPT – Like the KERBEROS authentication method described above, this authentication mode accepts KERBEROS authentication from the client requesting it for connection to the database. However, if the client does not specify KERBEROS in its database catalog, the database will accept SERVER_ENCRYPT connection requests. This mode is useful for environments where both KERBEROS and non-KERBEROS clients may require connection to the DB2 database.

Passwords are considered sensitive information and are required to be encrypted when transmitted across a network. Therefore, the DB2 authentication mode will be set to SERVER_ENCRYPT, KERBEROS, or KRB_SERVER_ENCRYPT for the DB2 instance. The authentication mode is specified in the DB2 instance authentication configuration parameter.

DB2 clients will specify the same for connection to DB2 databases by specifying one of these values in the client database catalog.

- *(N/A: CAT II) The DBA will ensure that the instance authentication mode parameter is set to SERVER_ENCRYPT, KERBEROS, or KRB_SERVER_ENCRYPT.*

- *(N/A: CAT II) The DBA will ensure that DB2 clients specify SERVER_ENCRYPT, KERBEROS, or KRB_SERVER_ENCRYPT authentication for connection to the DB2 database.*

D.4.2 DB2 Administrative OS Groups

Administrative or privileged access to the database is controlled at the database server operating system level. The administrative privileges are defined for three groups and are called *authorities*. Below is a list of the authorities and the privileges assigned to them:

PRIVILEGE	SYSADM	SYSCTRL	SYSMAINT
Access any data in any table in any database in the instance	X		
Migrate a database	X		
Change the database manager configuration file (including specifying the groups having SYSCTRL or SYSMAINT authority)	X		
Grant DBADM authority	X		
Update a database, node, or distributed connection services (DCS) directory	X	X	
Force users off the system	X	X	
Create or drop a database	X	X	
Drop, create, or alter a table space	X	X	
Restore to a new database	X	X	
Update database configuration files	X	X	X
Back up a database or table space	X	X	X
Restore to an existing database	X	X	X
Perform roll forward recovery	X	X	X
Start or stop an instance	X	X	X

PRIVILEGE	SYSADM	SYSCTRL	SYSMAINT
Restore a table space	X	X	X
Run trace	X	X	X
Take database system monitor snapshots of a database manager instance or its databases.	X	X	X
Read log files	X	X	X
Create, activate, and drop event monitors.	X	X	X
Query the state of a table space	X	X	X
Update log history files	X	X	X
Quiesce a table space	X	X	X
Reorganize a table	X	X	X
Collect catalog statistics using the **RUNSTATS** utility.	X	X	X

Table 15. DB2 UDB ADMINISTRATIVE GROUP PERMISSIONS

SYSADM – This authority has full privileges at both the instance and database levels to perform all activities including instance and database configuration and maintenance as well as full access to the data level including INSERT, UPDATE, DELETE on all data objects.

SYSCTRL – This authority has full privileges to maintain the instance and databases, but cannot access data. This authority also does not include the privilege to mirate a database, change instance configuration parameters, nor grant DBADM authority.

SYSMAINT – This authority has access to maintain the instance and databases, but does not include privileges to update the database, node, or distributed connection services (DCS) directory, force users off the database, create or drop a database, drop, create or alter a tablespace, or restore a new database. This privilege also does not include access to the data level.

Other authorities and privileges are defined and managed within the DB2 database.

The operating system group names may be specified within the DB2 instance configuration parameters (called database manager parameters). On Windows, the default value of NULL assigns the local Administrator group to all authority groups. On UNIX, the default value of NULL assigns the primary group of the DB2 software installation account to all authority groups.

In order to preserve separation of duty, the database manager parameters specifying the SYSADM, SYSCTRL, and SYSMAINT operating system groups will be set to a custom group name. The SA or DBA will create Windows groups to be used specifically for this purpose. Please see detailed configuration parameter requirements in the DB2 Configuration Parameters section.

Please note that, on Windows systems, group membership is verified by DB2 at the domain or local level depending upon the users login. That is, if the user logs into a domain, DB2 will search only domain groups for membership. To force DB2 to search for membership on the local machine, specify the global and/or instance registry value DB2_GRP_LOOKUP=local. This can be set using the DB2set command line utility.

- *(N/A: CAT II) The SA/DBA will create custom groups on a Windows DB2 database server to be used exclusively for assignment of SYSADM, SYSCTRL, and SYSMAINT privileges by the DB2 instance.*

- *(N/A: CAT II) The DBA will ensure that the custom Windows groups created exclusively for DB2 SYSADM, SYSCTRL, and SYSMAINT authority assignment are specified in the DB2 database manager configuration parameters.*

- *(N/A: CAT II) The IAO will ensure that only authorized DBAs are assigned the SYSADM, SYSCTRL, and SYSMAINT authorities.*

D.4.3 DB2 Operating System Accounts

The following operating system accounts are used for support and maintenance of the DB2 system. The list of required accounts and their default names are listed below. Operating system privileges and permissions required for these accounts are included below.

> DB2 Installation OS Account – This is the account used to install the DB2 software on the database server host machine. This account becomes the owner of all the DB2 directories and files on a UNIX system. On Windows systems, the local Administrators group is assigned ownership. This account should not be confused with the DB2 instance account that has a default name of db2inst1.
>
> On UNIX systems, the account used for DB2 installation must have root authority. On Windows systems, this account must be a member of the local Administrators group, requires the *Act as part of the operating system* User Right, and Full Control permissions to the SQLLIB directory, subdirectories, and files, and Full Control permissions to the DB2 database directory, subdirectories, and files. This account will be created and used exclusively for installation and maintenance of the DB2 directories and files. The DB2 installation account will be assigned the least privileges on the database server required to support DB2 operation.

- *(N/A: CAT II) The DBA/SA will ensure that a custom account is created to support the DB2 installation.*

- *(N/A: CAT II) The DBA/SA will ensure that the DB2 software installation account is assigned the least privileges required to support operation of DB2 database and functions.*

- *(N/A: CAT II) The IAO will ensure that access to the DB2 installation account is restricted to IAO-approved users.*

DB2 Service/Daemon Accounts – These accounts are used to run Windows services or UNIX daemons required by the DB2 database. The default names for these accounts are dasusr1, db2inst1, and db2iadm1.

On UNIX systems, only the db2inst1 account is required for DB2 daemon use. The account requires access to the DB2 instance directories and files. It does not require root access to the database server.

On Windows systems, these accounts should not be members of the Administrator group and may be limited to the following user rights:

Act as part of the operating system
Create token object
Increase quotas
Log on as a service
Replace a process level token

Please see Windows OS permissions for minimum required file and directory permissions for these accounts. A custom account will be created to support the DB2 services/daemons and be assigned the least privileges.

- *(N/A: CAT II) The DBA/SA will ensure that a custom account is created to support the DB2 services/daemons and that this account is assigned the least privileges required to support operation of the DB2 instance.*

Instance OS Account – default name db2inst1 - This account is used by the UNIX daemon and Windows service to run the DB2 instance service/daemon. The instance OS account requires the same privileges as listed for the service/daemon accounts above.

DB2 Fenced User OS Account – default name db2fenc1 – This OS account is used to process applications spawned externally to the DB2 system processes. This account should have the least privileges assigned to it. The fenced user OS account requires only the privilege to log into the server and read/execute permissions to files stored in its home directory.

On UNIX systems, all OS accounts that use User Defined Functions (UDF) must also be granted read/execute access to the directory where the UDFs are stored. This access is granted via OS group membership.

- *(N/A: CAT II) The DBA/SA will ensure that the DB2 fenced user OS account is created and restricted to the minimum OS privileges required.*

- *(N/A: CAT II) The DBA/SA will ensure that users of DB2 fenced procedures are granted the least privileges via the OS fenced user group.*

DAS OS Account – default name dasusr1 – This OS account is used by the DAS service/daemon. The DAS service/daemon requires different OS privileges than the db2inst1 account and so requires a dedicated account. A separate, dedicated account will be created to support use of DAS. On Windows servers, the DAS OS account requires the Windows privileges required for the other DB2 services listed above. On UNIX servers, the DAS account must be the owner of the dasusr1 subdirectories and files.

- *(N/A: CAT II) The DBA/SA will ensure that a dedicated OS account is created to support the DB2 Administration Serve and restricted to the minimum OS privileges required.*

D.4.4 Default DB2 Passwords

During installation, DB2 prompts for OS account names and passwords for use by the DB2 database system. The default account names are db2inst1, db2fenc1, and db2as. Default passwords are not assigned. Custom passwords must be assigned at installation time.

D.5 DB2 Authorizations

DB2 separates database privileges into two separate types: Authorities and Privileges. Authorities are groups of privileges that allow the grantee access to certain system commands and are assigned at one time. Privileges are further divided into two types: database privileges and object privileges.

DB2 authorities are granted to users by means of operating system group membership assignment (SYSADM, SYSCTRL, SYSMAINT) and are described in detail in the DB2 Administrative OS Groups section. DB2 authorities granted to users with the DB2 GRANT command include DBADM and LOAD.

Privileges assigned by GRANT statements including all object privileges, database privileges, and the DBADM and LOAD authorities, may be granted to individual OS accounts or groups identified to the database. Both individual users accounts and groups are created on the operating system. Group membership is assigned using database server OS commands. Individual users and groups are defined separately for each DB2 database. Privileges are required to be assigned to groups to provide secure administration of privileges. However, in the case of DB2, this requires that the database server SA be responsible for defining group membership unless the DBA has SA privilege on the database server. OS Group membership for groups defined within DB2 will be reviewed a minimum of every 30 days by the DBA or IAO to discover unauthorized user assignment.

- *(N/A: CAT II) The DBA or IAO will ensure that OS group membership for groups defined within DB2 is reviewed for unauthorized assignment a minimum of every 30 days.*

D.5.1 DB2 Database Authorities

The DBADM authority includes the privileges allowing the execution of the following types of actions:

- Access any data in any table in the database
- Grant database privileges within the database
- Read log files
- Create, activate, and drop event monitors
- Query the state of a table space
- Update log history files
- Quiesce a table space
- Reorganize a table
- Collect catalog statistics using the **RUNSTATS** utility

These privileges apply only within the database where the DBADM privilege is assigned. The DBADM privilege is automatically assigned to the database creator at database creation time. The DBADM privilege will be granted only to authorized DBAs and application owner accounts.

- *(N/A: CAT II) The IAO will ensure that only authorized DBAs and application owner accounts are assigned the DBADM authority.*

The LOAD authority allows the grantee to load data in a database table to which they have been granted INSERT privileges. This authority also allows the grantee to issue QUIESCE TABLESPACES FOR TABLE, RUNSTATS, and LIST TABLESPACES commands.

D.5.2 DB2 Database Privileges

Database privileges grant the user the ability to create, modify, or drop database objects within the database. Following is the list of available database privileges:

- Connect – access the database
- Bindadd – create procedures, functions, and triggers
- Createtab – create tables and views
- Create_not_fenced – create procedures that are not fenced
- Implicit_schema – create a schema implicitly
- Load – load data into a table
- Quiesce_connect – access a quiesced database
- Create_external_routine – create a reference to a routine run externally to the database

The connect database privilege is required to allow access to the database. If the connect privilege is assigned to groups, the SA has the ability to grant connect privilege to the database by means of OS group membership assignment without approval by the DBA. In order to preserve separation of duties, connect privilege will not be granted to groups unless justified and documented by the IAO. In cases where the database server system administrator(s) also serve as DBA(s), it may be appropriate to allow groups connect privilege assignment.

- *(N/A: CAT II) The DBA will ensure that DB2 connect privileges are not assigned to groups unless justified and documented with the IAO.*

- *(N/A: CAT II) The DBA will ensure that application users are not assigned any database privileges except for the CONNECT database privilege.*

- *(N/A: CAT II) The DBA will ensure that database privileges with the exception of the CONNECT privilege are restricted to application owner accounts and DBA accounts on a production database.*

- *(N/A: CAT II) The DBA will ensure that database privileges with the exception of the CONNECT privilege are restricted to application owner accounts, application developer accounts, and DBA accounts on a development database.*

The connect privilege is assigned to PUBLIC by default upon database creation. The connect privilege will be revoked from PUBLIC upon database creation to prevent unauthorized access to the database. PUBLIC is also granted the CREATETAB, BINDADD, IMPLICIT_SCHEMA, USE database privileges at database creation time. These privileges should be restricted to application owner and DBA accounts. The DBA will revoke the CREATETAB, BINDADD, IMPLICIT_SCHEMA database privileges from PUBLIC at database creation time.

- *(N/A: CAT II) The DBA will ensure that PUBLIC is not granted the CONNECT, CREATETAB, BINDADD, IMPLICIT_SCHEMA database privilege.*

The CREATE_NOT_FENCED privilege allows the grantee to create procedures or functions that are processed within the DB2 database address or memory space. Applications run inside the database process may access database resources directly and inadvertently or maliciously corrupt database files or otherwise disrupt the database. Fenced procedures run outside the database process and within the security context of the OS account specified as the Fenced User operating system account. Unfenced procedures will not be defined within the database. The CREATE_NOT_FENCED database privilege will not be granted to any user.

- *(N/A: CAT II) The DBA will ensure that no database account is assigned the CREATE_NOT_FENCED database privilege.*

- *(N/A: CAT II) The DBA will ensure that no unfenced procedures or functions are defined with the database.*

The CREATE_EXTERNAL_ROUTINE allows the grantee to define a FENCED THREADSAFE routine. FENCED_THREADSAFE routines share a single process and may inadvertently or maliciously interfere with other FENCED THREADSAFE routines sharing the process. FENCED THREADSAFE routines should be tested carefully to ensure integrity. Use of FENCED THREADSAFE routines is left to the discretion of the application designer. The CREATE_EXTERNAL_ROUTINE will be restricted to application owner accounts.

D.5.3 DB2 Object Privileges

Object privileges grant the user the ability to read, add, modify, or delete data within existing database objects. Some object privileges also modify the structure of existing database objects. Following is the list of DB2 object privileges separated into those that alter object structure and those that do not. Object types accessed by these privileges are listed in parentheses.

Allow object structure alterations:
Alterin (schema)
Createin (schema)
Dropin (schema)
Passthru (server)
All (tables, views, nicknames)
Alter (tables, views, nicknames)
Index (tables, nicknames)
References (tables, nicknames)
Use (tablespaces)
Bind (packages)
Usage (sequences)
Control (sequences, nicknames, packages, procedures, functions, methods, tables, views, tablespaces)

Privileges to modify data only:
Delete (tables, views)
Insert (tables, views)
Update (tables, views)
Select (tables, views)
Execute (packages, procedures, functions, methods)

Privileges that create, modify, or delete database objects constitute a change to the database design and can effect operation of the database. To protect the integrity of the database, privileges that alter data structures will be restricted to DBAs and application object owners.

- *(N/A: CAT II) The DBA will ensure that assignment of the following object privileges are restricted to DBAs and application object owners:*

  Alterin
  Createin
  Dropin
  Passthru
  All
  Alter

  Index
  References
  Use
  Bind
  Usage
  Control

The IMPLICIT_SCHEMA database privilege allows users to create schemas implicitly by referencing an undefined schema. Whenever a schema is defined implicitly, PUBLIC is automatically granted the CREATEIN object privilege to the schema. The DBA will ensure that PUBLIC is not granted CREATEIN object privileges within any database.

- *(N/A: CAT II) The DBA will ensure that PUBLIC is not granted the CREATIN privilege in any database.*

The USE privilege to tablespaces is granted automatically to PUBLIC upon tablespace creation. This privilege will be revoked from PUBLIC in order to prevent usage of tablespace resources by unauthorized users.

- *(N/A: CAT II) The DBA will ensure that PUBLIC is not assigned the USE object privilege to any tablespace.*

When privileges are assigned with the CONTROL object privilege, several individual object privileges are granted with the WITH GRANT OPTION. The WITH GRANT OPTION allows the grantee to assign the granted privilege to other database users. Privilege assignment will be restricted to DBAs and application object owners. The CONTROL object privilege will not be granted to application user database accounts. Object privileges will not be granted to application users with the WITH GRANT OPTION.

- *(N/A: CAT II) The DBA will ensure that the CONTROL object privilege is restricted to DBAs and application object owner accounts.*

- *(N/A: CAT II) The DBA will ensure that application user database accounts are not assigned object privileges with the WITH GRANT OPTION.*

By default, PUBLIC is granted select privileges to 238 system catalog tables and views during a typical installation. These privileges should be reviewed to determine what is required by supported applications. Required permissions should be removed from PUBLIC and assigned to the appropriate application user role. At a minimum, access to the following system catalogs tables and views will be revoked from PUBLIC:

SYSCAT.DBAUTH
SYSCAT.TABAUTH
SYSCAT.PACKAGEAUTH
SYSCAT.INDEXAUTH
SYSCAT.COLAUTH
SYSCAT.PASSTHRUAUTH
SYSCAT.SCHEMAAUTH

- *(N/A: CAT II) The DBA will revoke access from PUBLIC to the system catalog tables and views listed above.*

D.5.4 DB2 Implicit Privileges

Creators of database objects, schemas, or tablespaces receive CONTROL privileges to the created object, schema, or tablespace. Control privileges include all object privileges available for that object type granted with the WITH GRANT OPTION. Creators of views only receive CONTROL privilege to the created view if they also have CONTROL on all the objects referenced in the view definition. Creation of views requires SELECT or CONTROL privileges on all the underlying tables or views.

By virtue of these implicitly assigned privileges, database object owner accounts are highly privileged accounts and will be locked when not in use for application update or maintenance.

- *(DO0160: CAT II) The DBA will ensure that the custom application object owner account is used only for update and maintenance of the application objects.*

D.6 Auditing in DB2

DB2 does provide an audit facility to record events as they occur within the database. Auditing is configured at the instance level meaning that all configured audit events are similarly recorded for actions in all databases belonging to the instance. Audit records are stored in the db2audit.log file located in the DB2 instance/security subdirectory where the audit configuration file, db2audit.cfg, is also stored. By default, access to the audit log files is restricted to Read and Write actions by the instance owner account. These permissions are assigned at audit log file creation time.

Access to the db2audit utility used to configure auditing is restricted to users with the SYSADM authority. Auditing is not started automatically at system or instance startup. It operates independently of the other DB2 processes. The *db2audit start* command must be entered manually or submitted by the system. DB2 auditing will be configured to start at system startup. The DBA will configure audit options as follows or more inclusive:

> Audit – required success and failure– audits audit configuration changes
> Checking – not required - audits authorization checking of attempts to access, create, alter, drop DB2 objects
> Objmaint – required success and failure– audits create, alter, or drop of objects
> Secmaint – required success and failure– audits privilege assignments and database configuration modifications
> Sysadm – required success and failure– audits SYSADM privileged activities
> Validate – required – audits authentication events
> Context – required failure only – provides

- *(N/A: CAT II) The DBA/SA will ensure that DB2 auditing is enabled at database server startup.*

- *(N/A: CAT II) The DBA/SA will ensure that access to the db2audit.log and db2audit.cfg files is restricted to the authorized users.*

- *(N/A: CAT II) The DBA will ensure that DB2 audit options are configured to the required settings or more inclusive.*

When set to the value of 0, the database manager parameter audit_buf_sz will cause audit records to be written at the time of generation. A value greater than 0 indicates the number of 4KB pages that will be used to create an internal buffer used to store audit records before writing a group of them to disk. Holding the audit records in a buffer before writing them to disk creates the potential of lost audit records during a system interruption – precisely the time when it is most advantageous to have them available. An audit_buf_sz of 0, however, can have a negative impact on system performance. It is recommended that the audit_buf_sz be set to 0.

