



**Enhanced  
Automated  
Graphical  
Logistics  
Environment**

**EAGLE SECURITY MANUAL  
VERSION 5**

**1 July 2001**

**A Product of Raytheon Company**  
1997-2001 Raytheon Company  
ALL RIGHTS RESERVED  
U.S. Patents 5,457,792; 5,493,679; 5,737,532  
4,847,795  
Made in the U.S.A.

COPYRIGHT 1997-2001 RAYTHEON COMPANY  
UNPUBLISHED WORK - ALL RIGHTS RESERVED.

This document contains proprietary information, and, except with written permission of Raytheon Company, such information shall not be published, or disclosed to others, or duplicated in whole or in part.

All other company and product names used herein may be the trademarks or registered trademarks of their respective companies.

Information in this manual may change without notice and does not represent a commitment on the part of Raytheon Company and its subsidiaries.

#### Revision History

Printed April 2001 First Edition

Printed July 2001 First Edition, Revised

Printed July 2001



The EAGLE Software Package has become the best of its kind thanks, in large part, to its clients. We would like to take this opportunity to thank all of you for your suggestions, insights and support. In addition, we want to renew our commitment to you, our valued clients.



***TABLE  
OF  
CONTENTS***

---



---

**TABLE OF CONTENTS**

<b>SECTION 1 THEORY OF OPERATION</b> .....	<b>1-3</b>
1.0 INTRODUCTION.....	1-3
1.1 SECURITY NEEDS IDENTIFIED .....	1-3
1.2 PRESUMPTIONS.....	1-4
1.3 USER CLASSIFICATION .....	1-4
1.4 BASIC CONCEPTS.....	1-5
1.4.1 Standard ORACLE Objects .....	1-5
1.4.2 User Connection Process.....	1-10
1.5 EAGLE SECURITY MANAGER .....	1-15
1.5.1 EAGLE Security Manager Overview.....	1-15
1.5.2 Random Password Generation and Encryption.....	1-20
1.5.2.1 The password encryption on Oracle works as follows.....	1-20
1.5.2.2 Randomize operates as follows: .....	1-20
1.5.3 ESM Data Access and Control.....	1-20
1.5.3.1 Accessing Data.....	1-22
1.5.3.2 Updating, Inserting, and Deleting Data.....	1-22
1.6 AUDITING USER ACTIONS.....	1-28
1.7 SECURITY EXAMPLES .....	1-28
<b>SECTION 2 EAGLE SECURITY MAINTENANCE</b> .....	<b>2-3</b>
2.0 INTRODUCTION.....	2-3
2.1 ADDING USERS.....	2-4
2.2 GRANTING PRIVILEGES TO EXISTING USER ACCOUNTS .....	2-6
2.3 REVOKING PRIVILEGES FROM USERS.....	2-7
2.4 DELETING USER ACCOUNTS .....	2-9
2.5 CLONING USER ACCOUNTS .....	2-10
2.6 CUSTOMIZING THE NAVIGATOR FOR INDIVIDUAL USER ACCOUNTS .....	2-12
2.7 ALTERING USER ACCOUNTS .....	2-14
2.8 VIEWING USER ROLES .....	2-16
2.9 UNLOCKING USER ACCOUNTS .....	2-18
2.10 VIEWING DATABASE AUDIT TRAIL INFORMATION .....	2-19
2.11 ENABLING/DISABLING CHANGE LOGGING .....	2-22
<b>INDEX</b> .....	<b>3</b>

---

---

**NOTES** .....3

**NOTES** .....4

**NOTES** .....5

**NOTES** .....6



***LIST  
OF  
ILLUSTRATIONS***

---



---

**LIST OF ILLUSTRATIONS**

<b><u>FIGURE</u></b>	<b><u>PAGE</u></b>
Figure 1 Profiles.....	1-11
Figure 2 SECURE_EAGLE Function (Sheet 1 of 3).....	1-12
Figure 3 Example Role Assignment - EAGLE_USER_ROLE.....	1-16
Figure 4 Example Role Assignment - EAGLE_SA.....	1-17
Figure 5 Login Creation Script for EAGLE login.....	1-19
Figure 6 Login Creation Script for EAGLE user.....	1-19
Figure 7 Sample SQL for VIEW Creation.....	1-21
Figure 8 Example INSTEAD OF DELETE Trigger.....	1-23
Figure 9 Example INSTEAD OF INSERT Trigger (Sheet 1 of 2).....	1-24
Figure 10 Example INSTEAD OF UPDATE Trigger (Sheet 1 of 2).....	1-26
Figure 11 Accessing User Security Maintenance From the Navigator.....	2-3
Figure 12 User Security Maintenance Window.....	2-4
Figure 13 Create New User Window.....	2-5
Figure 14 Granting Privileges to Existing User Accounts.....	2-6
Figure 15 Revoking Privileges From Users.....	2-7
Figure 16 Deleting User Accounts.....	2-9
Figure 17 User Security Maintenance Window - Cloning User Accounts.....	2-10
Figure 18 Clone User Window.....	2-11
Figure 19 User Security Maintenance Window - Customizing the Navigator.....	2-12
Figure 20 Modify User Menu Window.....	2-13
Figure 21 User Security Maintenance Window- Altering User Accounts.....	2-14
Figure 22 Alter User Window.....	2-15
Figure 23 User Security Maintenance Window - Viewing User Roles.....	2-16
Figure 24 User Roles Window.....	2-17
Figure 25 User Security Maintenance Window - Unlocking User Accounts.....	2-17
Figure 26 Accessing The Database Audit Trail From the Navigator.....	2-18
Figure 27 Navigator - Database Audit Trail.....	2-19
Figure 28 Database Audit Trail Information Window.....	2-20
Figure 29 Database Audit Trail Change Information.....	2-21
Figure 30 Database Audit Trail Information Window.....	2-22
Figure 31 System Defaults Window.....	2-23





***LIST  
OF  
TABLES***

---



---

**LIST OF TABLES**

<b><u>TABLE</u></b>	<b><u>PAGE</u></b>
Table 1 Security Related Objects .....	1-6
Table 2 Table ZID. Primary Security Table.....	1-7
Table 3 Table ZUID. Eagle Userid Table.....	1-7
Table 4 EAGLE Security System Data Element Definitions .....	1-8



# ***SECTION 1***



# ***THEORY OF OPERATION***

---



## SECTION 1 THEORY OF OPERATION

### 1.0 INTRODUCTION

The EAGLE Security System (ESS) is a product that functions together with EAGLE 4.0 or later and ORACLE 8.0.5 or later to provide improved data security for an installation. It is not a replacement for standard ORACLE RDBMS object oriented security. Rather, it is an extension of standard ORACLE Security by taking advantage of standard objects such as views, roles, profiles, privileges, etc.

The prime directive in establishing a security system for the EAGLE LSAR system is to reduce the likelihood of inadvertent (or deliberate) data exposure, including unauthorized access, modification, or destruction of data. And, at the same time, insure that the information stored in the system is easily accessible by authorized users.

### 1.1 SECURITY NEEDS IDENTIFIED

To meet the prime directive the following needs have been identified as being required to establish a security system.

- Identify and verify users
- Authorize users to access the protected resources
- Control the means of access to resources
- Audit users actions

## 1.2 PRESUMPTIONS

EAGLE Security is based upon basic ORACLE RDBMS standard code, such as views, roles, profiles, privileges, etc. Therefore, EAGLE Security is an extension of basic ORACLE standard security and not a replacement for ORACLE Security Server, Trusted ORACLE, etc. As such, the EAGLE Security discipline is not a replacement for the ORACLE Security Manager or other third party products that work with basic ORACLE standard security but is intended to allow a simplified approach to EAGLE Security Management. Organizations using the security system must establish proper managerial controls to ensure the system is not abused.

## 1.3 USER CLASSIFICATION

Four classifications of users are authorized varying degrees of access to the EAGLE system. Each of these is defined below along with the default password. The userid entered at the login screen is shown in parentheses – e.g., the userid entered for Database Administrator is EAGLE. It is recommended that the default passwords for EAGLE, EAGLESA, and SU be changed immediately by the Security Administrator. Also, passwords for sample logins provided by Oracle (such as login of ‘SCOTT’, with password ‘tiger’) should be changed or the login account deleted.

- Database Administrator

The Database Administrator (EAGLE) is a special userid. Logging on with this userid gives one the ability to create, modify, and/or delete database objects, act as the Security Administrator and work with the data in the database irregardless of EAGLE Security or ORACLE standard security. This userid cannot be deleted using the ESS. The default password for the database administrator can be obtained from the EAGLE help desk.

- Security Administrator

The Security Administrator (EAGLESA) is another special userid. Using this id allows one to create users on the system. It does not have DBA authority, cannot manipulate any data in the LSAR tables and cannot use the EAGLE disciplines. The security administrator has the responsibility to define users and resources to ESS. The security administrator can define and grant the authorities by which users access the protected resources. Thus, the security administrator sets down the guidelines that ESS uses to decide the user-resource interaction within the installation. ESS retains information about the users, resources, and access authorities in profiles on the EAGLE database and refers to the profiles when deciding which users should be permitted access to EAGLE resources. This userid cannot be deleted using the ESS. The default password for the security administrator can be obtained from the EAGLE help desk.