- *(N/A: CAT II) The DBA will ensure that the audit-buf-sz parameter is set to 0 unless authorized by the IAO.*

The errortype database manager parameter may be set to the following values with the listed result:

> Audit = successful audit record generation included in determining success of the audited action
>
> Normal = any error from audit facility resulting from an audited event does not report an error in the audited event result

The setting of this parameter is left to the discretion of the application data owner. Auditing of access or changes to data must comply with privacy, security classification, and other sensitivity considerations. Required auditing may be performed by the database auditing facility or designed into the capabilities of the application used to access the data.

D.7 DB2 Configuration Parameters

DB2 configuration parameters are used to define or manage DB2 use of server resources as well as determine operation of the instance and databases. Parameter values cannot be modified directly. They must be defined or modified using the DB2 Control Center, the Command Line Processor (CLP) utility, Application Programming Interfaces (API's), used via custom programs to modify the parameter values, or using the DB2 Configuration Assistant.

D.7.1 Database Manager Configuration Parameters

The Database Manager configuration parameters configure a DB2 instance. These parameters primarily affect system resources allocated to the DB2 instance. The parameter values are stored in the db2systm file that is found in the /sqllib subdirectory for the instance in UNIX and in the \sqllib\<instance-name> directory in Windows where instance-name is the name of the instance. On either system, if the DB2INSTPROF environment variable is set, the database manager configuration file is found in the directory named by that environment variable. The following parameters affect the security posture of the DB2 instance. They must be defined as required below for each instance on the database server.

D.7.1.1 Audit_buf_sz – Audit Buffer Size

If set to 0, then audit records are written as soon as they are generated. Setting this value to other than 0 allows the audit records to be cached in a buffer to be written at a more optimized performance time. Setting the value to 0 potentially decreases database performance.

- *(N/A: CAT II) The DBA will ensure that the audit-buf-sz parameter is set to 0 unless authorized by the IAO.*

D.7.1.2 Smp_log_path – Sync Point Manager Log Path

The smp_log_path parameter is used to specify alternate locations for log files generated by the Sync Point Manager (SPM). Access to the SPM log file directories will be restricted to authorized users. The default spm_log_file directory is /sqllib/spmlog.

- *(N/A: CAT II) The DBA will ensure that access to the directory specified by the smp_log_path parameter is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.1.3 Datalinks – Data Links Support

Data Links provides access to external host files. If set to YES, then Data Links are supported on the database. If not required, Data Links will be disabled on the database.

- *(N/A: CAT II) The DBA will set the datalinks value to NO unless Data Links are required by a database application.*

D.7.1.4 Discover – Discovery Mode

This parameter is used to enable or disable the DB2 client's method for determining access to DB2 information on the local or remote instances/databases. Setting this value to ENABLE allows the client to send broadcast requests on the network for response by any available DB2 servers on the server that are configured to respond. A response to a discovery request contains information on all instances and databases on the server. This could lead to unauthorized access attempts to a remote instance or database. While less convenient, clients and other servers should be required to have connection information manually defined for them or use LDAP for retrieving database catalog information. The discover parameter value will be set to DISABLE.

- *(N/A: CAT II) The DBA will ensure that the discover parameter is set to DISABLE on the database instance.*

D.7.1.5 Discover_comm – Discover Communication Protocols

This parameter is used to define the communications protocols on which discovery searches are listened for and responded to. The discovery service should be disabled to protect the database server from unauthorized access attempts. The discover_comm parameter value will be set to DISABLE.

- *(N/A: CAT II) The DBA will ensure that the discover_comm parameter is set to DISABLE.*

D.7.1.6 Discover_inst – Discover Server Instance

This parameter is used to enable or disable the database server's response to discovery requests by this instance. A response to a discovery request contains information on all instances and databases on the server. This could lead to unauthorized access attempts to the instance and databases. While less convenient, clients and other servers should be required to have connection information manually defined for them or use LDAP for retrieving database catalog information. The discover_inst parameter value will be set to DISABLE.

- *(N/A: CAT II) The DBA will ensure that the discover_inst parameter is set to DISABLE.*

D.7.1.7 Diaglevel – Diagnostics Error Capture Level

The diaglevel parameter determines what diagnostic messages are written to the db2diag.log file. Aside from assisting in regular database server maintenance, these messages may indicate possible security breaches or unauthorized access attempts. The level may be set to the following values to capture the types of diagnostic messages indicated:

0 – No diagnostic messages are written
1 – Server errors only are written
2 – All errors are written
3 – All errors and warnings are written
4 – All errors, warnings, and informational messages are written

The minimum types of messages that should be written are errors and warnings. Informational messages may prove to require too much storage space and may not provide valuable information for discovery of security breaches. The diaglevel will be set to a minimum value of 3.

- *(N/A: CAT II) The DBA will ensure that the diaglevel parameter is set to a minimum value of 3.*

D.7.1.8 Diagpath – Diagnostic Data Directory Path

The diagpath parameter value defines the directory path where the db2diag file containing the diagnostic messages generated by the database server is written. The path should specify a protected directory on the host system. If no path is specified, the sqllib\<instance-name> is used for Windows and the sqllib/db2dump directory is used for UNIX. Access to the directory path specified in the diagpath parameter will be restricted to authorized users.

- *(N/A: CAT II) The DBA will ensure that access to the directory specified in the diagpath parameter is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.1.9 Notifylevel – Notify Level

The notifylevel parameter indicates the types of notify messages written to the notification file. On Windows, notification messages are written to the Event Logs. On UNIX, notifications are written to the instance.nfy text file. The levels and corresponding typs of messages captured are listed below:

0 – No notification messages
1 – Fatal or unrecoverable notification messages
2 – Immediate action required messages
3 – Important information, no action required.
4 – Informational

The setting specified indicates that all messages of a lower value will also be written to the notification file. Notification messages, in addition to providing database server status, may also indicate possible security breaches. The notifylevel will be set to a minimum of 3.

- *(N/A: CAT II) The DBA will ensure that the notifylevel is set to a minimum value of 3.*

D.7.1.10 Federated – Federated Database System Support

This parameter, when set to a value of YES, indicates that the instance supports requests to remote databases in a distributed environment. Access to remote databases initiated by the local database requires special security considerations. Please see the section on Federated Database Systems in this STIG for more specific information. If use of federated databases is not required, the federated parameter will be set to NO. If required, this requirement will be documented with the IAO.

- *(N/A: CAT II) The DBA will ensure that the federated parameter is set to a value of NO unless required and documented with the IAO.*

D.7.1.11 Sysadm_group – System Administration Authority Group Name

This parameter specifies the operating system group name assigned sysadm authority within the instance. By default, this group is set to NULL. On a Windows platform, a null value defaults to assignment of sysadm authority to the local administrator group. On UNIX platforms, a null value assigns sysadm authority to the primary group of the DB2 software owner/installer account. Separation of duties is promoted on a Windows platform by reassigning the sysadm group value to a custom created group. On a Windows platform the sysadm_group parameter will not be NULL. On a Windows platform, the sysadm_group parameter will be assigned to a custom group.

- *(N/A: CAT II) The DBA will assign the name of a custom local group to the sysadm_group parameter on a Windows platform.*

D.7.1.12 Sysctrl_group – System Control Authority Group Name

This parameter specifies the operating system group name assigned sysctrl authority within the instance. By default, this group is set to NULL. On a Windows platforms, a null value defaults to assignment of sysctrl authority to the local administrator group. On UNIX platforms, a null value assigns sysadm authority to the primary group of the DB2 software owner/installer account. Separation of duties is promoted on a Windows platform by reassigning the sysctrl group value to a custom created group. On a Windows platform the sysadm_group parameter will not be NULL. On a Windows platform, the sysctrl_group parameter will be assigned to a custom group.

- *(N/A: CAT II) The DBA will assign the name of a custom local group to the sysctrl_group parameter on a Windows platform.*

D.7.1.13 Sysmaint_group – System Maintenance Authority Group Name

This parameter specifies the operating system group name assigned sysmaint authority within the instance. By default, this group is set to NULL. On a Windows platforms, a null value defaults to assignment of sysmaint authority to the local administrator group. On UNIX platforms, a null value assigns sysmaint authority to the primary group of the DB2 software owner/installer account. Separation of duties is promoted on a Windows platform by reassigning the sysmaint group value to a custom created group. On a Windows platform the sysmaint_group parameter will not be NULL. On a Windows platform, the sysmaint_group parameter will be assigned to a custom group.

- *(N/A: CAT II) The DBA will assign the name of a custom local group to the sysmaint_group parameter on a Windows platform.*

D.7.1.14 Authentication – Authentication Type

The authentication type parameter defines the method of authentication used to connect users to the database. Available options are SERVER, SERVER_ENCRYPT, CLIENT, KERBEROS, and KRB_SERVER_ENCRYPT. For a more in-depth discussion on the requirements of this parameter, please see the section DB2 Authentication in this STIG appendix. To protect passwords from being sent in clear text within a database connection request, the authentication parameter will be set to SERVER_ENCRYPT, KERBEROS, or KRB_SERVER_ENCRYPT.

- *(N/A: CAT II) The DBA will ensure that the authentication parameter is set to a value of SERVER_ENCRYPT, KERBEROS, or KRB_SERVER_ENCRYPT.*

D.7.1.15 Use_sna_auth – Use SNA Authentication

This parameter allows users to connect to the database if they have already authenticated to SNA. This requires that they be using the SNA protocol to communicate to the database. Use of this authentication mode requires that the instance authentication mode be set to SERVER. Since an authentication mode of SERVER allows passwords to be transmitted across the network in clear text, the SNA authentication will not be used.

- *(N/A: CAT II) The DBA will ensure that the sna_auth parameter value is set to NO.*

D.7.1.16 Fed_noauth – Bypass Federated Authentication

The fed_noauth enables or disables the requirement to authenticate to the instance when the federated parameter is enabled and the authentication mode is either SERVER or SERVER_ENCRYPT. The fed_noauth will be set to NO.

- *(N/A: CAT II) The DBA will ensure that the fed_noauth parameter is set to NO.*

D.7.1.17 Catalog_noauth

The catalog_noauth parameter when set to YES or 1 allows users without SYSADM authority to catalog databases. Unauthorized changes to the database catalogs could result in errors in access to local and remote databases. The catalog_noauth parameter will be set to NO or 0.

- *(N/A: CAT II) The DBA will ensure that the catalog_noauth parameter is set to No or 0.*

D.7.1.18 Dftdbpath – Default Database Path

The value of dftdbpath parameter determines the default file storage location of newly created databases. Access to the path indicated by this parameter will be restricted to authorized users.

- *(N/A: CAT II) The DBA will ensure that access to the path indicated in the dftdbpath parameter is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

- *(N/A: CAT II) The DBA will ensure that access to the path indicated in the dftdbpath parameter is a valid path on the server operating system.*

D.7.1.19 Trust_allclnts – Trust All Clients

This parameter is used in conjunction with an authentication mode of CLIENT to allow or disallow all clients to be authenticated at the client operating system for DB2 access. This parameter is only effective when the authentication mode is set to CLIENT. Since the requirement for authentication mode is that it be set to a value other than CLIENT, the value of this parameter must be left to YES.

D.7.1.20 Trust_clntauth – Trusted Clients Authentication

This parameter specifies whether trusted clients are authenticated at the client or at the server. This parameter is not considered unless the authentication mode is CLIENT. If the client presents a username and password when a database connection is requested and the trust_clnauth parameter is set to SERVER, then the client will be authenticated by the server. Since the requirement for authentication mode is that it be set to a value other than SERVER, the value of this parameter must be left to CLIENT.

D.7.2 Database Configuration Parameters

Database configuration parameters configure a single DB2 database within an instance. Each database has its own database configuration file named SQLDBCON that stores the values assigned for each parameter. The SQLDBCON file is stored with other database control files in the SQLnnnn subdirectory where nnnn is the number assigned to the database at database creation time. This directory is found under the \DB2 directory in Windows and the /home/db2inst1/dbinst1 directory under UNIX. If the database is partitioned, a SQLDBCON file is created for each partition. Like the database manager configuration file, the SQLDBCON file cannot be edited directly. The database parameters listed below affect the security posture of the database. These database parameters must be defined as required for each database on the database server.

D.7.2.1 Logpath – Current Log Path

This parameter is not configurable, but is used to store the value for the current path of the log files. It can be changed only after the new value specified in the newlogpath parameter takes effect. By default, log files are stored in the SQLOGDIR under the database directory. In addition to log files being necessary to recover a database after an uncontrolled shutdown, for example, a disk or power failure, log files may include sensitive information. The directory specified in the logpath parameter will be a protected directory. For UNIX, access to the logpath directory will be restricted to 750 or more restrictive. For Windows, access to the logpath directory will be restricted to Administrators, SYSADMS, and the DB2 service account.

- *(N/A: CAT II) The DBA/SA will ensure that access to the DB2 logpath directory is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.2.2 Loghead – Current Log File

This parameter is also not configurable, but instead stores the name of the log file currently in use. Log files, in addition to being required to recovery a database after an uncontrolled shutdown, may contain sensitive information. Access to log files will be restricted to authorized users.

- *(N/A: CAT II) The DBA/SA will ensure that access to the DB2 file specified in the loghead database parameter is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.2.3 Newlogpath – Log File Path

The newlogpath parameter is used to store the path where the log files will be stored upon next database restart. Access to the directory specified in the newlogpath directory will be restricted to authorized users.

- *(N/A: CAT II) The DBA/SA will ensure that access to the DB2 newlogpath directory is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.2.4 Mirrorlogpath – Mirror Log File Path

The mirrorlogpath specifies a directory where a mirror copy of the log file is written. Specifying a directory on a separate physical disk ensures that recovery of the database to a point in time can be made after a disk failure that houses the logpath directory is lost. The mirrorlogpath parameter will specify a directory on a physical disk separate from the logpath directory unless the logpath directory is already housed on a mirrored or RAID 5 disk. The mirrorlogpath, if specified, will be restricted to authorized users.

- *(N/A: CAT II) The DBA/SA will ensure that access to the DB2 mirrorlogpath directory is restricted to authorized users.*

- *(N/A: CAT II) The DBA will ensure that the mirrorlogpath specifies a location on a separate physical disk unless the logpath specifies a directory on a mirrored or RAID 5 disk.*

D.7.2.5 Overflowlogpath – Overflow Log Path

The overflowlogpath parameter specifies a directory path for storage of log files used for a ROLLFORWARD database recovery or to store active log files retrieved from the archive logs. This parameter should specify a protected directory restricted to authorized users only.

- *(N/A: CAT II) The DBA/SA will ensure that access to the DB2 overflowlogpath directory is restricted to SAs, DBAs, the DB2 software installation account, and DB2 service/daemon accounts.*

D.7.2.6 Logretain – Log Retention Logging

The logretain parameter specifies whether log files are retained as archive log files to be used in a database recovery operation. The DBA will set this value to RECOVERY to enable log file retention unless the IAO approves recovery only to the time of the last full database backup. Retaining log files is appropriate for dynamic data that is modified on a frequent basis. For databases housing only static copies of data, this capability may be unnecessary and the parameter value may be set to NO.

Please also see the userexit parameter that has the same effect as the logretain parameter.

- *(N/A: CAT II) The DBA will ensure that the logretain parameter is set to RECOVERY unless authorized by the IAO.*

D.7.2.7 Userexit – User Exit Enable

The userexit parameter is used to specify whether log retention is enabled. The value may be set to YES or NO. When set to YES, regardless of the logretain setting (see above), log retention is enabled. The userexit parameter will be set to YES unless the IAO approves no log retention.

- *(N/A: CAT II) The DBA will ensure that userexit is set to YES unless authorized by the IAO.*

D.7.2.8 Tsm_password – Tivoli Storage Manager Password

This parameter is used to store the password for the Tivoli Storage Manager (TSM). The password is displayed in clear text upon entry and is stored encrypted in the configuration file. There are no password management features such as complexity or length requirements for the password. The DBA will ensure that the value entered for tsm_password follows all possible DOD password requirements.

- *(N/A: CAT II) The DBA will ensure that the tsm_password follows all password requirements as listed in Section 3.2.1, Password Guidelines when TSM is in use.*

D.7.2.9 Discover_db – Discover Database

This parameter is used to enable or disable the database's response to Discovery requests. A response to a discovery request contains information on all instances and databases on the server. This could lead to unauthorized access attempts to the instance and databases. While less convenient, clients and other servers should be required to have connection information manually defined for them or use LDAP for retrieving database catalog information. The discover_db parameter value will be set to DISABLE.

- *(N/A: CAT II) The DBA will ensure that the discover_db parameter is set to DISABLE.*

D.7.3 DB2 Administration Server Configuration Parameters

The DB2 Administration Server (DAS) provides administrative access to the DB2 instances and databases on the local server. Only one DAS is defined per server. The configuration parameters below provide enhanced security to the DB2 instances and databases accessed by the DAS.

D.7.3.1 Discover – DAS Discovery mode

The discover parameter value determines the how the DAS responds to discovery requests issued by DB2 clients. When set to SEARCH, the DAS responds to SEARCH and KNOWN requests issued by DB2 clients by providing all information about instances and databases available on its local database server. When set to KNOWN, the DAS provides the local instance and databases only when a KNOWN discovery request is received from a client. A KNOWN request indicates that the client was configured with connection information for a specific DB2 instance and/or databases. When set to DISABLE, the DAS ignores discovery requests. The DISABLE setting helpst to protect the local instances and databases from unauthorized access attempts. Discovery is only available when the DAS is configured for TCP/IP communication.

- *(N/A: CAT II) The DBA will ensure that the DAS discover parameter is set to DISABLE when the DAS is configured for TCP/IP communications.*

D.7.3.2 Dasadm_group – DAS Administration Group Name

This parameter defines the database server OS group name granted DAS administrative authority (DASADM). When set to NULL on a Windows server, the local Administrators group is used. When set to NULL on a UNIX server, the primary group assigned to the DB2 instance account is used. To maintain separation of duties, a custom group will be assigned to the dasadm_group parameter

- *(N/A: CAT II) The DBA will ensure that the dasadm_group is set to a custom account on a Windows server.*

D.7.3.3 Exec_exp_task – Execute Expired Tasks

The exec_exp_task parameter defines whether the DAS executes upon startup any tasks found in the task scheduler that have passed their scheduled start time. To protect against unauthorized tasks being run upon startup of the DAS, this parameter will be set to NO. This setting requires that the task schedule be reviewed upon startup and authorized tasks manually started after a DAS restart.