- Superuser

The Superuser (SU) is another special userid. This super userid allows one to get around the EAGLE Security System, to a certain extent. The super userid will not have DBA authority, and will not have Security Administrator authority. It will allow a user to manipulate any data in the EAGLE ORACLE LSAR database regardless of table/row ownership. This userid cannot be deleted by the ESS. The default password for the superuser can be obtained from the EAGLE help desk.

- Authorized users

The “authorized user” userid is the user for which the system is intended to protect and serve. This userid will be able to access the EAGLE disciplines and have access to specific portions of the data stored on the database as predetermined by proper management and implemented by the Security Administrator. EAGLE uses a user ID to identify the person who is trying to gain access to the system and a password to then verify the authenticity of that identity. EAGLE uses the concept of only one person knowing a particular user ID – password combination to verify user identities and to ensure personal accountability. This is accomplished through the CREATE USER utility provided by the basic ORACLE RDBMS. EAGLE supplies a userid, EAGLEID, to use as a guide. This userid can be deleted. The default password for EAGLEID can be obtained from the EAGLE help desk.

## 1.4 BASIC CONCEPTS

The EAGLE Security System consists of standard ORACLE objects (i.e. tables, views, triggers, and synonyms) combined with a specially designed application called the EAGLE Security Manager. It controls access at the row level by comparing two columns on each table to a users profile in an authorization table.

### 1.4.1 Standard ORACLE Objects

Prior to Oracle8, an object was a term used to identify anything in the database but a user. With Oracle8 embracing the concept on an object-oriented database, the term object can be confused with relational objects, database objects, and object types. Objects in this document refer to anything in the database data dictionary such as tables, views, triggers and synonyms. Each type of object, related to security, is defined in Table 1.

Table 1 Security Related Objects

Object	Description
Function	A function is a block of PL/SQL code that performs a predetermined operation.
Password	A set of characters that you must enter when you connect to your host computers operating system or to an Oracle database.
Profile	A collection of settings in Oracle that limit database resources.
Role	A set of privileges that a user can grant to another user.
Synonym	A name assigned to a table or view that may thereafter be used to refer to it.
Table	A table is the basic data storage structure. A table consists of one or more units of information (rows), each of which contains the same kinds of values (columns).
Trigger	A stored procedure associated with a table that Oracle automatically executes on one or more specified events affecting the table.
User id	A word that identifies you as an authorized user of your host computer's operating system or Oracle. Associated with each user id is a password.
View	A logical representation of a table. It is derived from a table but has no storage of its own and often may be used in the same manner as a table.

The basic MIL-STD 1388-2B database has 104 tables; each defined with a set of key values to ensure there are no duplicate rows stored in any of the tables. There are two predominate key sets established in the database. The first consists of EIACODXA, LSACONXB, ALTLCNXB, and LCNTYPXB. The second is CAGECDXH and REFNUMHA. There are tables with both of these key sets combined as well as a few other unique key sets.

To secure data at the row level a common denominator is needed across all tables. To accomplish this, two modifications to the 2B standard have been made. First, the column EIACODXA has been added to each table as part of the tables key set of columns for all tables where EIACODXA was not originally included in the 2B standard. This gives EAGLE the flexibility to not only group all elements in the database to a specific end item, but to allow certain end users access to specific end item related data. Second, a column has been added to each table in the database to show "ownership". Data stored in the column, USERIDZU, is organizationally oriented. That is, USERIDZU does not indicate a specific userid; rather, it indicates a grouping of users within an organization. This gives ESS the flexibility to permit access to data within an EIACODXA to a group of users while denying access to the same EIACODXA to other groups of users. Consequently, prime contractor and sub contractor data can be segregated in a common database.

In addition to the modifications to existing tables described, a set of tables has been designed to store EAGLE user profiles. These tables are detailed in Table 2, Table 3, and Table 4. Table ZID is the primary security table and contains the primary user identification code of the user. The

primary user identification code, column ZOID is the identification code assigned to each user. This is the id that the user will enter when prompted by the EAGLE login screen. It contains the assigned EAGLE identification code (internal EAGLE user id), the password to the internal user id, and optional data about the user, such as name, organization, address, and phone number. Table ZUID defines which user has access to which end item, which organizations data they can view and which organizations data they can update.

Table ZUID is joined to each of the EAGLE tables, separately, in a view. These views then control which data the user is allowed to see. However, this is not necessarily the data a user can update. Oracle does not allow direct updates (insert, update, and/or deletes) to a table through a view containing a join. Instead, Oracle provides the ability to utilize INSTEAD OF triggers. These triggers fire in place of the insert, update, or delete statement that the user enters against the table. Through these triggers, the system enforces a different set of rules for updates to the database. This allows different access for select (read-only) and update.

Table 2 Table ZID. Primary Security Table.

This table contains the primary loginid of the end user and the matching Eagle Identification/password as well as various options. The primary key is ZOID thus ensuring a unique 1 to 1 relationship between an Oracle ID versus and EAGLE ID.