- *(N/A: CAT II) The DBA will ensure that the exec_exp_task parameter is set to NO.*

D.7.3.4 Sched_userid - Scheduler User ID

The sched_userid names the userid used to connect to by the DAS to connect to a remote tools database that stores the DAS schedule data. This parameter is only referenced if the tools database is remote to the DAS server. If this parameter is required, access to the userid specified should be restricted to the DAS server. The userid specified in the sched_userid parameter will be restricted to authorized DAS use.

- *(N/A: CAT II) The DBA will ensure that the userid specified by the sched_userid parameter is restricted to authorized DAS use.*

D.7.3.5 Authentication – Authentication type DAS

Like the database manager authentication mode, the DAS authentication parameter specifies the accepted authentication modes for accessing the DB2 database. Both available authentication modes for the DAS, SERVER_ENCRYPT and KERBEROS_ENCRYPT, encrypt passwords transmitted over the network.

- *(N/A: CAT II) The DBA will ensure that DAS authentication is set to SERVER_ENCRYPT or KERBEROS_ENCRYPT.*

D.8 Network Security

DB2 UDB Version 8.1 offers no network communication encryption services. This function will be available in the "Stinger" version of DB2 that has not yet been released for production as of the date of this STIG. Database communications requiring encryption must be configured to use VPN or some other method available via the network or database server operating system.

D.9 DB2 Operating System Security Requirements

In addition to instance and database privileges granted by means of OS group definition and membership grants as described earlier in the appendix, OS accounts and groups are used as means to provide access to OS resources, namely directory and file access.

During installation, DB2 creates the following default account names and groups. Custom accounts and group names may be substituted, but the functions of each remain the same. UNIX groups defined and used to assign access to directories, files, and DB2 privileges include the following listed by default names:

Account Name	Description
Db2inst1	Instance owner/administrator
Db2fenc1	Fenced user
Dasusr	DAS owner/administrator

Table 16. DB2 UDB OS ACCOUNTS

Group Name	Description
Db2iadm1	Instance owner/administrator group
Db2fadm1	Fenced user group
Db2asgrp	DAS owner/administrator group

Table 17. DB2 UDB OS GROUPS

Upon installation, the following OS groups are defined: dasadm1 for DAS administration, db2iadm1 for theDB2 installation account for DB2 software maintenance, dbfgrp1 to manage privileges for the db2fenc1 user for the processing of DB2 external routine calls, and the db2grp1 account to support privileges assigned to the db2inst1 account.

D.9.1 UNIX

Directories created by the DB2 installation are:

/home/db2inst1
/home/dasusr1
/home/db2fenc1
/var/db2
/opt/IBM
other data storage directories as specified in the DB2 database

All files and subdirectories located in the directories listed above will have world or others permissions removed. All files and subdirectories located in instance owner account home directory will have their ownership set to the instance owner account. All files and subdirectories located in the DAS OS account home directory will have their ownership set to the DAS OS account. All files and subdirectories located in the DB2 fenced user OS account home directory will have their ownership set to the DB2 fenced user OS account. These restrictions will prevent DB2 daemons being run under the Root account and protect the files and directories by requiring discrete (group membership required) assignment of privileges for access. DB2 executable files will not have the SUID or GUID.

This can be accomplished by setting the current directory to the directory to be configured and using the following commands:

```
Chown –R account-name *
Chmod –R o-rwx *
```

- *(N/A: CAT II) The DBA/SA will set DB2 file and directory ownership to the DB2 instance owner, DB2 fenced user, and DAS account as appropriate.*

- *(N/A: CAT II) The DBA/SA will revoke all world privileges from DB2 files and directories.*

- *(N/A: CAT II) The DBA/SA will ensure that no DB2 executable files have the SUID or GUID bit set.*

DB2 authorities and DAS administrative authority are by default granted to the UNIX db2iadm1 group. If the SYSADM, SYSCTRL, SYSMAINT, and DASADM authorities require separation for your site, custom groups must be created and membership assigned appropriately to accommodate this.

D.9.2 Windows

Database resources including directories, files, and registry entries are protected by the Windows operating system. Default directories created by the DB2 installation are:

Disk:\Program Files\
Disk:\DB2

- *The SA/DBA will ensure that access to the DB2 directories listed above is limited to Full Control permissions granted to Administrators and the DB2 software installation account, and Modify, Read & Execute, List Folder Contents, Read, and Write permissions granted to DB2 service accounts.*

Registry keys and values are found under:
HKLM\Software\IBM\DB2
HKLM\System\CurrentControlSet\Services (services beginning with DB2)

- *(N/A: CAT II) The SA/DBA will ensure that access to the DB2 registry keys and values located under the registry hives listed above is limited to Read and Full Control permissions granted to Administrators, the DB2 software installation account, and DB2 service accounts.*

DB2 authorities and DAS administrative authority are by default granted to the Windows Administrators group. If the SYSADM, SYSCTRL, SYSMAINT, and DASADM authorities require separation for your site, custom groups must be created and membership assigned appropriately to accommodate this.

D.10 DB2 Replication

Replication supports multiple copies of database objects on remote databases. This requires that databases participating in replication authenticate to remote databases to receive or share copies of database objects with the remote databases. These activities are accomplished using standard database client to server connections using the authentication mode defined for the database. A single OS account will be used to support replication connections.

- *(N/A: CAT II) The DBA will ensure that a single OS account is used to authenticate to databases to support replication activities.*

- *(N/A: CAT II) The DBA will ensure that the minimum DB2 privileges are assigned to the replication account on the database server to support the replication activities on that database.*

- *(N/A: CAT II) The DBA will ensure that DASADM and SYSADM authorities are not granted to replication OS accounts.*

This page is intentionally left blank.

APPENDIX E ORACLE DATABASE STIG COMPLIANCE CONFIGURATION

NOTE: This list of database security compliance configuration instructions is not all-inclusive and will not complete the secure configuration of your Oracle database. Some compliance checks such as backup and recover procedures or encrypting sensitive data require steps specific to your environment.

E.1 Update Version/Apply Security Patches

To check the database version:

From SQLPLUS:

```
Select * from v$version;
```

Current Versions are as follows: 9i: 9.2.0, 10.1.0
8i: 8.1.7.4

See http://otn.oracle.com/deploy/security/alerts.htm for a list of current security alerts and patches required to resolve associated vulnerabilities.

NOTE:

VERSION	***TERMINAL RELEASE?***	***DESUPPORT DATE***	***CURRENT/ LAST PATCH***
10.1.0 (Oracle 10g)	No	TBD	NA
9.2.0 (Oracle 9i, Release 2)	no	31 December 2005	9.2.0.5 / NA
9.0.1 (Oracle 9i)	no	31 December 2003	9.0.1.4 / 9.0.1.4
8.1.7 (Oracle 8i, Release 3)	yes	31 December 2003	8.1.7.4 / 8.1.7.4
8.1.6 (Oracle 8i, Release 2)	yes	31 October 2001	8.1.6.3
8.1.5	yes	01 January 2001	8.1.5.1
8.0.6 (Oracle 8)	yes	30 September 2001	8.0.6.3
7.3.4	yes	31 December 2000	7.3.4.5

Table 18. ORACLE VERSION SUPPORT STATUS

Install available patches according to patch instructions.

- To view alerts reported by Oracle, go to the Oracle Technology Network (OTN) security alert page at http://otn.oracle.com/deploy/security/alerts.htm, or log into Metalink.oracle.com, select Product Lifecycle, select Alerts, select Oracle Server – Enterprise Edition, or whichever product you wish to check, open the link to the Support Status and Alerts for your database version. Review the list for security alerts. Apply patches as appropriate.

- Check for Oracle database IAVM notices on http://www.cert.mil.

E.2 Set Initialization Parameters

NOTE: The name and location of the pfile directory is version and operating system specific.

Using any text editor, edit the $ORACLE_BASE\admin\<*db_name*>/INIT<*sid*>.ORA file.

Add or edit to include the following lines:

PARAMETER NAME	***VALUE***
*DBLINK_ENCRYPT_LOGIN	TRUE
AUDIT_TRAIL	TRUE, OS, DB
RESOURCE_LIMIT	TRUE
REMOTE_OS_AUTHENT	FALSE
REMOTE_OS_ROLES	FALSE
OS_ROLES	FALSE
UTL_FILE_DIR	*Valid, Protected directory*
SQL92_SECURITY	TRUE
O7_DICTIONARY_ACCESSIBILITY	FALSE
REMOTE_LOGIN_PASSWORDFILE	EXCLUSIVE or NONE
**AUDIT_SYS_OPERATIONS	TRUE
GLOBAL_NAMES	TRUE
_TRACE_FILES_PUBLIC	FALSE
****MAX_ENABLED_ROLES	30
****REMOTE_LISTENER	NULL
***AUDIT_FILE_DEST	Valid, Protected Directory
USER_DUMP_DEST	Valid, Protected Directory
BACKGROUND_DUMP_DEST	Valid, Protected Directory
CORE_DUMP_DEST	Valid, Protected Directory
****LOG_ARCHIVE_START	TRUE
LOG_ARCHIVE_DEST	Valid, Protected Directory
LOG_ARCHIVE_DUPLEX_DEST(_n)	Valid, Protected Directory
OS_AUTHENT_PREFIX	Not OPS$

Table 19. ORACLE REQUIRED INITIALIZATIOIN PARAMETER SETTINGS

*	Applies for version 9.0.1 and earlier.
**	Applies to version 9.2 and later.
***	Applies to UNIX versions only.
****	Not Required.

Use OS file permissions to protect the directories. Permissions require that the Oracle process have permissions or privileges to write to files and change permissions on files in the directory. Only System Administrators and DBAs should have full permissions to the directory. Please note that all database accounts have *read* and *write* access from within the database to all files in the UTL_FILE_DIR directory by virtue of the Oracle process OS permissions.

NOTE: Prior to starting the database with LOG_ARCHIVE_START=TRUE, the database should be set to ARCHIVELOG enabled. To do this the database needs to be in EXCLUSIVE mode. Shutdown the database, issue a START MOUNT EXCLUSIVE command at the SQL*Plus prompt, and enter the ALTER DATABASE ARCHIVELOG; command.

Stop and restart the database after these parameters have been set.

NOTE: One parameter must be set on Oracle client workstations to ensure encryption of passwords transmitted across the network during database logon. The ORA_ENCRYPT_LOGIN must be set to TRUE on the workstation. On a Windows 2000 system, do the following:

From the Windows desktop:

Right-click on My Computer.
Select Properties.
Click on Environment Variables button.
Under System Variables, click NEW.
In Variable Name text box enter: ORA_ENCRYPT_LOGIN.
In Variable Value text box enter: TRUE.
Click OK.

E.3 Verify Domain Name Prefix in Use (Windows only)

Use REGEDT32 to verify the value for the key that specified use of the Windows domain to identify OS-authenticated database accounts is set to TRUE.

From REGEDT32:

Select HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\homeid\.
Double click on key OSAUTH_PREFIX_DOMAIN and enter TRUE in String text Box.
Click OK to save.

E.4 Verify SYSDBA and SYSOPER User Authorization in ORAPWD

If a REMOTE_LOGIN_PASSWORDFILE is in use (=EXCLUSIVE), then list database accounts assigned SYSDBA and SYSOPER database privileges and review for appropriate authorization as follows:

From SQLPLUS:

```
Select * from V$PWFILE_USERS;
```

To revoke SYSDBA or SYSOPER from unauthorized database accounts:

From SQLPLUS:

```
Revoke sysdba from user;
Revoke sysoper from user;
```

If a REMOTE_LOGIN_PASSWORDFILE is NOT in use, the SYSDBA and SYSOPER are being authorized by virtue of membership to an OS Group. Consult the installation guide for your version and platform to determine the name of the OS Groups. Review memberships for authorized OS accounts.

E.5 Verify Status of Default Accounts

Review the status of Oracle default accounts. Only accounts required for daily operation should be Open. All other accounts should be removed if possible or locked and/or expired. Following is a list of Oracle recommended status for default accounts.

From SQLPLUS, enter:

```
Select username from dba_users where account_status<>’OPEN’ order by username;
```

To lock accounts:

```
Alter user username account lock;
```

To expire accounts:

```
Alter user username password expire;
```

USERNAME	ACCOUNT STATUS		
ADAMS	EXPIRED	&	LOCKED
AURORA$JIS$UTILITY$	OPEN		
AURORA$ORB$UNAUTHENTICATED	OPEN		
BLAKE	EXPIRED	&	LOCKED
CLARK	EXPIRED	&	LOCKED
CTXSYS	EXPIRED	&	LOCKED
DBSNMP	OPEN		

USERNAME	ACCOUNT STATUS		
HR	EXPIRED	&	LOCKED
JONES	EXPIRED	&	LOCKED
LBACSYS	EXPIRED	&	LOCKED
MDSYS	EXPIRED	&	LOCKED
OE	EXPIRED	&	LOCKED
OLAPDBA	EXPIRED	&	LOCKED
OLAPSVR	EXPIRED	&	LOCKED
OLAPSYS	EXPIRED	&	LOCKED
ORDPLUGINS	EXPIRED	&	LOCKED
ORDSYS	EXPIRED	&	LOCKED
OSE$HTTP$ADMIN	OPEN		
OUTLN	OPEN		
PM	EXPIRED	&	LOCKED
QS	EXPIRED	&	LOCKED
QS_ADM	EXPIRED	&	LOCKED
QS_CB	EXPIRED	&	LOCKED
QS_CBADM	EXPIRED	&	LOCKED
QS_CS	EXPIRED	&	LOCKED
QS_ES	EXPIRED	&	LOCKED
QS_OS	EXPIRED	&	LOCKED
QS_WS	EXPIRED	&	LOCKED
SCOTT	OPEN		
SH	EXPIRED	&	LOCKED
SYS	OPEN		
SYSTEM	OPEN		

Table 20. ORACLE DEFAULT ACCOUNT STATUS

E.6 Change Default Passwords

Change any and all default passwords. Following is a list of many default accounts created during the installation of an Oracle database. If the following default accounts or any others exist in your database, change their passwords immediately:

USERNAME	DEFAULT PASSWORD
!DEMO_USER	!DEMO_USER
ADAMS	WOOD
ADLDEMO	ADLDEMO
ADMIN	JETSPEED
APPLSYS	APPLSYS
APPLSYSPUB	PUB
APPS	APPS
AQ	AQ
AQDEMO	AQDEMO
AQJAVA	AQJAVA
AQUSER	AQUSER
AUDIOUSER	AUDIOUSER
AURORA$JIS$UTILITY$	invalid
AURORA$ORB$UNAUTHENTICATED	invalid
BC4J	BC4J
BLAKE	PAPER
CATALOG	CATALOG
CDEMO82	CDEMO82
CDEMOCOR	CDEMOCOR
CDEMORID	CDEMORID
CDEMOUCB	CDEMOUCB
CENTRA	CENTRA
CIDS	CIDS
CISINFO	ZWERG
CLARK	CLOTH
COMPANY	COMPANY
COMPIERE	COMPIERE
CQSCHEMAUSER	PASSWORD
CSMIG	CSMIG
CTXSYS	CTXSYS
DBI	MUMBLEFRATZ
DBSNMP	DBSNMP
DEMO	DEMO
DEMO8	DEMO8

USERNAME	DEFAULT PASSWORD
DES	DES
EJSADMIN	EJSADMIN_PASSWORD
EMP	EMP
ESTOREUSER	ESTORE
EVENT	EVENT
FINANCE	FINANCE
FND	FND
FROSTY	SNOWMAN
GPFD	GPFD
GPLD	GPLD
HCPARK	HCPARK
HLW	HLW
HR	HR
IMAGEUSER	IMAGEUSER
IMEDIA	IMEDIA
INTERNAL	ORACLE
JMUSER	JMUSER
JONES	STEEL
LBACSYS	LBACSYS
LIBRARIAN	SHELVES
MASTER	PASSWORD
MDSYS	MDSYS
MFG	MFG
MIGRATE	MIGRATE
MILLER	MILLER
MMO2	MMO2
MODTEST	YES
MOREAU	MOREAU
MTS_USER	MTS_PASSWORD
MTSSYS	MTSSYS
MXAGENT	MXAGENT
NAMES	NAMES
OAS_PUBLIC	OAS_PUBLIC
OCITEST	OCITEST
ODM	ODM
ODM_MTR	ODM_MTR
ODS	ODS
ODSCOMMON	ODSCOMMON
OE	OE

USERNAME	DEFAULT PASSWORD
OEMADM	OEMADM
OLAPDBA	OLAPDBA
OLAPSVR	OLAPSVR
OLAPSYS	OLAPSYS
OPENSPIRIT	OPENSPIRIT
ORACACHE	ORACACHE
ORAREGSYS	ORAREGSYS
ORDPLUGINS	ORDPLUGINS
ORDSYS	ORDSYS
OSE$HTTP$ADMIN	invalid
OSP22	OSP22
OUTLN	OUTLN
OWA	OWA
OWA_PUBLIC	OWA_PUBLIC
PANAMA	PANAMA
PATROL	PATROL
PERSTAT	PERSTAT
PLSQL	SUPERSECRET
PM	PM
PO	PO
PO7	PO7
PO8	PO8
PORTAL30_DEMO	PORTAL30_DEMO
PORTAL30_PUBLIC	PORTAL30_PUBLIC
PORTAL30_SSO	PORTAL30_SSO
PORTAL30_SSO_PS	PORTAL30_SSO_PS
PORTAL30_SSO_PUBLIC	PORTAL30_SSO_PUBLIC
POWERCARTUSER	POWERCARTUSER
PRIMARY	PRIMARY
PUBSUB	PUBSUB
QS	QS
QS_ADM	QS_ADM
QS_CB	QS_CB
QS_CBADM	QS_CBADM
QS_CS	QS_CS
QS_ES	QS_ES
QS_OS	QS_OS
QS_WS	QS_WS
RE	RE

USERNAME	DEFAULT PASSWORD
REP_MANAGER	DEMO
REP_OWNER	DEMO
REP_OWNER	REP_OWNER
REPADMIN	REPADMIN
RMAIL	RMAIL
RMAN	RMAN
SAMPLE	SAMPLE
SAP	SAPR3
SCOTT	TIGER
SDOS_ICSAP	SDOS_ICSAP
SECDEMO	SECDEMO
SH	SH
SITEMINDER	SITEMINDER
SLIDE	SLIDEPW
STARTER	STARTER
STRAT_USER	STRAT_PASSWD
SYMPA	SYMPA
SYS	CHANGE_ON_INSTALL
SYSADM	SYSADM
SYSMAN	OEM_TEMP
SYSTEM	MANAGER
TAHITI	TAHITI
TDOS_ICSAP	TDOS_ICSAP
TESTPILOT	TESTPILOT
TRAVEL	TRAVEL
TSDEV	TSDEV
TSUSER	TSUSER
TURBINE	TURBINE
ULTIMATE	ULTIMATE
USER	USER
USER0	USER0
USER1	USER1
USER2	USER2
USER3	USER3
USER4	USER4
USER5	USER5
USER6	USER6
USER8	USER8
USER9	USER9

USERNAME	DEFAULT PASSWORD
UTLBSTATU	UTLESTAT
VIDEOUSER	VIDEOUSER
VIF_DEVELOPER	VIF_DEV_PWD
VIRUSER	VIRUSER
VRR1	VRR1
WEBCAL01	WEBCAL01
WEBDB	WEBDB
WEBREAD	WEBREAD
WKSYS	WKSYS
WWW	WWW
WWWUSER	WWWUSER
XPRT	XPRT

Table 21. ORACLE ACCOUNT DEFAULT PASSWORDS

From DBA Studio or Security Manager:

Select the username and set the password in the text boxes for that purpose.

From SQLPLUS:

Alter user *username* identified by *new password*.

The SYS password may also be set with the ORAPWD utility. First, shut down the database! The default location for the password file is in the $ORACLE_HOME\database\ directory and is named pwd*sid*.ora. Using the ORAPWD utility will lose any existing entries. The ORAPWD will not overwrite an existing password file of the same name.

From the operating system prompt, type:

ORAPWD file=$ORACLE_HOME\database\pwd*sid*.ora password=*password*

E.7 Remove Demo Applications and Demo Database Accounts

Demo applications should be removed from the database. The default demo accounts provided during installation of the database include SQL scripts to remove them. Following is the list of these demo applications and the scripts to remove them. The scripts may be found in the $ORACLE_HOME/demo/schema/*demo_name* directory.

Human_resources	hr_drop.sql
Order_entry	oe_drop.sql, oc_drop.sql,
Product_media	pm_drop.sql

Sales_history	sh_drop.sql
Shipping	qs_drop.sql

Other demo applications that may be installed after database installation may be found in other $ORACLE_HOME directories. Similarly, they may also be provided with scripts to remove them. If not, then they may effectively be removed with the SQL command: drop user *username* cascade; where the *username* is the name of the account that owns the demo's objects. The default user SCOTT must be removed in this way.

E.8 Set a Listener Password

The Listener password must be set on all listeners running on the system. The password may be set either by editing it directly into the listener.ora file or by command in the LSNRCTL utility. Using the LSNRCTL enters the password in encrypted format into the listener.ora file. Because this file does contain the password, permissions to this file should be restricted and verified.

From the OS command prompt, type LSNRCTL and press Enter.

Status local (N/Ame of the listener)

If Security is On, then a password had been set. Type Exit.

```
LSNRCTL> set password
Password: (blank since there is initially no password assigned)
The command completed successfully
LSNRCTL> change_password
Old password: (blank)
New password: xxxxxx
Reenter new password: xxxxxx
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC0)))
Password changed for LISTENER
The command completed successfully
LSNRCTL> set password (you must do a set password here)
Password:
The command completed successfully
LSNRCTL> save_config (required to save the password ! !)
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC0)))
Saved LISTENER configuration parameters.
Listener Parameter File F:\oracle\ora81\network\admin\listener.ora
Old Parameter File F:\oracle\ora81\network\admin\listener.bak
The command completed successfully
LSNRCTL> exit
```

OR

NOTE: The following method stores the Listener password unencrypted in the listener.ora file.

Edit listener.ora file.
Add the following entry: PASSWORDS_*listener_name*=*password*
Stop and Restart listener from LSNRCTL utility.

E.9 Set Listener ADMIN_RESTRICTIONS On

Prevent remote administration of the listener by enabling ADMIN_RESTRICTIONS in the listener.ora file. Edit the file to include the following line:

ADMIN_RESTRICTIONS_*listener_name* = true

Replace *listener_name* with the name of your listener.

E.10 Create a Password Verification Function

NOTE: The password verification function must be owned by SYS!

Create the password verify function (available from the IASE web site and printed below):

From SQLPLUS connected as SYS or as SYSDBA:
@*full path*\utlpwdmgdisa.sql

Assign the password verify function to the all profiles:

From SQLPLUS:
Alter profile DEFAULT limit password_verify_function verify_password;

For a list of existing profiles:

From SQLPLUS:
Select distinct profile from dba_profiles;

PASSWORD VERIFY FUNCTION

```
Rem This script was modified from the Oracle utlpwdmg.sql
devault script.
Rem
-- This script sets the default password resource parameters
-- This script needs to be run to enable the password features.
-- However the default resource parameters can be changed based
-- on the need.
-- A default password complexity function is also provided.
-- This function makes the minimum complexity checks like
-- the minimum length of the password, password not same as the
-- username, etc. The user may enhance this function according
to
-- the need.
-- This function must be created in SYS schema.
-- connect sys/<password> as sysdba before running the script

CREATE OR REPLACE FUNCTION verify_function_disa
(username varchar2,
  password varchar2,
  old_password varchar2)
  RETURN boolean IS
   n boolean;
   m integer;
   differ integer;
   isdigit boolean;
   ispunct boolean;
   ischar boolean;

BEGIN

   -- Check if the password is same as the username
   IF NLS_LOWER(password) = NLS_LOWER(username) THEN
     raise_application_error(-20001, 'Password same as or
similar to user');
   END IF;

   -- Check for the minimum length of the password
   IF length(password) < 8 THEN
      raise_application_error(-20002, 'Password length less than
8');
   END IF;

   -- Check if the password is too simple. A dictionary of words
may be
   -- maintained and a check may be made so as not to allow the
words
   -- that are too simple for the password.
   IF NLS_LOWER(password) IN ('database', 'password',
'computer', 'abcdefgh') THEN
      raise_application_error(-20002, 'Password too simple');
   END IF;

   -- Check if the password contains at least one letter, one
digit and one
   -- punctuation mark.

   m := length(password);

   FOR i IN 1..m LOOP
```

```
     -- Check for digit, character, punctuation
     IF substr(password,i,1) BETWEEN '0'AND '9' THEN
       isdigit:=true;
     ELSIF substr(password,i,1) BETWEEN 'A' AND 'z' THEN
       ischar:=true;
     ELSIF substr(password,i,1) IN
('!','"','#','$','%','&','(',')','`','*','+','-
','/',':',';','<','=','>','?','_') THEN
       ispunct:=true;
     END IF;

     -- Exit loop if password contains a character, digit and
punctuation
     IF isdigit AND ischar AND ispunct THEN
       GOTO repeats;
     END IF;

   END LOOP;

   raise_application_error(-20003, 'Password should contain at
least one digit, one character and one punctuation');


   -- Check if the password contains repeating characters

   <<repeats>>
   FOR i IN 1..m-1 LOOP
     IF substr(password,i,1) = substr(password,i+1,1) THEN
       raise_application_error(-20003, 'Password may not contain
repeating characters');
       GOTO endsearch;
     END IF;
   END LOOP;


   -- Check if the password differs from the previous password
by at least 3 letters

   <<endsearch>>
   IF old_password IS NOT NULL THEN
   differ := length(old_password) - length(password);

     IF abs(differ) < 4 THEN
       IF length(password) < length(old_password) THEN
         m := length(password);
       ELSE
         m := length(old_password);
       END IF;

       differ := abs(differ);
       FOR i IN 1..m LOOP
         IF substr(password,i,1) != substr(old_password,i,1)
THEN
           differ := differ + 1;
         END IF;
       END LOOP;

       IF differ < 4 THEN
         raise_application_error(-20004, 'Password should differ
by at least 3 characters');
       END IF;
     END IF;

   END IF;


   -- Everything is fine; return TRUE ;
   RETURN(TRUE);
END;
/
ALTER PROFILE DEFAULT LIMIT
PASSWORD_LIFE_TIME 90
PASSWORD_GRACE_TIME 10
PASSWORD_REUSE_TIME UNLIMITED
PASSWORD_REUSE_MAX 10
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 1/24
IDLE_TIME 15
PASSWORD_VERIFY_FUNCTION verify_function_disa;
```

E.11 Set the Default Profile Security Parameters

The user profiles are used to restrict system resource uses as well as define some security parameters. The DEFAULT profile is used when no other profile is specified for the database account. The Default profile should be modified to secure database accounts that are not assigned a specific profile. Any custom profiles created in the database should also have the following the security parameters set

Using OEM/DBA Studio, select security, select profiles, and select Default. Under the general tab, enter/verify the following settings:

```
idle_time  15
password_life_time  90
password_reuse_max  10
password_reuse_time 365
failed_login_attempts  3
```

NOTE: The password_reuse_max and password_reuse_time may only both be set in versions 9i and later. For earlier versions, set one to the required setting and the other to UNLIMITED.

From SQLPLUS:

```
ALTER PROFILE DEFAULT LIMIT
PASSWORD_LIFE_TIME 90
PASSWORD_REUSE_TIME 365
PASSWORD_REUSE_MAX 10
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 1/24
IDLE_TIME 15
PASSWORD_VERIFY_FUNCTION verify_function_disa;
```

Repeat the above steps replacing *DEFAULT* with the target profile name for any other profiles in use within the database.

NOTE: DOD policy requires a password reuse limit of one year. Oracle versions earlier than 10g do not support the setting of both the password history count limit and the password history time limit simultaneously. Set one or the other according to your site's requirements.

Consider defining security policy for other account profile parameters such as sessions per user. These parameters help further reduce the potential for database account compromise.

E.12 Set Time Restrictions for Application/Batch Processing Accounts

One method to accomplish database account time restrictions from within the database is to lock and unlock the target account at specified intervals using the Oracle job queue. To restrict account access this way, follow the procedures below:

1. Enable the database job queue by setting job_queue_processes in the init.ora file to a value greater than 0.
2. Create a procedure to enable/disable the target database account:
   ```
   Create or replace procedure restrict_db_user
   (usrname varchar2, onoff varchar2) as
   begin
   execute immediate ‘alter user’||usrname||’ account ‘||onoff;
   end;
   ```
3. Submit the job at 5PM to enable the account at midnight each night:
   ```
   Variable :jobno;
   Begin
     Dbms_job.submit
       (:jobno,
        ‘restrict_db_user(‘’joe’’,’’unlock’’);’,
        sysdate+(7/24),
        ‘sysdate + 1+(7/24)’);
   end;
   /
   print jobno
   ```
4. Submit the job at 5PM to disable the account at 1:00 am each night:
   ```
   Variable :jobno;
   Begin
     Dbms_job.submit
       (:jobno,
        ‘restrict_db_user(‘’joe’’,’’lock’’);’,
        sysdate + (8/24),
        ‘sysdate +1+(8/24)’);
   end;
   /
   print jobno
   ```
5. To view the jobs in the queue:
   ```
   Select job,next_date,next_sec from dba_jobs;
   ```

E.13 Oracle File and Director Ownership

All files and directories installed by Oracle should be owned by the installation account and, for UNIX, group. On UNIX this is typically user Oracle and group OINSTALL although it is recommended by security experts that a different user name be used. For Windows, Oracle files and directories are typically installed and owned by the BUILTIN\Administrators group.

For UNIX, use chmod OWNER:GROUP FILENAME. Example: chmod oracle:oinstall *

For Windows, use the Windows Explorer. Right-click on the file name, select Properties, select the Security tab, select the Advanced button under the Permissions section, select the Owner tab. Files and directories should be owned by the BUILTIN\Administrators group.

E.14 Set Operating System (OS) Permissions

OS file permissions are granted to individual OS accounts or OS groups. In the case of an Oracle DBMS, the OS accounts we are interested in are the Oracle installation account, the System Administrators, the DBAs, and the application user OS accounts. The Oracle UNIX Inventory Group or ORAINVENTORY, typically named oinstall, is the owner of the Oracle Universal Installer directory. Any accounts used to install Oracle software must be a member of this group. If system backups are performed by separately defined OS accounts, then these accounts are also included. If the application user OS accounts will not be running the Oracle application software from the server, then they require no access rights of their own to any Oracle directories or files.

On Windows systems, Oracle is installed using the local administrator account. This account is a member of the local BUILTIN\Administrators group on the server. The BUILTIN\Administrators group should be listed as the owner of all Oracle software files and directories. A custom, dedicated, and restricted local administrator account should be used as the service account for all Oracle services on the host. If necessary, the account may be a Windows domain account. Under no circumstances should the account be a domain administrator account.

No OS accounts outside of these groups should have any permissions to any directories or files related to the Oracle database.

E.14.1 UNIX File Permissions

The following settings are from the Oracle *9i Installation Guide Release 2 for UNIX Systems*. Please consult the installation guide for your version for correct settings. These permissions are configured by default during installation. On a UNIX system, the umask must be set to 022 for the Oracle installation account during the Oracle software installation.

UNIX ACCESS PERMISSIONS ON ORACLE DIRECTORIES AND FILES - FROM THE INSTALLATION GUIDE		
Directories/Files	***Permissions***	***Comments***
All database, redo log, and control files (extensions for these files are typically Dbf, .log, and .ctl)	640 rw-r-----	To maintain discretionary access to data, all databases, redo logs, and control files must be readable only by the *oracle* account and oinstall group.
$ORACLE_HOME /bin/	755* rwxr-x--x	Must be writeable by the *oracle* software owner, and executable by all users.
$ORACLE_HOME/bin/oracle $ORACLE_HOME/bin/dbsnmp $ORACLE_HOME/bin/oradism	6751*** rws-r-s--x	The 6 sets the setuid bit and the setgid bit so the executables run as the *oracle* user and DBA group, regardless of who executes them.
All other executables.	755 ** rwxr-xr--x	Must be writeable by the *oracle* account and executable by others.
$ORACLE_HOME/lib/	755 ** rwxr-xr-x	The directory is readable, writeable, and executable by the owner, readable and executable by all other users.
All files under $ORACLE_HOME/lib/	644 rw-r--r--	The files are readable and writeable by the owner, read-only for all other users.
$ORACLE_HOME/rdbms/log	751 ** rwxr-x--x	Restricts access to files in the directory to the *oracle* account and ORAINENTORY group.
Product subdirectories such as $ORACLE_HOME/sqlplus or $ORACLE_HOME/rdbms	751 ** rwxr-x--x	Restricts access to log files to the *oracle* account and ORAINVENTORY group.
Files in $ORACLE_HOME/sqlplus or $ORACLE_HOME/rdbms	644 rw-r--r--	The files are readable and writeable by the owner, read-only for all other users.
$ORACLE_HOME/network/trace	777 rwxrwxrwx or 730 **** rwx-wx---	777 allows broad access to view and create trace files during development. Use 730 in a production environment to ensure that only the *oracle* account and the ORAINVENTORY group have access to trace files.
All files under product admin directories like $ORACLE_HOME/rdbms/admin and $ORACLE_HOME/sqlplus/admin	644 rw-r--r--	SQL scripts should typically be run as the SYS user.

Table 22. ORACLE UNIX FILE PERMISSIONS

* For versions earlier than 9.2, the permissions for these files should be set to 751 or more restrictive.

** Files in these directories should be set 750 or more restrictive than the installation default.

***A variance to Oracle’s listed permissions for these files is required. The set group id is not required and should not be set. Thus the permissions should be 4751.

****The $ORACLE_HOME/network/trace should have file permissions set to 730, which is different than the default installation setting. This will prevent unauthorized user from viewing potentially sensitive information stored in database log and trace files.

Verify that all oracle files have as their group the oracle dba group.

Use chgrp to change group ownership of all files in the current directory:

```
% chgrp dba *
```

- Check that all directories in the path of $ORACLE_HOME have at least a 755 mask.

  For example, if ORACLE_HOME = /u01/app/oracle/product/9.2.0.1.0, then the directories…

  app/oracle   and   app/oracle/product/9.2.1.0

  should all have a mask of 755. This means their permissions should be: rwxr-xr-x.

  If they are less restrictive then modify them with the command:

  ```
  % chmod 755 <directory_name>
  ```

- Verify that the only Oracle executables with the SETUID bit set are oracle.exe, oradism.exe, and dbsnmp.exe. If Oracle Internet Directory is in use, then oidldapd.exe also requires the SETUID bit.

  ```
  ls -l $ORACLE_HOME/bin/*
  ```

  Confirm that the following executables show permissions as follows:

  ```
  -rwsr-s---   1 oracle   dba   7330847 Oct 20 11:01 oracle
  -rwsr-x---   1 oracle   dba     62009 Dec 31 1993  dbsnmp
  -r-sr-s---   1 root     dba      9807 Feb 27 2003  oradism
  ```

If necessary to correct the ownership or permissions of any entries, use the following as an example:

```
chmod 4750 $ORACLE_HOME/bin/oracle  (for SETUID specific files)
chmod 751 $ORACLE_HOME/sqlplus/* (for other files)

chown oracle $ORACLE_HOME/bin/dbsnmp (for all files in the oracle directories except
oradism which must be owned by root)
```

E.14.2 Windows File Permissions

The following Windows file permission specifications for Oracle directories and files is from the Oracle *9i Database Installation Guide Release 2 for Windows*:

DIRECTORY	***GROUP AND PERMISSIONS***
\ORACLE_BASE\ ORACLE_HOME	- Administrators - Full Control - System - Full Control - Authenticated Users - Read, Execute, and List Contents
- \ORACLE_BASE\admin\ database_name - \ORACLE_BASE\oradata\database_name - \ORACLE_BASE\ORACLE_HOME\database\ - spfile SID.ora	- Administrators - Full Control - System - Full Control

Table 23. ORACLE WINDOWS FILE PERMISSIONS

NOTE 1: By default, the Oracle database Windows services use the Windows local SYSTEM built-in security account. Therefore, file permissions must be granted to the SYSTEM account of the local computer running the Oracle database. However, this STIG requires that a custom account be created for exclusive use for the Oracle services. This account is a member of the local Administrators group. Only SYSTEM, BUILTIN\Administrators group, and the DBA group if created (ORA_DBA) may be granted Full-Control access to the Oracle directories.

NOTE 2: To remove inherited user rights, you must deselect the ‘inherit from parent option’.

NOTE 3: Restrict access to the Program Files\Oracle folder to Oracle DBA, SYSTEM, and BUILTIN\Administrators only.

E.14.3 Special Files

- The $ORACLE_BASE/dbs/Orapwd<SID> file stores passwords for privileged database users in encrypted format. Restrict access to the ORAPWD<*SID*>.ORA file to the appropriate users.

- UNIX: chmod 640 $ORACLE_HOME/dbs/ORAPWD<SID>.ora

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME\dbs\ORAPWD<SID>.ora, select Properties, Security tab, remove all access privileges except those assigned to local Administrators, and DBAs.

- The $ORACLE_HOME/network/admin/listener.ora file contains the password for the Oracle listener. If the password has been defined by editing this file, the password is stored in clear text rather than in encrypted format. Restrict access to the listener.ora file to the appropriate users.

- UNIX: chmod 640 $ORACLE_HOME/rdbms/admin/listener.ora

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME\rdbms\admin\listener.ora, select Properties, Security tab, and remove all access privileges except those assigned to local Administrators, and DBAs.

- The $ORACLE_HOME/network/admin/Dbsnmp_rw.ora and Dbsnmp_ro.ora files support the Oracle Intelligent Agent. They may contain the password of the DBSNMP database account if the password has been changed from the default. Restrict access to these files to the appropriate accounts.

- UNIX: chmod 640 $ORACLE_HOME/rdbms/admin/dbsnmp_rw.ora

- UNIX: chmod 640 $ORACLE_HOME/rdbms/admin/dbsnmp_ro.ora

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME\rdbms\admin\dbsnmp_rw.ora, select Properties, Security tab, and remove all access privileges except those assigned to local Administrators, and DBAs.