Code	Data Element Title	Format	DED	KEY
ZOID	Oracle Identification Code	29 X L	SEC001	K
ZEID	Eagle Identification Code	30 X L	SEC002	M
ZPID	Eagle Password	24 X L	SEC003	M
ZNMEID	User Name	255 X L	SEC006	
ZORGID	User Organization	255 X L	SEC007	
ZLOCID	User Location	255 X L	SEC008	
ZPHNID	User Phone	255 X L	SEC009	

Table 3 Table ZUID. Eagle Userid Table.

This table identifies which end items and which Ownership ids within the end item an EAGLE user may access. There is no relational integrity between this table and Table ZID.

Code	Data Element Title	Format	DED	KEY
ZEIDUID	Eagle Identification Code	30 X L	SEC002	K
ZEIACUID	End Item Acronym Code	10 X L	SEC004	K
ZRFIDUID	Team Ownership Identification Code	30 X L	SEC005	M
ZSRFIDUID	Select Team Ownership Identification Code	30 X L	SEC005	M

Table 4 EAGLE Security System Data Element Definitions

DED	Title/Description	Format
SEC001	<p>ORACLE IDENTIFICATION CODE</p> <p>This is the authorized id that each user will have assigned to them to access the database.</p>	29 X L
SEC002	<p>EAGLE IDENTIFICATION CODE</p> <p>This is the id automatically assigned to authorized users. It is the same id as the ORACLE IDENTIFICATION CODE with the addition of an underscore ‘_’ added as the last position.</p>	30 X L
SEC003	<p>EAGLE Password</p> <p>This is the password for the EAGLE IDENTIFICATION CODE. It is an encrypted value.</p>	24 X L
SEC004	<p>END ITEM ACRONYM CODE</p> <p>This is the End Item the user is authorized to access.</p>	10 X L
SEC005	<p>TEAM OWNERSHIP IDENTIFICATION CODE/ SELECT TEAM OWNERSHIP IDENTIFICATION CODE</p> <p>This column shows row ownership. Proper ownership combined with the EIAC (SEC004) will allow updates to data. Proper select ownership combined with the EIAC (SEC004) will allow select or read only access to data.</p>	30 X L

Table 4 EAGLE Security System Data Element Definitions (Continued)

SEC006	USER NAME The EAGLE User's name.	255 X L
SEC007	USER ORGANIZATION The users organization or company.	255 X L
SEC008	USER LOCATION The users location, building, city, country, etc.	255 X L
SEC009	USER PHONE The users phone number.	255 X L

### 1.4.2 User Connection Process

When a user logs on to the EAGLE system, the userid and password information, are passed to ORACLE for account authentication (user identification and verification). During account authentication, Oracle checks for the proper id and password combination. ESS uses the concept of only one person knowing a particular user id/password combination to verify user identities and to ensure personal accountability. Then Oracle checks that the password standards are met.

Password standards are set through the use of the Oracle object PROFILES. The password standards are those parameters beginning with PASSWORD. These standards are set in EAGLE through the EAGLE\_LOGIN\_PROFILE shown in Figure 1. The PASSWORD\_VERIFY\_FUNCTION parameter is the only one of these we will discuss here. The rest of the descriptions of these parameters can be found in the documentation provided with the Oracle RDBMS product.

The PASSWORD\_VERIFY\_FUNCTION allows the installation site to control the complexity of the password through a SQL function script. This script is based on the default script that Oracle supplies, UTLPWDMG.SQL. The Oracle function has been modified and renamed to SECURE\_EAGLE. The SECURE\_EAGLE function is user modifiable. If you choose to modify the SECURE\_EAGLE function, you must be logged on with the ORACLE userid of SYS. The script is contained in Figure 2. Its salient features are:

- Password cannot be the same as the userid
- Password must be at least 6 positions in length
- Password cannot be greater than 8 positions in length
- Password must contain at least one digit and one character. Note that Oracle requires passwords to begin with a character (A-Z)
- When creating a new password it must differ from the old password by at least three characters

When a successful login has occurred and control has been passed back to EAGLE, the system then uses the user id to find a corresponding entry in table ZID. The system retrieves the EAGLE Identification Code and the EAGLE Password for this match, disconnects from the database and reconnects to the Oracle server with the new id and password. Once connected to the database the user is able to perform their tasks. This feature locks the database down in such a manner as to disallow users from accessing EAGLE except through the EAGLE disciplines. The only exceptions to this are the EAGLE provided userids, EAGLE, EAGLESA and SU, and any userids created by the DBA not utilizing the EAGLE Security Manager.

/\* This profile is assigned to each user as the loginid used to access EAGLE. \*/

```
CREATE PROFILE EAGLE_LOGIN_PROFILE
LIMIT SESSIONS_PER_USER          1
  CPU_PER_SESSION                DEFAULT
  CPU_PER_CALL                   DEFAULT
  CONNECT_TIME                   1
  IDLE_TIME                      DEFAULT
  LOGICAL_READS_PER_SESSION     DEFAULT
  LOGICAL_READS_PER_CALL       DEFAULT
  COMPOSITE_LIMIT               DEFAULT
  PRIVATE_SGA                   DEFAULT
  FAILED_LOGIN_ATTEMPTS        3
  PASSWORD_LIFE_TIME            30
  PASSWORD_REUSE_TIME           365
  PASSWORD_REUSE_MAX            UNLIMITED
  PASSWORD_LOCK_TIME            1/24
  PASSWORD_GRACE_TIME           15
  PASSWORD_VERIFY_FUNCTION      SECURE_EAGLE;
```

```
CREATE PROFILE EAGLE_USER_PROFILE
LIMIT SESSIONS_PER_USER          DEFAULT
  CPU_PER_SESSION                DEFAULT
  CPU_PER_CALL                   DEFAULT
  CONNECT_TIME                   DEFAULT
  IDLE_TIME                      30
  LOGICAL_READS_PER_SESSION     DEFAULT
  LOGICAL_READS_PER_CALL       DEFAULT
  COMPOSITE_LIMIT               DEFAULT
  PRIVATE_SGA                   DEFAULT
  FAILED_LOGIN_ATTEMPTS        DEFAULT
  PASSWORD_LIFE_TIME            DEFAULT
  PASSWORD_REUSE_TIME           UNLIMITED
  PASSWORD_REUSE_MAX            UNLIMITED
  PASSWORD_LOCK_TIME            1
  PASSWORD_GRACE_TIME           DEFAULT
  PASSWORD_VERIFY_FUNCTION      NULL;
```

Figure 1 Profiles

## CREATE OR REPLACE FUNCTION SECURE\_EAGLE

```
(username varchar2,
 password varchar2,
 old_password varchar2)
RETURN boolean IS
  n boolean;
  m integer;
  differ integer;
  isdigit boolean;
  ischar boolean;
  ispunct boolean;
  digitarray varchar2(20);
  punctarray varchar2(25);
  chararray varchar2(52);

BEGIN
  digitarray:= '0123456789';
  chararray:= 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
  punctarray:= '!"#$%&()``*+,-/;<=>?_';

  -- Check if the password is same as the username
  IF password = username THEN
    raise_application_error(-20915, 'Password same as user');
  END IF;

  -- Check for the minimum length of the password
  IF length(password) < 6 THEN
    raise_application_error(-20915, 'Password length less than 6 characters. ');
  END IF;

  -- Check for the maximum length of the password
  IF length(password) > 8 THEN
    raise_application_error(-20915, 'Password length greater than 8 characters. ');
  END IF;
```

Figure 2 SECURE\_EAGLE Function (Sheet 1 of 3)

```
-- Check if the password contains at least one letter and one digit
-- 1. Check for the digit
isdigit:=FALSE;
m := length(password);
FOR i IN 1..10 LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(digitarray,i,1) THEN
      isdigit:=TRUE;
      GOTO findchar;
    END IF;
  END LOOP;
END LOOP;
IF isdigit = FALSE THEN
  raise_application_error(-20915, 'Password should contain at least one digit, and one
character.');
```

```
END IF;
-- 2. Check for the character
<<findchar>>
ischar:=FALSE;
FOR i IN 1..length(chararray) LOOP
  FOR j IN 1..m LOOP
    IF substr(password,j,1) = substr(chararray,i,1) THEN
      ischar:=TRUE;
    END IF;
  END LOOP;
END LOOP;
IF ischar = FALSE THEN
  raise_application_error(-20915, 'Password should contain at least one \
digit, and one character.');
```

```
END IF;

<<endsearch>>
-- Check if the password differs from the previous password by at least
-- 3 letters
IF old_password = " THEN
  raise_application_error(-20915, 'Old password is null!');
END IF;
-- Everything is fine; return TRUE ;
differ := length(old_password) - length(password);
```

Figure 2 SECURE\_EAGLE Function (Sheet 2 of 3)

```
IF abs(differ) < 3 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 < 3 THEN
    raise_application_error(-20915, 'Password should differ by at \
      least 3 characters');
  END IF;
END IF;
-- Everything is fine; return TRUE ;
RETURN(TRUE);
END;
```

Figure 2 SECURE\_EAGLE Function (Sheet 3 of 3)

## 1.5 EAGLE SECURITY MANAGER

### 1.5.1 EAGLE Security Manager Overview

The purpose of the EAGLE Security Manager (ESM) is to allow EAGLE user accounts to be created and maintained. It does not allow for the creation/modification of roles and the assignment or removal of privileges to these roles or other Oracle related functions such as the creation of profiles. Functions such as these are provided by other Oracle products and third party add-ons. As part of the EAGLE installation, login profiles, roles, and privileges assigned to the USER, SA, REPORTS, and READONLY roles are provided. Examples of profiles are shown in Figure 1 and example roles with the assignment of privileges are shown in Figure 3 and Figure 4.