- The $ORACLE_HOME/network/admin/sqlnet.ora (and protocol.ora file for Oracle database version 8i and 8) file contains network configuration parameters for the listener. Restrict access to the sqlnet.ora file to the the appropriate users.

- UNIX: chmod 640 $ORACLE_HOME/rdbms/admin/sqlnet.ora

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME\rdbms\admin\sqlnet.ora, select Properties, Security tab, and remove all access privileges except those assigned to local Administrators, and DBAs.

- The $ORACLE_HOME/network/log directory contains the log files produced by the Oracle listener, the Intelligent Agent, and other network components. Restrict access to these files in this directory to the appropriate users.

- UNIX: chmod 640 $ORACLE_HOME/network/log/listener.log

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME\network\log\listener.log, select Properties, Security tab, and remove all access privileges except those assigned to local Administrators, and DBAs.

- The $ORACLE_HOME/*/log-or-trace files directory contains the log files produced by the Oracle listener, the Intelligent Agent, and other network components. Restrict access to these files in this directory to the appropriate users.

- Following is a list of possible log and trace file directories all found under ORACLE_HOME:
  - Admin/bdump, admin/cdump, admin/create, admin/udump
  - Ctx/log
  - Hs/log
  - Ldap/log
  - Network/log
  - Otrace/admin
  - Sysman/log

- UNIX: chmod 640 $ORACLE_HOME/*/log-trace-directory/*.log and *.trc

- WINDOWS: From Windows Explorer, right click on $ORACLE_HOME*\log-trace-directory*.log or *.trc, select Properties, Security tab, and remove all access privileges except those assigned to local Administrators, and DBAs.

E.14.4 Windows Registry Permissions

Restrict access to the HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE keys to Full Control to local Administrators and the Oracle DBA group. Grant only Read permissions to local groups of users that may require it. To remove/adjust permissions:

1. Open the registry.
2. Select HKEY_LOCAL_MACHINE.
3. Expand SOFTWARE.
4. Select Oracle.
5. Select Permissions from the Security main menu.

The *Registry Key Permissions* dialog box appears.

6. Click on the Add button to add Administrators and/or Oracle DBAs if necessary; Select Full Control; Select Apply.
7. Select the name to remove. Click on the Remove button. Select Apply.
8. Click on the OK button.
9. Exit the registry.

E.14.5 OS/390 Security Settings

The following security steps must be completed to secure an Oracle installation on OS/390 systems. For details on completing these steps, please refer to Oracle's *Oracle9i Enterprise Edition Installation Guide Release 1 for OS/390* or Oracle's *Oracle8i Enterprise Edition for OS/390 Installation Guide.*

- Configure library security.
- Configure VSAM File Security.
- Secure Oracle MPM restricted commands CRTCNV, MPMCMD, SVRMGRL.
- Disable user of MPM user logon and role exits.
- Configure resource classes for Oracle OSDI.
- Configure resource profiles for Oracle OSDI binds.
- Set LOGIN_AUTH to None or SAF.

E.15 Set the Audit Configuration

E.15.1 Auditing Options

There are three (3) types of auditable events—use of system privileges, use of object privileges, or issuance of statements. Activating some auditing options sometimes activates others. For example, the use of a system privilege requires the issuance of a system command. Auditing for use of the privilege also audits for the statement.

- This STIG requires auditing of the use of all auditable system privileges.

  From SQLPLUS:

  ```
  AUDIT ALL PRIVILEGES BY ACCESS:
  AUDIT SYSDBA BY ACCESS;
  AUDIT SYSOPER BY ACCESS;
  AUDIT ALTER ANY OPERATOR BY ACCESS;
  ```

- This STIG additionally requires auditing of the SQL statements enabled by the following commands with the exceptions listed afterwards:

  From SQLPLUS:

  ```
  AUDIT ALTER SEQUENCE by access;
  AUDIT ALTER TABLE by access;
  AUDIT COMMENT TABLE by access;
  AUDIT GRANT DIRECTORY by access;
  AUDIT GRANT PROCEDURE by access;
  AUDIT GRANT SEQUENCE by access;
  AUDIT GRANT TABLE by access;
  AUDIT GRANT TYPE by access;
  ```

The following SQL statements will disable audits set by the commands above that are not required:

```
NOAUDIT EXECUTE ANY LIBRARY;
NOAUDIT EXECUTE ANY PROCEDURE;
NOAUDIT EXECUTE ANY TYPE;
NOAUDIT EXECUTE LIBRARY;
NOAUDIT LOCK ANY TABLE;
NOAUDIT SELECT ANY SEQUENCE;
NOAUDIT SELECT ANY TABLE;
NOAUDIT UPDATE ANY TABLE;
NOAUDIT DELETE ANY TABLE;
NOAUDIT EXECUTE ANY INDEXTYPE;
NOAUDIT EXECUTE ANY OPERATOR;
NOAUDIT INSERT ANY TABLE;
NOAUDIT NETWORK;
NOAUDIT DELETE TABLE;
NOAUDIT INSERT TABLE;
NOAUDIT UPDATE TABLE;
NOAUDIT EXECUTE PROCEDURE;
NOAUDIT SELECT TABLE;
NOAUDIT SELECT SEQUENCE;
```

- The only application *object* auditing required is RENAME. This option must be applied by default to any newly created objects.

  From SQLPLUS:

  ```
  Audit rename on default by access;
  ```

  If application objects have already been created, then the audit rename on *object* statement should be issued for all application objects.

  From SQLPLUS:

  ```
  Audit rename on application_object_name by access;
  ```

E.15.2 Protecting Auditing Data

If the audit table has been created in the database, auditing for update and delete must be enabled whether or not the table is currently in use. Old records may exist or auditing could be redirected to the database tables. If you do not wish to maintain auditing on these tables, then drop the table from your database.

For an audit trail stored within the database:

- From SQLPLUS:

  Audit update, delete on SYS.AUD$;

For audit trails stored in external OS files, verify with the System Administrator that these files are being audited for the same.

- Verify that only authorized database accounts have access to the audit data:

  From SQLPLUS:

  Select GRANTEE, PRIVILEGE from DBA_TAB_PRIVS where TABLE_NAME = 'AUD$' and GRANTEE !='DELETE_CATALOG_ROLE';

  Review the list to confirm that any assignments are authorized.

- Verify that audit data stored in the database is owned by SYS, SYSTEM, or an authorized user.

  From SQLPLUS:

  Select OWNER from DBA_TABLES where TABLE_NAME='AUD$';

- Verify that only authorized database accounts have privileges to enable or disable auditing.

  From SQLPLUS:

  Select GRANTEE from DBA_SYS_PRIVS where PRIVILEGE like '%AUDIT%';

E.16 Revoke Privileges Assigned to PUBLIC

NOTE: Revoking all default installation privilege assignments from PUBLIC is not required at this time. However, execute permissions to the specified packages is required to be revoked from public.

PUBLIC is a default system role to which every database account is granted membership automatically. While convenient, use of this role to grant privileges to database accounts bypasses the responsibility to assign privileges with discrimination.

At a minimum, revoke the following:

From the SQLPLUS prompt:

```
Revoke execute on SYS.UTL_SMTP from PUBLIC;
Revoke execute on SYS.UTL_TCP from PUBLIC;
Revoke execute on SYS.UTL_HTTP from PUBLIC;
Revoke execute on SYS.UTL_FILE from PUBLIC;
Revoke execute on SYS.DBMS_RANDOM from PUBLIC;
Revoke execute on SYS.DBMS_LOB from PUBLIC;
Revoke execute on SYS.DBMS_SQL from PUBLIC;
Revoke execute on SYS.DBMS_JOB from PUBLIC;
Revoke execute on SYS.DBMS_BACKUP_RESTORE from PUBLIC;
```

NOTE: DBMS_BACKUP_RESTORE does not exist in 9i and later.

From SQLPLUS:

- Revoke any system privileges granted to PUBLIC
  (No system privileges are assigned to PUBLIC by default.)
  select privilege from dba_sys_privs where grantee='PUBLIC';

  For each privilege returned, issue:
  Revoke *privilege* from PUBLIC;
  Replace *privilege* with the system privilege to revoke.

- Revoke any custom roles granted to PUBLIC
  (No roles are assigned to PUBLIC by default.)
  Select granted_role from dba_role_privs where grantee='PUBLIC';

For each role listed, issue:
Revoke *role* from PUBLIC;
Replace *role* with the role to revoke.

- Revoke custom object privileges from PUBLIC.
  Select privilege||' '||owner||'. '||table_name from dba_tab_privs
    Where grantee='PUBLIC'
    And owner not in('SYS', 'CTXSYS', 'MDSYS', 'ODM', 'OLAPSYS', 'MTSSYS', 'ORDPLUGINS', 'ORDSYS', 'SYSTEM', 'WKSYS', 'WMSYS', 'XDB', 'LBACSYS');

  For each privilege returned, issue:
  Revoke privilege on object_name from PUBLIC;
  You will need to connect as the user that granted the privilege in order to revoke it.

- Following is a method to create scripts to remove all default object privileges assigned to PUBLIC. This is not a current requirement, however, it may be implemented to ensure the concept of least privilege is applied to your database. Make sure that you create custom roles and assign any necessary privileges to system objects to those roles.

NOTE: The spool files resulting from the SQL command below require editing. The command entry and results lines need to be deleted. Revoke statements will be created in order of object owner. Before each new owner, create a “connect username;” statement to connect as that user. Do not enter the password in the file! You will be prompted for it when it is run. In some cases the object owner is not the object privilege grantor.

```
Set pagesize 0
Set linesize 132

spool c:\temp\pubjavaprivs.sql;
Select 'revoke execute on '||owner||'."'||TABLE_NAME||'" from PUBLIC; ' from
DBA_TAB_PRIVS
  where GRANTEE='PUBLIC'
  and TABLE_NAME in (select object_name from dba_objects where object_type =
'JAVA CLASS')
  order by owner;

Select 'revoke execute on JAVA SOURCE '||OWNER||'."'||TABLE_NAME||
  '" from PUBLIC; ' from DBA_TAB_PRIVS
  where GRANTEE='PUBLIC'
  and TABLE_NAME in
  (select object_name from dba_objects where object_type ='JAVA SOURCE')
   order by OWNER;
```

```
Select 'revoke execute on JAVA RESOURCE '||OWNER||'."'||TABLE_NAME||
  '" from PUBLIC; ' from DBA_TAB_PRIVS
  where GRANTEE='PUBLIC'
  and TABLE_NAME in
  (select object_name from dba_objects where object_type = 'JAVA RESOURCE')
  order by OWNER;
spool off;
##
spool c:\temp\pubjavapolicy.sql
select 'call dbms_java.disable_permission('||seq||');' from dba_java_policy
where grantee='PUBLIC' and enabled='ENABLED' and kind='GRANT';
spool off;
##
spool c:\temp\pubprivs.sql
Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type in
  ('LIBRARY', 'PROCEDURE', 'FUNCTION', 'PACKAGE'))
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type ='OPERATOR')
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type in
  ('INDEXTYPE', 'INDEX PARTITION', 'INDEX'))
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type ='TABLE')
  order by OWNER;
```

```
Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type in
  ('VIEW', 'MATERIALIZED VIEW'))
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type ='TYPE')
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC; ' from
DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type ='SYNONYM')
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on '||OWNER||'.'||TABLE_NAME||' from PUBLIC force;'
from DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type in
  ('PACKAGE BODY', 'TYPE BODY'))
  order by OWNER;

Select 'revoke '||PRIVILEGE||' on directory '||OWNER||'.'||TABLE_NAME||' from
PUBLIC; ' from DBA_TAB_PRIVS
where GRANTEE='PUBLIC'
and TABLE_NAME in
  (select object_name from dba_objects where object_type ='DIRECTORY')
  order by OWNER;
spool off;
```

E.17 Secure Host Directory Access

The UTL_FILE capability in Oracle, allows users to gain access to host system files and directories. Applications may take advantage of this capability for a variety of reasons. However, this capability may be exploited and grant unauthorized access to the host system file structure. The following SQLPlus command will list any procedures that use this feature. Verify that any use of this package is justified.

From SQLPLUS:

```
Select distinct dbo.object_type, dbo.object_name from
  dba_objects dbo, dba_tab_privs dbt, dba_source ds
  where dbo.owner not in
    ('CTXSYS','DBSNMP','MDSYS','ORDPLUGINS',
     'ORDSYS','OAS_PUBLIC','OUTLN','SYS','SYSTEM')
  and dbo.object_type in ('PACKAGE BODY','PACKAGE','PROCEDURE')
  and dbo.object_name !='UTL_FILE'
  and dbt.owner = dbo.owner
  and dbt.table_name = dbo.object_name
  and ds.owner = dbo.owner
  and ds.name = dbo.object_name
  and (ds.text like '%UTL_FILE%' or ds.text like '%utl_file%');
```

E.18 Revoke Privileges to Assign Permissions

DBA, application owner, and application administrator accounts should be the only database accounts with the privilege to assign permissions to other users. While object owners gain this privilege automatically, in a secure environment, only the DBAs carry the privilege to create objects in a production environment. Application objects are created only during application installation and maintenance operations. The following SQLPlus statements will list all users and roles that have been assigned these administrative privileges.

- Check for roles granted with the administrative option:

  From SQLPLUS:

  ```
  Select GRANTEE, GRANTED_ROLE from DBA_ROLE_PRIVS where
  ADMIN_OPTION='YES'
  and GRANTEE not in ('SYS','SYSTEM','DBA', 'LBACSYS ', 'WKSYS ');
  ```

  Revoke *role* from *username*;
  Grant *role* to *username;* (without admin_option)

- Check for system privileges granted with the administrative option:

  From SQLPLUS:

  ```
  Select GRANTEE, PRIVILEGE from DBA_SYS_PRIVS where
  ADMIN_OPTION='YES' and GRANTEE not in
  ('SYS','SYSTEM','DBA','AQ_ADMINISTRATOR_ROLE', 'MDSYS', 'LBACSYS');
  ```

  Revoke *system privilege* from *username*;

- Find and disable object privileges to ensure custom users/roles or PUBLIC do not have administrative privileges:

  From SQLPLUS:

  ```
  Select GRANTEE||' '||PRIVILEGE||' '||OWNER||'.'||TABLE_NAME from
  DBA_TAB_PRIVS
  where GRANTABLE='YES' and
  GRANTEE not in
  ('SYS','SYSTEM','DBA', 'OLAPSYS', 'CTXSYS', 'PUBLIC', 'LBACSYS')
  and TABLE_NAME not in
  (Select SYNONYM_NAME from DBA_SYNONYMS where
  SYNONYM_NAME=TABLE_NAME);
  ```

E.19 Revoke Predefined Role Assignments

Predefined roles are those roles defined by default during an Oracle installation. Depending upon which options were selected, the list of predefined roles within an environment can vary. Default roles are assigned privileges based on Oracle's default user group types and are not appropriate for most general application usage. For example, the CONNECT default role has the privilege to create several object types including tables, views, and database links. Custom roles should be used that limit privileges to those specifically needed for the user to perform the assigned function.

NOTE: You may have different predefined roles defined at your site based on the options selected at installation. Add any not listed below in the command when you issue it.

From SQLPLUS:

```
Select GRANTEE, GRANTED_ROLE from DBA_ROLE_PRIVS
where GRANTED_ROLE in (
'AQ_ADMINISTRATOR_ROLE',
'AQ_USER_ROLE',
'CONNECT',
'CTXAPP',
'DBSNMP',
'DELETE_CATALOG_ROLE',
'EXECUTE_CATALOG_ROLE',
'EXP_FULL_DATABASE',
'HS_ADMIN_ROLE',
'IMP_FULL_DATABASE',
'JAVA_ADMIN',
'JAVADEBUGPRIV',
'JAVAIDPRIV',
'JAVASYSPRIV',
'JAVAUSERPRIV',
'OEM_MONITOR',
'OSDBA',
'OSOPER',
'OUTLN',
'PLUSTRACE',
'RECOVERY_CATALOG_OWNER',
'RESOURCE',
'SELECT_CATALOG_ROLE',
'SNMPAGENT',
'SYS',
'SYSDBA',
'SYSOPER',
'SYSTEM',
'TIMESERIES_DBA',
'TIMESERIES_DEVELOPER',
'TKPROFER')
 and GRANTEE not in
('SYS','SYSTEM','DBA','EXP_FULL_DATABASE','IMP_FULL_DATABASE',
'EXECUTE_CATALOG_ROLE','JAVASYSPRIV','OEM_MONITOR',
'OUTLN', 'WKSYS', 'OSE$HTTP$ADMIN', 'ORDPLUGINS', 'LBACSYS',
'WKUSER', 'ORDSYS', 'SELECT_CATALOG_ROLE', 'CTXSYS',
'AURORA$JIS$UTILITY$',’DBSNMP’);
```

Revoke *role* from *username/role*;

E.20 Application Administration Roles Enabled by Default

Application Administration roles are determined by the granting of create user, alter user, and drop user privileges. These roles should not be enabled by default upon connection to the database, but should be enabled/disabled as required by the application administration function.

From SQLPlus:

```
Select grantee||' '||granted_role from dba_role_privs
  Where default_role='YES'
  And granted_role in
  (select grantee from dba_sys_privs where privilege like '%USER%'
  and grantee not in ('CTXSYS', 'DBA', 'IMP_FULL_DATABASE', 'MDSYS',
  'SYS', 'WKSYS'))
  and grantee not in ('DBA', 'SYS', 'SYSTEM');
```

For each role assignment returned, issue:

```
Alter user username default role all except role;
```

If the user has more than one application administration role assigned, then you will have to remove assigned roles from default assignment and assign individually the appropriate default roles.

E.21 Configure Privileges Assigned to Users

Efficient administration of privileges improves system security. The categorization of user privilege requirements by function and the use of roles to assign these privileges improve the efficiency of privilege administration.

- Check for any system privileges assigned to non-default users and roles or PUBLIC.

  System privileges should not be granted directly to any user nor should they be granted to application user roles. From the listing of users and roles produced from the following SQL statement, verify that roles granted system privileges are authorized and are not assigned directly to database accounts or PUBLIC.

  From SQLPLUS:

```
    Select GRANTEE,PRIVILEGE from DBA_SYS_PRIVS
    where PRIVILEGE <> 'CREATE SESSION'
    and GRANTEE not in
('AQ_ADMINISTRATOR_ROLE',
'AQ_USER_ROLE',
'AURORA$ORB$UNAUTHENTICATED',
'CONNECT',
'CTXAPP',
'DBSNMP',
'DELETE_CATALOG_ROLE',
'EXECUTE_CATALOG_ROLE',
'EXP_FULL_DATABASE',
'HS_ADMIN_ROLE',
'IMP_FULL_DATABASE',
'JAVA_ADMIN',
'JAVADEBUGPRIV',
'JAVAIDPRIV',
'JAVASYSPRIV',
'JAVAUSERPRIV',
'MDSYS',
'OEM_MONITOR',
'OSDBA',
'OSOPER',
'OUTLN',
'PLUSTRACE',
'RECOVERY_CATALOG_OWNER',
'RESOURCE',
'SELECT_CATALOG_ROLE',
'SNMPAGENT',
'SYS',
'SYSDBA',
'SYSOPER',
'SYSTEM',
'TIMESERIES_DBA',
'TIMESERIES_DEVELOPER',
     'TKPROFER');
```

To remove unauthorized privileges:
Revoke privilege from username/role;

- Check for any object privileges granted directly to users.

  Object privileges should be granted to roles and the roles granted to the appropriate users. Any results of the command below indicate direct assignment of object privileges.

  From SQLPLUS:

  ```
  Select distinct GRANTEE from DBA_TAB_PRIVS
  where GRANTEE not in (select ROLE from DBA_ROLES);
  ```

  Review the list of GRANTEEs to see if any are not Oracle predefined accounts. For any custom accounts, connect as the grantor of the privilege and:

  Select PRIVILEGE||’ ‘||OWNER||’.’||TABLE_NAME from
  DBA_TAB_PRIVS where GRANTEE =’*myGrantee’*;

  Revoke *privilege* on *owner.object* from *username/role*;

  To determine non-default object privileges assigned to PUBLIC:

  ```
  Select privilege||' on '||owner||'. '||table_name from dba_tab_privs
    Where grantee=’PUBLIC’
    And owner not in ('SYS', 'CTXSYS', 'MDSYS', 'ODM', 'OLAPSYS',
    'MTSYS', 'ORDPLUGINS', 'ORDSYS', 'SYSTEM', 'WKSYS', 'XDB',
    'LBACSYS');
  ```

- No users or roles should have the Alter or Reference privilege on any database objects.

  From SQLPLUS:

  ```
  Select GRANTEE||' '||PRIVILEGE||' '||OWNER||'.'||TABLE_NAME from
  DBA_TAB_PRIVS
  where (PRIVILEGE like '%ALTER%' or PRIVILEGE like '%REFERENCE%')
  and GRANTEE !='SYSTEM'
  and GRANTOR != 'MDSYS';
  ```

Revoke *privilege* on *owner.object* from *username/role*;

- Verify that users do not have access to DBA data.

From SQLPLUS:

```
Select GRANTEE||' '||PRIVILEGE||' '||TABLE_NAME from DBA_TAB_PRIVS
          where
        (owner='SYS' or table_name like 'DBA_')
        and grantee not in ('AQ_ADMINISTRATOR_ROLE', 'AQ_USER_ROLE',
        'AURORA$JIS$UTILITY$', 'DBA', 'EXECUTE_CATALOG_ROLE',
        'EXP_FULL_DATABASE', 'HS_ADMIN_ROLE',
        'IMP_FULL_DATABASE', 'ORDSYS',
        'OSE$HTTP$ADMIN', 'OUTLN', 'PUBLIC',
        'SELECT_CATALOG_ROLE', 'SNMPAGENT', 'SYSTEM',
        'DELETE_CATALOG_ROLE',
        'GATHER_SYSTEM_STATISTICS', 'LOGSTDBY_ADMINISTRATOR',
        'MDSYS', 'ODM', 'OEM_MONITOR', 'OLAPSYS', 'WKUSER', 'WMSYS',
        'WM_ADMIN_ROLE', 'XDB', 'TRACESVR')
        and grantee not in (select grantee from dba_role_privs where
        granted_role='DBA');
```

Revoke *privilege* on *owner.object* from *role/username*;

- Verify that any object owner accounts have been disabled.

Object owners are implicitly assigned special privileges to their objects by virtue of their ownership.

From SQLPLUS:

```
    Select distinct owner from dba_objects, dba_users
    Where owner not in ('SYS', 'SYSTEM', 'MDSYS', 'CTXSYS',
    'ORDSYS', 'ORDPLUGINS', 'AURORA$JIS$UTILITY$',
    'ODM', 'ODM_MTR', 'OLAPDBA', 'OLAPSYS', 'MTSSYS',
    'OSE$HTTP$ADMIN', 'OUTLN', 'LBACSYS',
    'PUBLIC', 'DBSNMP', 'RMAN', 'WKSYS',
    'WMSYS', 'XDB')
    and owner=username
    and account_status not like '%LOCKED';
```

Alter user *username* account lock;

E.22 Disable the PL/SQL EXTPROC Module

If the EXTPROC module is not in use, then it should be removed and/or disabled. Edit the listener.ora and tnsnames.ora files in the $ORACLE_HOME/network/admin directory and remove one of the following entries from each of the configuration files (depends upon the OS and release of the Oracle Database server installed):

* icache_extproc, or
* PLSExtproc, or
* extproc

Remove the EXTPROC executable from the $ORACLE_HOME/bin directory.

E.23 Configure a Non-Default Port for the Oracle Listener

To protect the listener from casual access do not use the well-known default port of 1521. To change the default port edit the LISTENER.ORA file to specify a different available port. Following is an example:

```
MYLISTENR =
 (DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.122.67)(PORT = 1527))
 )
```

When the default name and port assignment for the listener are not used, then the listener must be defined in the INIT<SID>.ORA file using the LOCAL_LISTENER parameter. An example of this parameter seeing follows:

```
LOCAL_LISTENER= (ADDRESS = (PROTOCOL = TCP)(HOST =
192.168.122.67)(PORT = 1527))
```

E.24 Listener Connection Request Timeout

The listener will wait indefinitely for a client to complete a database network connection. This leaves the listener vulnerable to a denial of service attack. To prevent this, specify a timeout limit for the listener. In Oracle 8i, this is done by specifying the CONNECT_TIMEOUT_listener_name parameter in the LISTENER.ORA file. In Oracle 9i, use the LISTENER.ORA parameter INBOUND_CONNECT_TIMEOUT_listener_name to specify the timeout limit. For 9i, the SQLNET.ORA parameter SQLNET.INBOUND_CONNECT_TIMEOUT parameter is also required on the database host server. The timeout is expressed in seconds and should be set to 3 seconds or less where feasible.

```
8i:  LISTENER.ORA:  CONNECT_TIMEOUT_MyListener = 3
9i:  LISTENER.ORA:  INBOUND_CONNECT_TIMEOUT_MyListener = 3
9i:  SQLNET.ORA:    SQLNET.INBOUND_CONNECT_TIMEOUT = 3
```

E.25 Restrict Database Access from the Network

To further protect your database from unauthorized remote access, network address restrictions may be enforced by the Oracle listener. Network address restriction is required when the PLSQL EXTPROC is in use to protect against unauthenticated access to the database. (Please see Oracle Alert #29 for more information.) To enable network address restriction, edit the SQLNET.ORA file on the database host system to include the following:

```
tcp.validnode_checking = YES
tcp.invited_nodes = {list of IP addresses}
tcp.excluded_nodes = {list of IP addresses}
```

E.26 Disable/Secure the Oracle XML Protocol Server

The Oracle XML Protocol Server is configured during database installation if XML Support is selected. Unless required, the XML Protocol Server should be disabled. If required, the XML Protocol server should be configured to log a minimum of unsuccessful logins. To disable the XML Protocol Server, remove any dispatcher reference to it in the INIT<SID>.ORA file. Such a reference usually appears as:

```
Dispatchers=”(PROTOCOL=TCP)(SERVICE=<SID>XDB)”
```

If the XML Protocol Server is required, then the logging of logins must be enabled. This configuration is stored in the XML resource named /xdbconfig.xml and is most easily configured using Enterprise Manager. Expand the Database, expand XML Database, select Configuration. The XML Database Parameters table will be displayed. Set both the ftp-log-level and http-log-level parameters to a value of 1 instead of 0.

It is also a good idea to configure the ports to a non-default port. Specify a port other than 2100 for the FTP port and 8080 for the http port if possible.

E.27 Remove OEM Database Components/Intelligent Agent

If not required, remove the Oracle Enterprise Manager (OEM) database components. This can be done by running the $ORACLE_HOME/rdbms/admin/catnsnmp.sql procedure. Also delete or rename the Oracle Intelligent Agent executable file $ORACLE_HOME/bin/dbsnmp.

E.28 Database Link Passwords

Database links define connections to external databases. This STIG stipulates that database links will only be used to support database replication. Database links are required to use the current database session connection credentials to access remote databases. This requirement prevents the storage of static or fixed user database accounts with their unencrypted passwords inside the database link table, SYS.LINK$.

- List any defined static database links without passwords.

  From SQLPLUS:

  ```
  Select NAME from LINK$ where PASSWORD is NOT NULL;
  ```

- Drop the current database link after noting definition.

  Drop database link *database link*;

- Recreate the database link without a password.

  From SQLPLUS:

  Create [PUBLIC] database link *dblink name*.

- Confirm that replication is being used if any database links were found.

  From SQLPLUS:

  ```
  Select GNAME from REPGROUP;
  ```

E.29 Configure Database Architecture

- To prevent loss of service during disk a failure, multiple copies of Oracle control files should be maintained on separate disks.

  From SQLPLUS:

  ```
  Select * from v$controlfile;
  ```

  To create additional control files:

  Edit the existing init.ora file.
  Find the line with “CONTROLFILES=”
  Add the full file spec for the new control file to the line.
  Save the changed file.
  Shut down the database.
  Copy one of the existing control files to the name and location of the new control file.
  Restart the database.

- Confirm that at least two redo log file groups with two members each are available:

  From SQLPLUS:

  ```
  Select * from v$logfile;
  ```

  To define additional redo log file groups:

  ```
  ALTER DATABASE database
    ADD LOGFILE GROUP 3
     ('diska:log3.log' ,
      'diskb:log3.log') SIZE 50K;
  ```

  To add additional redo log files to an existing redo log file group:

  ```
  ALTER DATABASE database
    ADD LOGFILE MEMBER 'diskc:log3.log'
    TO GROUP 3;
  ```

E.30 Configure Domain Prefix for Windows OS Authentication for Oracle 8.1.5 and Earlier Versions

Use the Windows REGEDT32 utility to configure this registry setting. From the Windows desktop of the database host system, type Start/Run and enter REGEDT32 in the Open: text box. Click the OK button to run the utility.

Select the HKEY_LOCAL_MACHINE window.
Expand Software; expand Oracle; select the HOME*ID* where *ID* is the Oracle home number.
Select Edit/Add Value from the menu bar.
Enter Value Name = OSAUTH_PREFIX_DOMAIN
Enter Data Type = REG_SZ
Enter String = TRUE
Close REGEDT32

E.31 Secure SQL*Plus Host Command

The SQL*Plus HOST command grants access from within the SQL*Plus to the host command line with the authorization of the Oracle process on the host system. At a minimum, the HOST command should be disabled to prevent such access. Review additional SQL*Plus commands in Oracle's *SQL*Plus User's Guide and Reference* for inclusion in the access control table PRODUCT_USER_PROFILE.

From SQL*Plus:

```
INSERT INTO PRODUCT_USER_PROFILE
VALUES ('SQL*Plus', '%', 'HOST', NULL, NULL, 'DISABLED', NULL, NULL);
```

E.32 Assign Non-SYSTEM Default and Temporary Tablespaces to Users

To view database accounts assigned the SYSTEM tablespace as either their default or temporary tablespace issue the following SQL command:

```
Select username from dba_users where default_tablespace = 'SYSTEM';
```

For each custom-defined user listed from above, issue the following SQL command replacing "USERS" with the name of the default tablespace for users system and USERNAME with the name of the user to alter.

```
Alter user USERNAME default tablespace DEFAULT;
```

Repeat these two steps for the temporary tablespace assignment, replacing TEMPORARY with the system temporary tablespace for users, using the following SQL commands:

```
Select username from dba_users where temporary_tablespace='SYSTEM';
```

```
Alter user USERNAME temporary tablespace TEMPORARY;
```

Revoke any unnecessary explicit tablespace quotas on the SYSTEM tablespace. To view explicit assignments, use the SQL command:

```
Select username from dba_ts_quotas
where max_bytes > 0 and tablespace_name=’SYSTEM’;
```

E.33 Set Database Archive Mode On

To view the database archive log mode, issue the SQL command:

```
Select log_mode from v$database;
```

Before changing the archivelog mode, make sure that you have storage capacity for the archived redo log files.

If the value returned is NOARCHIVELOG, then archive mode needs to be activated. To set archive log mode on, issue the following SQL statements (the database must be shutdown):

```
Shutdown;
Startup mount exclusive;
Alter database archivelog;
Shutdown;
Startup;
```

E.34 Disable the Oracle Trace Utility

The Oracle trace utility is enabled by default. To disable it, simply delete all *.dat files from the Oracle trace directory. The database must be shutdown during the following file deletion.

For UNIX, do the following:

```
Cd $ORACLE_HOME/otrace/admin
Rm –f *.dat
```

For Windows from the command prompt (replace ORACLE_HOME with the Oracle home directory path:

```
Cd ORACLE_HOME\otrace\admin
Del *.dat
```

E.35 Encrypt Stored Procedures

You can use WRAP utility:

e.g., proc.sql containts the sql to create the procedure.

having tested the procedure, u can use wrap utility as
$wrap iname=proc.sql

this will generate proc.plb, run proc.plb in sqlplus to create the encrypted procedure.

E.36 Create Baseline for Stored Procedures

The following function and SQL commands together define a capability to determine the MD5 hash value for stored procedure code. The returned hash values when stored may be used for later comparison to detect modification. Before running the procedure, consider spooling the results to a text file on the host. Output may also be directed to a database table with modification to the procedure.

```
CREATE OR REPLACE FUNCTION COMPUTE_MD5 (PROC_NAME_IN IN VARCHAR2)
RETURN VARCHAR2
IS
  all_text VARCHAR2(32767);
  cur_MD5  VARCHAR2(32767);
BEGIN
  for x in (select text from user_source where name=PROC_NAME_IN)
  loop
    cur_MD5  := dbms_obfuscation_toolkit.md5(input => utl_raw.cast_to_raw(x.text));
    all_text := dbms_obfuscation_toolkit.md5(input => (cur_MD5 || all_text));
  end loop;
  RETURN all_text;
END;
/
SHOW ERRORS;
SET SERVEROUTPUT ON SIZE 1000000;
DECLARE
BEGIN
    for x in (select distinct name from user_source)
    loop
      DBMS_OUTPUT.PUT_LINE(CHR(10));
      DBMS_OUTPUT.PUT_LINE('Procedure: ' || x.name) ;
      DBMS_OUTPUT.PUT_LINE('MD5: ' || COMPUTE_MD5(x.name));
    end loop;
END;
/
```

APPENDIX F MICROSOFT SQL SERVER DATABASE STIG COMPLIANCE CONFIGURATION

Most of the configurations options can be changed using either Transact SQL (T-SQL) or SQL Server Enterprise Manager.

F.1 Update Version/Apply Security Patches

- Check Current SQL Version.

  T-SQL:
  SQL Server 2000:
  select serverproperty('ProductVersion')
  select serverproperty('ProductLevel')

  SQL Server 7:
  Exec xp_msver productversion

  Enterprise Manager:
  Right-click on database name.
  Select Properties.
  Select General tab.
  View Product Version.

  Current versions: 8.00.543 SP2), 7.00.1063 SP4

- Install available Hotfixes or Service Packs according to patch instructions.

  - To view alerts reported by Microsoft, enter the Internet address http://www.microsoft.com/technet/treeview/default.asp?url=/technet/itsolutions/security/Default.asp in your browser and press Enter. Select Hot Fix and Security Bulletin Service and enter your SQL Server version in the product box. Review the list of Security Bulletins and download and apply any fixes as appropriate. A list of current service packs for your product can be found at http://support.microsoft.com/directory/content.asp?ID=FH;EN-US;sp&FR=0&SD=GN&LN=EN-US&CT=SD&SE=NONA.

  - Check for SQL server database IAVM notices on http://www.cert.mil.

F.2 Set Initialization Parameters

- Disable direct updates to system tables.

  T-SQL:
  Exec sp_configure 'allow updates', ‘0’
  reconfigure

  Enterprise Manager:
  Right-click on Server Name.
  Select Properties.
  Select Server Settings tab.
  Server Behavior/Clear checkbox for: Allow modifications to be made directly to the system catalogs.

- Enable C2 auditing (SQL Server 2000 only)

  T-SQL:
  Exec sp_configure 'c2 audit mode', ‘1’
  reconfigure

  Enterprise Manager:
  N/A

- Remote Access
  NOTE: This option should be disabled unless replication is in use or it is otherwise authorized.

  T-SQL:
  Exec sp_configure 'remote access', ‘0’
  reconfigure

  Enterprise Manager:
  Right-click on SQL Server name.
  Select Properties.
  Select Connections tab.
  Remote server connections/Clear checkbox for: Allow other SQL Servers to connect remotely to this SQL Server using RPC.

- Disable SCAN FOR STARTUP PROCS.

  T-SQL:
  Exec sp_configure ‘scan for startup procs’, ‘0’
  reconfigure

  Enterprise Manager:
  N/A

F.3 Accounts/Passwords

- Change Default sa account password.

  T-SQL:
  NOTE: *Replace* ***NU15pswd*** *with a custom password.*

  EXEC sp_password NULL, 'NU15pswd', 'sa'

  Enterprise Manager:
  Expand SQL Server name.
  Select Security.
  Select Logons.
  Double-click on user **sa**.
  Enter new password in password text box.
  Click OK.

- SQL Server uses Windows authentication only.

  T-SQL:
  N/A

  Enterprise Manager:
  Right-click on server name.
  Select Properties.
  Select Security tab.
  Security/Select Windows only.
  Click OK.

- Secure SQL Server Agent Service accounts.

1. Create local or domain user/group account for SQL Server services that is only a member of the users group. (If SQL Service is part of an Active Directory, the service account must be placed in the Power Users Group.)

   Usrmgr.exe:
   Start/Run/Usermgr.exe.
   Select User from menu bar.
   Select New User.
   Enter username ('AccountName').
   Click Add button.
   Click OK.
   Click OK.
   Close usrmgr.

2. Set account user rights. Act as part of the operating system, increase quotas, replace a process-level token, and log on as a service and deny this user the interactive logon right.

3. Add this user to SQLServer logons.

   T-SQL:
   Exec sp_grantlogin 'accountname'
   (Example: 'MSSQLService)

   Enterprise Manager:
   Expand SQL Server name.
   Expand Security.
   Right-click Logons.
   Select New Logon.
   Click on list button ('…') to retrieve list of logon names to select.
   Select AccountName.
   Click Add button.
   Click OK.

4. Grant this group the sysadmin privilege.

   T-SQL:
   exec sp_addsrvrolemember ‘accountname, ‘sysadmin’

   Enterprise Manager:
   Expand SQL Server name.
   Expand Security.
   Select Server Roles tab.
   Select System Administrators.
   Select Add.
   Select the ‘AccountName.’
   Select OK.
   Select OK.

5. The SQL Server Service will be configured later to use this account.

- An OS DBA Group has been created.
  NOTE: There are four steps to be followed:

  1. Create domain or local server group for DBAs for this SQLServer instance.

     Usrmgr.exe:
     Start/Run/Usermgr.exe.
     Select User from menu bar.
     Select New Local Group.
     Enter group name (‘MSSQL_DBA’).
     Click Add button.
     Select Windows accounts to be granted DBA group membership.
     Click Add.
     Click OK.
     Click OK.
     Close usrmgr.