To help explain the manner in which ESM establishes authorization and controls access, a scenario to create a userid has been created below.

The Security Manager or Administrator (SA) will log on to EAGLE and select the EAGLE Security Management Discipline. The first task that ESM does is to verify the userid of the person accessing ESM. The userid must be EAGLE or EAGLESA. If the userid is not one of these two then ESM will terminate and bring the user back to the EAGLE Navigator with the error '-20915 9999. Security Violation' displayed.

ESM will then prompt the SA for the following information, (See Table 2 and Table 4 for definitions).

ZOID	This would be a required entry by the SA.
ZNMEID	This would be an optional entry by the SA.
ZORGID	This would be an optional entry by the SA.
ZLOCID	This would be an optional entry by the SA.
ZPHNID	This would be an optional entry by the SA.

```
CREATE ROLE EAGLE_USER_ROLE NOT IDENTIFIED;
GRANT EAGLE_USER_ROLE TO EAGLE WITH ADMIN OPTION;
GRANT EAGLE_USER_ROLE TO SU WITH ADMIN OPTION;
GRANT CREATE SESSION TO EAGLE_USER_ROLE;
GRANT EXECUTE ON EAGLE.AB_PACK TO EAGLE_USER_ROLE;
GRANT EXECUTE ON EAGLE.DED_001 TO EAGLE_USER_ROLE;
GRANT EXECUTE ON EAGLE.UP_AA_COPY TO EAGLE_USER_ROLE;
GRANT EXECUTE ON EAGLE.UP_AA_DEL TO EAGLE_USER_ROLE;
GRANT INSERT ON EAGLE.VIEW_AA TO EAGLE_USER_ROLE;
GRANT UPDATE ON EAGLE.VIEW_AA TO EAGLE_USER_ROLE;
GRANT DELETE ON EAGLE.VIEW_AA TO EAGLE_USER_ROLE;
GRANT SELECT ON EAGLE.VIEW_AA TO EAGLE_USER_ROLE;
```

```
      . . .
      . . .
      . . .
      . . .
```

Note: This example role is assigned as the default to each userid that is given the profile EAGLE\_USER\_PROFILE. EXECUTE permission must be granted on all packages, procedures and functions owned by EAGLE. INSERT, UPDATE, DELETE, SELECT permission must be granted on views and tables owned by EAGLE. In addition SELECT permission need to be granted on V\$SESSION and V\$PROCESS by user SYS to the role and the ANALYZE ANY system privilege must be permitted.

Figure 3 Example Role Assignment - EAGLE\_USER\_ROLE

```
CREATE ROLE EAGLE_SA NOT IDENTIFIED;  
GRANT EAGLE_SA TO CONNECT;  
GRANT EAGLE_SA TO EAGLE WITH ADMIN OPTION;  
GRANT EAGLE_SA TO EAGLESA WITH ADMIN OPTION;  
GRANT EAGLE_SA TO RESOURCE;  
GRANT ALTER USER TO EAGLE_SA;  
GRANT CREATE SESSION TO EAGLE_SA;  
GRANT CREATE USER TO EAGLE_SA;  
GRANT DROP USER TO EAGLE_SA;  
GRANT INSERT ON EAGLE.ZID TO EAGLE_SA;  
GRANT SELECT ON EAGLE.ZID TO EAGLE_SA;  
GRANT UPDATE ON EAGLE.ZID TO EAGLE_SA;  
GRANT DELETE ON EAGLE.ZID TO EAGLE_SA;  
GRANT SELECT ON EAGLE.ZMENU TO EAGLE_SA;  
GRANT SELECT ON EAGLE.ZSYSTEM TO EAGLE_SA;  
GRANT DELETE ON EAGLE.ZUID TO EAGLE_SA;  
GRANT INSERT ON EAGLE.ZUID TO EAGLE_SA;  
GRANT SELECT ON EAGLE.ZUID TO EAGLE_SA;  
GRANT UPDATE ON EAGLE.ZUID TO EAGLE_SA;  
GRANT DELETE ON EAGLE.ZUSER TO EAGLE_SA;  
GRANT INSERT ON EAGLE.ZUSER TO EAGLE_SA;  
GRANT SELECT ON EAGLE.ZUSER TO EAGLE_SA;  
GRANT UPDATE ON EAGLE.ZUSER TO EAGLE_SA;
```

Figure 4 Example Role Assignment - EAGLE\_SA

From this information, ESM will perform the following steps, in order.