2. Add this group to SQLServer logons.

   T-SQL:
   Exec sp_grantlogin 'ServerOrDomainName\Groupname'
   (Example: 'MSSQLSERVER\MSSQL_DBA')

   Enterprise Manager:
   Expand SQL Server name.
   Expand Security.
   Right-click Logins.
   Select New Login.
   Click on list button ('…') to retrieve list of login names to select.
   Select ServerOrDomainName\GroupName (MSSQLSERVER/MSSQL_DBA).
   Click Add button.
   Click OK.

3. Grant this group the sysadmin privilege.

   T-SQL:
   exec sp_addsrvrolemember 'ServerOrDomainName\Groupname', 'sysadmin'

   Enterprise Manager:
   Expand SQL Server name.
   Expand Security.
   Select Server Roles tab.
   Select System Administrators.
   Select Add.
   Select the 'ServerOrDomainName\Groupname'.
   Select OK.
   Select OK.

4. Remove the BUILTIN/Administrators group from SQL Server logons.
   (Do not perform this step until the previous three steps have been completed.)

   T-SQL:
   Exec sp_revokelogin ‘Builtin\administrators’

   Enterprise Manager:
   Expand SQL Server name.
   Expand Security.
   Select Server Roles tab.
   Select System Administrators.
   Click OK.
   From user list, right-click BUILTIN/Administrators.
   Select Delete.
   Select Yes.

F.4 File Permissions

- Set proper file permissions.

  Windows Explorer:
  Start/Run/explorer.exe.
  Browse to SQL Server install directory (\mssql).
  Right-click on install directory name.
  Select properties.
  Select security tab.
  Click on permissions button.

Set permissions for all SQLServer files and directories to be ONLY Full Control to the following:

1. Administrators
2. CREATOR OWNER
3. SYSTEM
4. SQL Server service account (custom name)
5. DBA Group (MSSQLSERVER\MSSQL_DBA)

Check Replace Permissions on existing files.
Check Replace Permissions on subfolders.
Repeat this procedure for each of the .mdf and .ldf database files.

F.5 Registry Permissions

- Check to ensure that registry permissions have been granted properly.

Windows REGEDT32:
Start/Run/regedt32.exe.
Select Window HKEY_LOCAL_MACHINE.
Expand Software.
Expand Microsoft.
Select Microsoft SQL Server (7.0 if applicable).
Click on Security from menu bar.
Select permissions.
Set permissions to full control for the following: (when updating permissions – check Replace Permissions on Existing subkeys).

1. Administrators
2. SYSTEM
3. SQL Server service account (custom name)
4. DBA group (MSSQLSERVER\MSSQL_DBA)
5. CREATOR OWNER

Repeat for \Software\Microsoft\Windows NT\CurrentVersion\Perflib with permissions set to Full Control granted to Administrators, CREATOR OWNER, and SYSTEM; *read* and *write* (Query Value, Set Value, Create Subkey, Enumerate Subkeys, Notify, WriteDAC, Write Owner, Read Control) permissions to the DBA group account (MSSQLSERVER\MSSQL_DBA), and SQL Server Service account (custom name).

Repeat for \System\CurrentControlSet\Services\MSSQLServer with permissions set to Full Control granted to Administrators, CREATOR OWNER, and SYSTEM; *read* and *write* (Query Value, Set Value, Create Subkey, Enumerate Subkeys, Notify, WriteDAC, Write Owner, Read Control permissions to the DBA group account (MSSQLSERVER\MSSQL_DBA); and SQL Server Service account (custom name).

Upon completion of these permission steps, configure the SQL Server services to use the service account.

Enterprise Manager:
Expand server.
Expand Management.
Right-click on SQLServer Agent.
Select Properties.
Select General tab.
Set Service startup account/This account.
Enter new service account name.
Click OK.

F.6 Audit Configuration

- Enable logon auditing.

T-SQL:
N/A

Enterprise Manager:
Right-click on SQL server name.
Select Properties.
Select Security tab.
Select Security/Audit level selection All or Failure.
Click OK.

- Enable the following audit events (SQL 2000 only).
NOTE: A custom trace file requires an extension for enabling SCAN FOR STARTUP PROCS. The required audit event selections are automatic if C2 audit mode is selected.

The required events to be audited are as follows:

ID	Event Name
14	Audit Logon Event
15	Audit Logout Event
18	Audit Server Starts and Stops Event
20	Logon Failed
102	Audit Statement GDR
103	Audit Object GDR
104	Audit Add/Drop Logon
105	Audit Logon GDR
106	Audit Logon Change Property
107	Audit Logon Change Password

108	Audit Add Logon to Server Role
109	Audit Add DB User
110	Audit Add Member to DB
111	Audit Add/Drop Role
112	App Role Pass Change
113	Audit Statement Permission
117	Audit Change Audit
118	Audit Object Derived Permission

T-SQL:

NOTE 1**:** Set SQL Server instance to scan for startup procedures:
Exec sp_configure ‘scan for startup procs’, ‘1’

NOTE 2: Create a custom trace procedure. The following file was created using SQL Profiler. For each audit event, include the following information in the audit data:

- Application name(10)
- SQL Logon name (11)
- Server process ID(12)
- Start time(14)
- End time(15)
- Event sub-class(21)
- Object ID(22)
- Success (23)
- Object type (28)
- Database name (35)
- Login SID (41)

NOTE 3: Audit trail data directed to a table requires that ALL object permissions on ALL objects be audited since MS SQL Server does not provide the granularity to audit a single object or use of particular privileges on objects.

```
CREATE PROCEDURE my_audit AS
   -- Create a Queue
   declare @rc int
   declare @TraceID int
   declare @maxfilesize bigint
   set @maxfilesize = 5

   exec @rc = sp_trace_create @TraceID output, 6,
     'c:\mssql8\mssql8$mssql8\data\my_audit', @maxfilesize, NULL
   if (@rc != 0) go to error
```

```
-- Client side File and Table cannot be scripted.
-- Set the events:
    declare @on bit
    set @on = 1
    exec sp_trace_setevent @TraceID, 18, 10, @on
    exec sp_trace_setevent @TraceID, 18, 11, @on
    exec sp_trace_setevent @TraceID, 18, 12, @on
    exec sp_trace_setevent @TraceID, 18, 14, @on
    exec sp_trace_setevent @TraceID, 18, 15, @on
    exec sp_trace_setevent @TraceID, 18, 21, @on
    exec sp_trace_setevent @TraceID, 18, 22, @on
    exec sp_trace_setevent @TraceID, 18, 23, @on
    exec sp_trace_setevent @TraceID, 18, 28, @on
    exec sp_trace_setevent @TraceID, 18, 35, @on
    exec sp_trace_setevent @TraceID, 18, 41, @on
    exec sp_trace_setevent @TraceID, 20, 10, @on
    exec sp_trace_setevent @TraceID, 20, 11, @on
    exec sp_trace_setevent @TraceID, 20, 12, @on
    exec sp_trace_setevent @TraceID, 20, 14, @on
    exec sp_trace_setevent @TraceID, 20, 15, @on
    exec sp_trace_setevent @TraceID, 20, 21, @on
    exec sp_trace_setevent @TraceID, 20, 22, @on
    exec sp_trace_setevent @TraceID, 20, 23, @on
    exec sp_trace_setevent @TraceID, 20, 28, @on
    exec sp_trace_setevent @TraceID, 20, 35, @on
    exec sp_trace_setevent @TraceID, 20, 41, @on
    exec sp_trace_setevent @TraceID, 102, 10, @on
    exec sp_trace_setevent @TraceID, 102, 11, @on
    exec sp_trace_setevent @TraceID, 102, 12, @on
    exec sp_trace_setevent @TraceID, 102, 14, @on
    exec sp_trace_setevent @TraceID, 102, 15, @on
    exec sp_trace_setevent @TraceID, 102, 21, @on
    exec sp_trace_setevent @TraceID, 102, 22, @on
    exec sp_trace_setevent @TraceID, 102, 23, @on
    exec sp_trace_setevent @TraceID, 102, 28, @on
    exec sp_trace_setevent @TraceID, 102, 35, @on
    exec sp_trace_setevent @TraceID, 102, 41, @on
    exec sp_trace_setevent @TraceID, 103, 10, @on
    exec sp_trace_setevent @TraceID, 103, 11, @on
    exec sp_trace_setevent @TraceID, 103, 12, @on
    exec sp_trace_setevent @TraceID, 103, 14, @on
    exec sp_trace_setevent @TraceID, 103, 15, @on
    exec sp_trace_setevent @TraceID, 103, 21, @on
    exec sp_trace_setevent @TraceID, 103, 22, @on
    exec sp_trace_setevent @TraceID, 103, 23, @on
    exec sp_trace_setevent @TraceID, 103, 28, @on
    exec sp_trace_setevent @TraceID, 103, 35, @on
    exec sp_trace_setevent @TraceID, 103, 41, @on
    exec sp_trace_setevent @TraceID, 104, 10, @on
    exec sp_trace_setevent @TraceID, 104, 11, @on
    exec sp_trace_setevent @TraceID, 104, 12, @on
    exec sp_trace_setevent @TraceID, 104, 14, @on
    exec sp_trace_setevent @TraceID, 104, 15, @on
    exec sp_trace_setevent @TraceID, 104, 21, @on
    exec sp_trace_setevent @TraceID, 104, 22, @on
    exec sp_trace_setevent @TraceID, 104, 23, @on
    exec sp_trace_setevent @TraceID, 104, 28, @on
    exec sp_trace_setevent @TraceID, 104, 35, @on
    exec sp_trace_setevent @TraceID, 104, 41, @on
    exec sp_trace_setevent @TraceID, 105, 10, @on
    exec sp_trace_setevent @TraceID, 105, 11, @on
    exec sp_trace_setevent @TraceID, 105, 12, @on
    exec sp_trace_setevent @TraceID, 105, 14, @on
    exec sp_trace_setevent @TraceID, 105, 15, @on
    exec sp_trace_setevent @TraceID, 105, 21, @on
    exec sp_trace_setevent @TraceID, 105, 22, @on
    exec sp_trace_setevent @TraceID, 105, 23, @on
    exec sp_trace_setevent @TraceID, 105, 28, @on
    exec sp_trace_setevent @TraceID, 105, 35, @on
    exec sp_trace_setevent @TraceID, 105, 41, @on
    exec sp_trace_setevent @TraceID, 106, 10, @on
    exec sp_trace_setevent @TraceID, 106, 11, @on
    exec sp_trace_setevent @TraceID, 106, 12, @on
    exec sp_trace_setevent @TraceID, 106, 14, @on
    exec sp_trace_setevent @TraceID, 106, 15, @on
    exec sp_trace_setevent @TraceID, 106, 21, @on
    exec sp_trace_setevent @TraceID, 106, 22, @on
    exec sp_trace_setevent @TraceID, 106, 23, @on
    exec sp_trace_setevent @TraceID, 106, 28, @on
    exec sp_trace_setevent @TraceID, 106, 35, @on
    exec sp_trace_setevent @TraceID, 106, 41, @on
    exec sp_trace_setevent @TraceID, 107, 10, @on
    exec sp_trace_setevent @TraceID, 107, 11, @on
    exec sp_trace_setevent @TraceID, 107, 12, @on
    exec sp_trace_setevent @TraceID, 107, 14, @on
    exec sp_trace_setevent @TraceID, 107, 15, @on
    exec sp_trace_setevent @TraceID, 107, 21, @on
    exec sp_trace_setevent @TraceID, 107, 22, @on
    exec sp_trace_setevent @TraceID, 107, 23, @on
    exec sp_trace_setevent @TraceID, 107, 28, @on
    exec sp_trace_setevent @TraceID, 107, 35, @on
    exec sp_trace_setevent @TraceID, 107, 41, @on
    exec sp_trace_setevent @TraceID, 108, 10, @on
    exec sp_trace_setevent @TraceID, 108, 11, @on
    exec sp_trace_setevent @TraceID, 108, 12, @on
    exec sp_trace_setevent @TraceID, 108, 14, @on
    exec sp_trace_setevent @TraceID, 108, 15, @on
    exec sp_trace_setevent @TraceID, 108, 21, @on
    exec sp_trace_setevent @TraceID, 108, 22, @on
    exec sp_trace_setevent @TraceID, 108, 23, @on
    exec sp_trace_setevent @TraceID, 108, 28, @on
    exec sp_trace_setevent @TraceID, 108, 35, @on
    exec sp_trace_setevent @TraceID, 108, 41, @on
    exec sp_trace_setevent @TraceID, 109, 10, @on
    exec sp_trace_setevent @TraceID, 109, 11, @on
    exec sp_trace_setevent @TraceID, 109, 12, @on
    exec sp_trace_setevent @TraceID, 109, 14, @on
    exec sp_trace_setevent @TraceID, 109, 15, @on
    exec sp_trace_setevent @TraceID, 109, 21, @on
    exec sp_trace_setevent @TraceID, 109, 22, @on
    exec sp_trace_setevent @TraceID, 109, 23, @on
    exec sp_trace_setevent @TraceID, 109, 28, @on
    exec sp_trace_setevent @TraceID, 109, 35, @on
    exec sp_trace_setevent @TraceID, 109, 41, @on
    exec sp_trace_setevent @TraceID, 110, 10, @on
    exec sp_trace_setevent @TraceID, 110, 11, @on
    exec sp_trace_setevent @TraceID, 110, 12, @on
    exec sp_trace_setevent @TraceID, 110, 14, @on
    exec sp_trace_setevent @TraceID, 110, 15, @on
    exec sp_trace_setevent @TraceID, 110, 21, @on
    exec sp_trace_setevent @TraceID, 110, 22, @on
    exec sp_trace_setevent @TraceID, 110, 23, @on
    exec sp_trace_setevent @TraceID, 110, 28, @on
    exec sp_trace_setevent @TraceID, 110, 35, @on
    exec sp_trace_setevent @TraceID, 110, 41, @on
    exec sp_trace_setevent @TraceID, 111, 10, @on
    exec sp_trace_setevent @TraceID, 111, 11, @on
    exec sp_trace_setevent @TraceID, 111, 12, @on
    exec sp_trace_setevent @TraceID, 111, 14, @on
    exec sp_trace_setevent @TraceID, 111, 15, @on
    exec sp_trace_setevent @TraceID, 111, 21, @on
    exec sp_trace_setevent @TraceID, 111, 22, @on
    exec sp_trace_setevent @TraceID, 111, 23, @on
    exec sp_trace_setevent @TraceID, 111, 28, @on
    exec sp_trace_setevent @TraceID, 111, 35, @on
    exec sp_trace_setevent @TraceID, 111, 41, @on
    exec sp_trace_setevent @TraceID, 112, 10, @on
    exec sp_trace_setevent @TraceID, 112, 11, @on
    exec sp_trace_setevent @TraceID, 112, 12, @on
    exec sp_trace_setevent @TraceID, 112, 14, @on
    exec sp_trace_setevent @TraceID, 112, 15, @on
    exec sp_trace_setevent @TraceID, 112, 21, @on
    exec sp_trace_setevent @TraceID, 112, 22, @on
    exec sp_trace_setevent @TraceID, 112, 23, @on
    exec sp_trace_setevent @TraceID, 112, 28, @on
    exec sp_trace_setevent @TraceID, 112, 35, @on
    exec sp_trace_setevent @TraceID, 112, 41, @on
    exec sp_trace_setevent @TraceID, 113, 10, @on
    exec sp_trace_setevent @TraceID, 113, 11, @on
    exec sp_trace_setevent @TraceID, 113, 12, @on
    exec sp_trace_setevent @TraceID, 113, 14, @on
    exec sp_trace_setevent @TraceID, 113, 15, @on
    exec sp_trace_setevent @TraceID, 113, 21, @on
    exec sp_trace_setevent @TraceID, 113, 22, @on
    exec sp_trace_setevent @TraceID, 113, 23, @on
    exec sp_trace_setevent @TraceID, 113, 28, @on
    exec sp_trace_setevent @TraceID, 113, 35, @on
    exec sp_trace_setevent @TraceID, 113, 41, @on
    exec sp_trace_setevent @TraceID, 115, 10, @on
    exec sp_trace_setevent @TraceID, 115, 11, @on
    exec sp_trace_setevent @TraceID, 115, 12, @on
    exec sp_trace_setevent @TraceID, 115, 14, @on
    exec sp_trace_setevent @TraceID, 115, 15, @on
    exec sp_trace_setevent @TraceID, 115, 21, @on
    exec sp_trace_setevent @TraceID, 115, 22, @on
    exec sp_trace_setevent @TraceID, 115, 23, @on
    exec sp_trace_setevent @TraceID, 115, 28, @on
    exec sp_trace_setevent @TraceID, 115, 35, @on
    exec sp_trace_setevent @TraceID, 115, 41, @on
    exec sp_trace_setevent @TraceID, 117, 10, @on
    exec sp_trace_setevent @TraceID, 117, 11, @on
    exec sp_trace_setevent @TraceID, 117, 12, @on
    exec sp_trace_setevent @TraceID, 117, 14, @on
    exec sp_trace_setevent @TraceID, 117, 15, @on
    exec sp_trace_setevent @TraceID, 117, 21, @on
    exec sp_trace_setevent @TraceID, 117, 22, @on
    exec sp_trace_setevent @TraceID, 117, 23, @on
    exec sp_trace_setevent @TraceID, 117, 28, @on
    exec sp_trace_setevent @TraceID, 117, 35, @on
    exec sp_trace_setevent @TraceID, 117, 41, @on
    exec sp_trace_setevent @TraceID, 118, 10, @on
    exec sp_trace_setevent @TraceID, 118, 11, @on
    exec sp_trace_setevent @TraceID, 118, 12, @on
    exec sp_trace_setevent @TraceID, 118, 14, @on
```

```
            exec sp_trace_setevent @TraceID, 118, 15, @on
            exec sp_trace_setevent @TraceID, 118, 21, @on
            exec sp_trace_setevent @TraceID, 118, 22, @on
            exec sp_trace_setevent @TraceID, 118, 23, @on
            exec sp_trace_setevent @TraceID, 118, 28, @on
            exec sp_trace_setevent @TraceID, 118, 35, @on
            exec sp_trace_setevent @TraceID, 118, 41, @on

        -- Set the Filters.
            declare @intfilter int
            declare @bigintfilter bigint

        -- Set the trace status to start.
            exec sp_trace_setstatus @TraceID, 1

        -- Display trace ID for future references.
            select TraceID=@TraceID
            goto finish

        Error:
          Select ErrorCode=@rc

          Finish:
            GO
            exec sp_procoption 'my_audit', 'startup', 'true'
            GO
```

Enterprise Manager:

NOTE: Scan for startup procs must be set in T-SQL.

Expand SQL Server name.
Expand Databases.
Expand Master database.
Right-click on Stored Procedures.
Select New Stored Procedure.
Paste in procedure as listed above under T-SQL.
Click OK.

- **Verify audit data protection (SQL Server 2000 only).**

  NOTE: For the auditing trace, rollover should be enabled and shutdown on failure should be enabled. This is automatic if c2 audit mode is selected.