- Build and execute the EAGLE login script (see Figure 5) by automatically assigning the value of ZOID to the CREATE USER, IDENTIFIED BY, and GRANT sections of the script.
- Automatically create ZEID by appending an underscore ‘\_’ as the last character of ZOID.
- Randomly create a password and encrypt it (see paragraph 1.5.2) and store it in ZPID.
- Build and execute the EAGLE user script (see Figure 6) after automatically assigning the value of ZEID to the CREATE USER, GRANT and ALTER sections of the script. The value generated for ZPID is used in the IDENTIFIED BY section of the script.
- Insert a row into ZID.
- ESM will then prompt the SA for the following information, (See Table 3 and Table 4 for definitions) and once supplied with the data, populate table ZUID. Paragraph 1.7 explains various scenarios of teaming (ZRFIDUID) and select teaming (ZSRFIDUID) assignments.

ZEIDUID	Only value not prompted for. ESM will automatically assign this from ZEID in ZID.
ZEIACUID	ESM will present a list to select from, derived from ‘Select EIACODXA from XA’ if available, otherwise it will allow the SA to enter a value. This enables the SA to setup a user prior to putting any data in the database.
ZRFIDUID	ESM will present a list to select from, derived from ‘Select distinct zrfiduid from zuid’ otherwise it will allow the SA to enter a value. This enables the SA to setup a user prior to putting any data in the database.
ZSRFIDUID	ESM will present a list to select from, derived from ‘Select distinct zrfiduid from zuid’ otherwise it will allow the SA to enter a value. This enables the SA to setup a user prior to putting any data in the database.

- ESM will then populate ZUSER with some default data. This table holds information about how the user wants the applications to handle certain situations. A record is inserted into ZUSER when an ID is added. The only value set is the Auto Calculations and Change Logging switch in ZUSER. This will set with the default value of 1 (On).
- ESM terminates successfully and returns the SA to create another user or to exit the discipline.

```
CREATE USER TEST00
  IDENTIFIED BY          ETEST00
  DEFAULT TABLESPACE   USERS
  TEMPORARY TABLESPACE TEMP
  PASSWORD              EXPIRE
  PROFILE               EAGLE_LOGIN_PROFILE;
GRANT CREATE SESSION TO TEST00;
GRANT SELECT ON EAGLE.ZID TO TEST00;
GRANT SELECT ON EAGLE.ZSYSTEM TO TEST00;
GRANT SELECT ON EAGLE.ZUSER TO TEST00;
```

Figure 5 Login Creation Script for EAGLE login

```
CREATE USER TEST01
  IDENTIFIED BY          TEST01
  DEFAULT TABLESPACE   USERS
  TEMPORARY TABLESPACE TEMP
  PROFILE               EAGLE_USER_PROFILE;
GRANT EAGLE_USER_ROLE TO TEST01;
ALTER USER TEST01 DEFAULT ROLE EAGLE_USER_ROLE;
```

Figure 6 Login Creation Script for EAGLE user

## 1.5.2 Random Password Generation and Encryption

### 1.5.2.1 The password encryption on Oracle works as follows.

Table ZID holds the Oracle ID (ID that the user has that just logs them in), EAGLE ID and EAGLE Password. Since the Oracle ID must have select access on ZID to figure out how to login, the password needs to be encrypted.

The initial password is created as a random number between 10,000 and 32,000. The letter ‘A’ is added to the front of the randomly generated password. The random number is generated using the PowerBuilder Rand function. The Rand function uses a seed value, this seed value will be set using the PowerBuilder Randomize function. This technique ensures that each request after a re-login by the security administrator creates a new random number.

### 1.5.2.2 Randomize operates as follows:

A starting value (seed) for the random number generator is required. When n is 0, PowerBuilder takes the seed from the system clock and begins a non-repeatable sequence. The sequence of numbers generated by repeated calls to the Rand function is a computer-generated pseudo-random sequence. ESM is using a seed value of 0 so that the Rand function is initialized with a value from the system clock.

When a new user is created the password is encrypted using the PowerBuilder encrypt function. The Oracle Database does not allow you to store non-printable characters in a character type column. To get around this the encrypted password is changed to a character string of ASCII numbers that represents the encrypted password. This is accomplished using the PowerBuilder Function `f_oracle_encrypt`. This value is then stored into table ZID.

At login, the EAGLE password is unencrypted by calling the PowerBuilder Function `f_oracle_unencrypt`. This function converts the ASCII representation of the encrypted password into the character representation of the encrypted password. This password is then unencrypted using the PowerBuilder function `decrypt`.