  Review the trace SQL stored procedure script to set the option parameter specification to 6. This parameter is located directly after the word “output” in the sp_trace_create stored procedure call.

- Check to ensure that updates and deletes on the audit data is being audited.

For audit data stored in files use Windows Explorer:

NOTE: If a custom audit trace is being used, the file location and name to be verified for audit will be specified in the trace definition.

Start/Run/explorer.exe.
Browse to c:\winnt\system32\config\appevent.evt or to file as specified in trace output specification.
Right-mouse click on appevent.evt.
Select Properties.
Select Security tab.
Click Auditing button.
Click Add button.
Select Everyone group.
Click Add button.
Click OK.
Select audit for Success for delete, change permissions, and take ownership.
Select audit for Failure for execute, delete, change permissions, and take ownership.
Click apply.
Click OK.

For audit data stored in a **table**:

T-SQL:
NOTE: Replace 0 below with the trace id of the audit trace to review only audit trace information.

```
Declare @on bit
Set @on = 1
exec sp_trace_setevent TraceID, 114, 10, @on
exec sp_trace_setevent TraceID, 114, 11, @on
exec sp_trace_setevent TraceID, 114, 12, @on
exec sp_trace_setevent TraceID, 114, 14, @on
exec sp_trace_setevent TraceID, 114, 15, @on
exec sp_trace_setevent TraceID, 114, 21, @on
exec sp_trace_setevent TraceID, 114, 22, @on
exec sp_trace_setevent TraceID, 114, 23, @on
exec sp_trace_setevent TraceID, 114, 28, @on
exec sp_trace_setevent TraceID, 114, 35, @on
exec sp_trace_setevent TraceID, 114, 41, @on
```

F.7 Host Directory/Stored and Extended Procedure Access

- Deny access to specified stored procedures.

NOTE: No registry access procedures or office automation procedures should be available to non-DBA users. Either delete these procedures from the system or deny all permissions to these procedures to all users.

T-SQL:
NOTE: List all procedures to be protected or removed. Select name from sysobjects where name like 'xp_reg%' or name like 'sp_OA%'.

Deny execute on ProcedureName to PUBLIC;
OR
Drop procedure ProcedureName.

Enterprise Manager:
Expand SQL Server.
Expand Databases.
Expand Master database.
Select Extended Stored Procedures.
Scroll down the list of procedures.
For each procedure that begins with ‘sp_OA’ and ‘xp_reg’:
    Right-click on the procedure name.
    Select Delete or Properties.
    Select permissions.
    Select List only users/user and database roles/public with permissions to this object.
    Click on check box under EXEC column until a red X shows.
    Click Apply button.
    Click OK.
    Close and repeat for all procedures that begin with ‘sp_OA’ or ‘xp_reg’.

- Encrypt user-defined stored procedures.

T-SQL:
(Ensure you have source code available for all stored procedures before performing this task.)

First list all unencrypted user-defined stored procedures. This list does not include objects owned by the database owner.

```
Select sysobjects.name from sysobjects.
    inner join syscomments on sysobjects.id = syscomments.id
    where syscomments.encrypted = 0 and
    (sysobjects.type=’S’ or sysobjects.type=’X’)
    and sysobjects.uid >4 and sysobjects.uid<16384
```

For each stored procedure listed:
Alter procedure ProcedureName with encryption as ProcedureSQLStatements.

Enterprise Manager:
N/A

- Remove any user-defined extended procedures.

Enterprise Manager:
Expand SQL Server.
Expand Databases.
Expand Master database.
Select Extended Stored Procedures.
Scroll down the list of procedures.
For each procedure whose owner is not dbo:
Right-click on the procedure name.
Select Delete.
Click Yes.

Repeat for all user-defined extended stored procedures.

Repeat for all databases.

- Protect system-defined extended stored procedures from user access.

T-SQL:
List system extended stored procedures:

```
Select sysobjects.name, sysusers.name, sysprotects.action
    from sysprotects
    inner join sysobjects on sysobjects.id=sysprotects.id
    inner join sysusers on sysusers.uid=sysprotects.uid
    where sysobjects.type = 'X' and
    sysobjects.uid <5
```

For each procedure listed:
Deny execute on ProcedureName to public.

Enterprise Manager:
Expand SQL Server.
Expand Databases.
Expand Master database.
Select Extended Stored Procedures.
Scroll down the list of procedures.
Right-click on a procedure name.
Select All Tasks.
Select Manage Permissions.
For user public, click on check box under EXEC column until a red X shows.

Repeat for all user-defined extended stored procedures.

Repeat for all databases.

- Remove the extended procedure xp_cmdshell from the system.

T-SQL:
Drop procedure xp_cmdshell

Enterprise Manager:
Expand server name.
Expand databases.
Expand Master database.
Expand Extended Procedures.
Select and delete xp_cmdshell if listed.

F.8 Privileges to Assign Permissions

- Revoke any administrative privileges to objects from users.

T-SQL:
NOTE: List administrative privileges assigned to users.

```
Select sysusers.name, sysobjects.name, sysprotects.action from sysprotects
   inner join sysusers on sysusers.uid=sysprotects.uid
   inner join sysobjects on sysobjects.id=sysprotects.id
   where sysprotects.protecttype = 204
```

Revoke grant option for ObjectName from UserName.

Enterprise Manager:
N/A

F.9 Privileges Assigned to PUBLIC and Guest

- Remove the Guest account from all databases except master and tempdb.

  T-SQL:
  exec sp_dropuser ‘guest’

  repeat for all databases except Master and Tempdb

  Enterprise Manager:
  Expand server name.
  Expand Databases.
  Expand database name (except Master and tempdb).
  Select users.
  Right-click guest.
  Select delete.
  Click Yes.

  Repeat for all databases (except Master and tempdb)

- Remove statement permissions assigned to Public or Guest.

  T-SQL:
  Deny all to public
  Deny all to guest
  Repeat for each database

  Enterprise Manager:
  N/A

- Remove role assignments granted to Guest.

  T-SQL:
  EXEC sp_helpuser 'GUEST'

  For each role assignment listed:
  Exec sp_droprolemember ‘RoleName’, ‘Guest’

  Enterprise Manager:
  Expand server name.
  Expand Databases.
  Expand Master database.
  Select users.
  Double-click user Guest.
  Clear all checks except public.
  Click OK.

- Revoke object permissions granted to PUBLIC or Guest.

  T-SQL:
  List any permissions assignments:
  EXEC sp_helpuser 'PUBLIC'.
  EXEC sp_helpuser 'GUEST'

  For each object listed:
  Deny all on ObjectName to ‘PUBLIC’.
  Deny all on ObjectName to ‘Guest’.

  Enterprise Manager:
  Expand server name.
  Expand Databases.
  Expand database name (repeat for Master and tempdb).
  Select users.
  Right-click on Guest.
  Select Properties.
  Select Permissions.
  Select list only objects with permissions for this user.
  Clear any boxes that are checked.

  Repeat for Roles/Public.

  NOTE: Some permissions assigned to PUBLIC within the master database may require that the “Allow modifications to be made directly to the system catalogs” database setting be temporarily be enabled.

F.10 Predefined Role Assignments

- Remove any predefined server or fixed database roles assigned to non-DBAs.

  T-SQL:
  List users granted server role memberships:
  Exec sp_helpsrvrolemember

  For any unauthorized users:
  Exec sp_dropsrvrolemember ‘RoleName’,’UserName’

  List users granted database roles:
  For non-dba account granted a group role beginning with ‘db_’:
  Exec sp_droprolemember ‘RoleName’,’UserName’

Enterprise Manager:
Expand server.
Expand Security.
Select Server Roles.
Repeat for each role listed.
Double-click on server role.
Review users listed.
For each unauthorized user:
  Select name.
  Click Remove button.
  When all unauthorized users are removed click OK.

For fixed database roles:
  Expand server.
  Expand Databases.
  Expand database name.
  Select Roles.
  Double-click on each database role (begins with ‘db_’) listed to view users assigned.
  Select any unauthorized users and press Delete.

Repeat for each database.

- **Revoke DBA privileges from unauthorized users.**

T-SQL:
Review list of users granted ‘sysadmin’ server role.
EXEC sp_helpsrvrolemember ‘sysadmin’
For each unauthorized user:
Exec sp_droprolemember ‘sysadmin’, ‘UserName’

Enterprise Manager:
Expand server.
Expand Security.
Select Server Roles.
Right-click on sysadmin server role.
Review users listed.
For each unauthorized user:
Select name.
Click Remove button.
When all unauthorized users are removed, click OK.

F.11 Privileges Assigned to Users

- Revoke statement privileges assigned to users.

  T-SQL:
  List all assigned statement permissions:
  EXEC sp_helprotect NULL, NULL, NULL, 's'

  For each user listed:
  Revoke StatementPrivilege from UserName.

  Enterprise Manager:
  N/A

- Revoke any references privileges granted to application users or application roles.

  NOTE: Repeat for each database.

  T-SQL:

  ```
  Select sysusers.name, sysobjects.name, sysprotects.action from sysprotects
      Inner join sysobjects on sysobjects.id = sysprotects.id
      Inner join sysusers on sysusers.uid=sysprotects.uid
      where sysprotects.action=204
  ```

  For each object listed:
  Revoke references on ObjectName from UserName.

  Enterprise Manager:
  N/A

- Restrict Cmdexec and Active Scripting job step privileges to DBAs.

  T-SQL:
  N/A

  Enterprise Manager:
  Expand server.
  Expand Management.
  Right-click on SQLServer Agent.
  Select Properties.
  Select Job System tab.
  Select Non-SysAdmin job step proxy account/Only users with SysAdmin privileges can execute CmdExec and Active Scripting job steps.

- Deny access to DBA Views to users.

  T-SQL:

  ```
  Select sysobjects.name, sysprotects.uid from sysobjects:
     inner join sysprotects on sysobjects.id = sysprotects.id
     where (sysobjects.type = 'S' or  sysobjects.type = 'V') and
     sysobjects.uid >4
  For each permission listed:
  Deny all on ObjectName from UserOrRoleName.
  ```

- Secure backup files on disk.

  Determine backup location:

  T-SQL:

  To list drives where backup files are stored:
  Use msdb.
  Select physical_drive,physical_name from backup file.

  Secure the files:
  Windows Explorer:
  Start/Run/explorer.exe.
  Browse to backup directory and file listed from T-SQL.
  Right-click on directory.
  Select Properties.
  Select Security tab.
  Select Permissions.
  Restrict permissions to full control to:

  - SYSTEM
  - Administrators
  - SQL Server Agent service account
  - OS DBA group account
  - CREATOR OWNER

F.12 Application Roles

- Define Application administrator roles.

  T-SQL:
  Exec sp_addrole ‘RoleName’
  OR
  Exec sp_addapprole ‘RoleName’, ‘Password’

  Create application administrator roles for all of the systems applications.
  Create application user roles for all systems applications.

Enterprise Manager:
Expand SQL Server name.
Expand Databases.
Select the appropriate application database.
Right click on Roles.
Select New Database Roles.
Type in Role name.
Select Database Role.
Click OK.

F.13 Object Privileges

- Revoke object privileges granted directly to users.

T-SQL:
Exec sp_helprotect NULL, NULL, NULL, 'o'

For any user (not group) names listed:
Revoke Privilege on ObjectName from UserName.

Enterprise Manager:
Review by user in each database or by object in each database.

- Remove or change ownership of objects not owned by application accounts.

T-SQL:
List objects not owned by the database owner:
Select name, uid from sysobjects where uid <> 1

Delete object or change ownership to the application user:
Drop ObjectType ObjectName.
OR
Exec sp_changeobjectowner 'ObjectName,' 'Owner'.

Enterprise Manager:
Expand Server name.
Expand Databases.
Expand a database.
Select Tables.
Review Owner column for listed tables.
For each table not owned by dbo, select table and press Delete.
Repeat for Views, Stored procedures, Extended stored procedures, user defined data types, and user defined functions.
Repeat for each database.

- Disable application object owner accounts.

  T-SQL:
  Select sysusers.name, sysobjects.name from sysobjects:
  inner join sysusers on sysusers.uid=sysobjects.uid
  where sysusers.hasdbaccess=1 and sysusers.name<>'dbo'

  For each application (object) owner account:
  exec sp_denylogin 'UserName'

  Enterprise Manager:
  Expand SQL Server.
  Expand Security.
  Select Logins.
  Double-click on application owner account to disable.
  Select deny access.
  Click OK.

  Repeat for all application (object) owner accounts.

F.14 Linked or Remote Servers

- Configure Linked Server definitions to use current authentication.

  T-SQL:
  N/A

  Enterprise Manager:
  Expand server.
  Expand Security.
  Select Linked Servers.
  Double-click each linked server.
  Select Impersonate.
  Click OK.

F.15 Replication Security Configuration

- Configure Snapshot folder security.

  Valid value: The value should be an explicit share and not an administrative share; full control to Administrators, DBA group, CREATOR OWNER, SYSTEM; *read*/*write* to SQL Server service account.

  T-SQL:
  exec sp_browsesnapshotfolder

Enterprise Manager:
Expand SQL Server.
Expand Replication.
Expand Publications.
For each publication, right-click on publication.
Select Snapshot location tab.
Select generate snapshots in the following location.
Enter the path to a secure folder that is not an Administrative share (same security as other SQL Server directories).
De-select generate snapshots in the normal snapshot folder.
Click Apply.
Click OK.

- Verify Distributor Database security.

Enterprise Manager:
Select Tools from menu bar.
Select Replication.
Select Configure Publishing, Subscribers, and Distribution.
Select subscribers tab.
Double-click on each subscriber.
Under Agent connection to the subscriber:
Select Impersonate the SQL Server Agent account on SQL Server (trusted connection).
Click OK.

Repeat for each subscriber listed.

Repeat for each Publisher listed under Publisher tab.

Close.

F.16 Naming Standards

- Check SQL Server Instance Name (SQL Server 2000 only).

Instance names are defined at SQL Server installation and cannot be modified. A valid instance name will not contain any version specific identifiers.

T-SQL:
To display:
@@SERVERNAME

Enterprise Manager:
To display:
Select server.
Right-click on server.
Select Properties.
Select General tab.
View Name.

- Check database datafile allocations.

NOTE: Only empty datafiles can be removed from the database. Datafiles should not be shared by databases.

To view datafiles in use and their database assignments:

T-SQL:
Select name, filename from sysdatabases.

Enterprise Manager:
Expand server.
Expand Databases.
Expand Master.
Select tables.
Right-click on sysdatabases table.
Select Open Table.
Select Return all rows.
View database names and file names.

To add a datafile to a database:

T-SQL:
Alter database DatabaseName; add file FileSpecification.

APPENDIX G LIST OF ACRONYMS

ACL	Access Control List
ADP	Automated Data Processing
AIS	Automated Information Systems
APF	Authorized Program Facility
API	Application Program Interface
ASDC3I	Assistant Secretary of Defense for Command, Control, Communications, and Intelligence

C2	Level C Security for Computer Products (provides Discretionary Access Control [DAC]).
CA	Certificate of Authority
C&A	Certification and Accreditation
CGI	Common Gateway Interface
CICS	Customer Information Control System
CJCS	Chairman, Joint Chiefs of Staff
COTS	Commercial-Off-The-Shelf
CRT	Display Monitor (Cathode Ray Tube)
CSA	Command, Service, and Agency

DAA	Designated Approving Authority
DAC	Discretionary Access Control
DAD	Database Access Descriptor
DB	Database
DBA	Database Administrator
DCE	Distributed Computing Environment
DES	Data Encryption Standard
DFHSM	Data Facility Hierarchical Storage Manager
DII COE	Defense Information Infrastructure Common Operating Environment
DISA	Defense Information Systems Agency
DISAI	DISA Instruction
DITSCAP	DOD Information Technology Security Certification and Accreditation Process
DML	Data Manipulation Language
DO	Data Owner
DOD	Department of Defense
DODIG	Department of Defense Inspector General

EAS	Extended Assistance Support
EAL	Evaluated Assurance Level
E-mail	Electronic Mail
EMS	Extended Maintenance Support
ESAF	External Subsystem Attachment Facility

FSO	Field Security Operations
FTP	File Transfer Protocol

GOTS	Government-Off-The-Shelf

HLQ	High Level Qualifier
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transport Protocol
HTTPD	Hyper Text Transport Protocol Daemon
HTTPS	Secure Hyper Text Transport Protocol
I&A	Identification and Authentication
I&RTS	Integration and Run Time Specification
IAM	Information Assurance Manager
IAO	Information Assurance Officer
IASE	Information Assurance Support Environment
IAVA	Information Assurance Vulnerability Alert
IAVM	Information Assurance Vulnerability Management
IMS	IBM Hierarchical Database and Transaction Server
IMS/TM	IMS Transaction Manager
INFOCON	Information Operations Condition
IP	Internet Protocol
IT	Information Technology
MVS	Multiple Virtual Storage
NAC	National Agency Check
NIPRNet	Non-classified (but Sensitive) Internet Protocol Routing Network
NLS	National Language Support
NSO	Network Security Officer
OCI	Oracle Call Interface
ODBC	Open Database Connectivity
OFA	Optimal Flexible Architecture
Oracle9iAS	Oracle 9i Application Server
OS	Operating System
PDI	Potential Discrepancy Item
PDS	Partitioned Data Set
PL/SQL	Procedural Language/Structured Query Language
QA	Quality Assurance
RACF	Resource Access Control Facility
RDBMS	Relational Database Management System
RMAN	Recovery Manager
RNOSC	Regional Network Operations And Security Center
ROSC	Regional Operations and Security Center
SA	System Administrator
SAP	Special Access Program
SCI	Secure Compartmented Information
SGA	System Global Area
SID	System Identifier

SIPRNet	Secret Internet Protocol Router Network
SM	Security Manager
SMF	System Management Facility
SQL	Structured Query Language
SQL*Plus	Low Level User Interface to Oracle DBMS
SRR	Security Readiness Review
SRRDB	SRR Database
SSBI	Single Scope Background Investigation
SSL	Secure Sockets Layer
SSN	Subsystem Name
SSO	Systems Support Office
STIG	Security Technical Implementation Guide
STC	Started Task Name
SU	Switch User (UNIX)
TASO	Terminal Area Security Officer
TCB	Trusted Computing Base
TNS	Transparent Network Substrate (SQL*NET Configuration)
TSO	Time Sharing Option
Umask	Command to display or change a user’s file permissions mask (UNIX)
URL	Uniform Resource Locator
VSAM	Virtual Storage Access Method
WAS	Web Application Server (now called Oracle9iAS – Oracle 9i Application Server)
WDB	Web DB (now called Oracle Portal)
WRB	Web Request Broker
WRBX	Web Application Execution Instance
WWW	World Wide Web

.CTL	Control File (Extension)
.DBF	Database File (Extension)
.GOV	World Wide Web Uniform Resource Locator Domain for the U.S. Government
.IDX	Index File (Extension)
.LOG	Log File (Extension)
.MIL	World Wide Web Uniform Resource Locator Domain for the U.S. Military

This page is intentionally left blank.