## 1.5.3 ESM Data Access and Control

When an attempt is made to access a data table, data stored in the EIACODXA and USERIDZU columns are matched to the ZUID table (see Table 3). If EIACODXA from the table being accessed matches ZEIACUID from table ZUID and USERIDZU from the table being accessed matches ZRFIDUID (Team Ownership Identification Code) from table ZUID for a user’s EAGLE identification code, updates to the data are allowed. If EIACODXA from the table being accessed matches ZEIACUID from table ZUID and USERIDZU from the table being accessed matches ZSRFIDUID (Select Team Ownership Identification Code) from table ZUID for a user’s EAGLE identification code, read only access is allowed. This is accomplished by joining the table being accessed with the ZUID table through the standard Oracle object VIEW. See Figure 7 for an example of a view. For more information on data access and control, see paragraph 1.7.

---

```
CREATE OR REPLACE VIEW EAGLE.VIEW_AA
AS SELECT
    EAGLE.AA.EIACODXA,
    EAGLE.AA.LSACONXB,
    EAGLE.AA.ALTLCNXB,
    EAGLE.AA.LCNTYPXB,
    EAGLE.AA.SERDESAA,
    EAGLE.AA.MAXTTRAA,
    EAGLE.AA.PERCENAA,
    EAGLE.AA.ACHAVAAA,
    EAGLE.AA.INHAVAAA,
    EAGLE.AA.OMAMDTAA,
    EAGLE.AA.TMAMDTAA,
    EAGLE.AA.OPMTTRAA,
    EAGLE.AA.TEMTTRAA,
    EAGLE.AA.NUOPLOAA,
    EAGLE.AA.CREWSZAA,
    EAGLE.AA.TOSYSUAA,
    EAGLE.AA.RCMLOGAA,
    EAGLE.AA.USERIDZU
FROM EAGLE.AA, EAGLE.ZUID
WHERE EAGLE.AA.EIACODXA = EAGLE.ZUID.ZEIACUID
AND ISNULL(EAGLE.AA.USERIDZU, ' ') LIKE EAGLE.ZUID.ZSRFIDUID
AND USER = EAGLE.ZUID.ZEIDUID
WITH CHECK OPTION;
```

Figure 7 Sample SQL for VIEW Creation

### 1.5.3.1 Accessing Data

To enable the users to access the EAGLE database using the table names documented in the MIL-STD 1388-2B, a SYNONYM has been created for each of the views. The following SQL syntax for the VIEW shown in Figure 7 is typical:

```
CREATE PUBLIC SYNONYM AA
FOR EAGLE.VIEW_AA
```

When a query is executed against 'AA' such as 'SELECT \* FROM AA', the query is executed against the View EAGLE.VIEW\_AA. If the match described in paragraph 1.5.3 does not exist, no records are returned.

### 1.5.3.2 Updating, Inserting, and Deleting Data

INSTEAD OF triggers are used to update, insert, or delete data. The INSTEAD OF trigger verifies that the requirements for database update described in paragraph 1.5.3 are met. If the requirements are not met, the following error message is raised:

```
-20915: '9999. SECURITY VIOLATION'
```

Examples of the INSTEAD OF triggers are shown in Figure 8 through Figure 10.

```
CREATE OR REPLACE TRIGGER EAGLE.AA_DEL_IO
INSTEAD OF DELETE
ON EAGLE.VIEW_AA
FOR EACH ROW
DECLARE
V_EIAC VARCHAR2(10);
V_RFID VARCHAR2(30);
V_RFID1 VARCHAR2(30);
E_SEC_1 EXCEPTION;
BEGIN
    BEGIN
        SELECT EAGLE.ZUID.ZEIACUID, EAGLE.ZUID.ZRFIDUID
            INTO V_EIAC, V_RFID
        FROM EAGLE.ZUID
        WHERE
            USER = EAGLE.ZUID.ZEIDUID AND
            :OLD.EIACODXA = EAGLE.ZUID.ZEIACUID;
    EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RAISE E_SEC_1;
    END;
    IF :OLD.USERIDZU IS NULL THEN
        V_RFID1 := V_RFID;
    ELSE
        V_RFID1 := :OLD.USERIDZU;
    END IF;
    IF :OLD.EIACODXA = V_EIAC AND
        ISNULL(V_RFID1, '') = ISNULL(V_RFID, '') THEN
        DELETE EAGLE.AA WHERE
            EAGLE.AA.EIACODXA = :OLD.EIACODXA AND
            EAGLE.AA.LSACONXB = :OLD.LSACONXB AND
            EAGLE.AA.ALTLCNXB = :OLD.ALTLCNXB AND
            EAGLE.AA.LCNTYPXB = :OLD.LCNTYPXB AND
            EAGLE.AA.SERDESAA = :OLD.SERDESAA;
    ELSE
        RAISE E_SEC_1;
    END IF;
EXCEPTION
    WHEN E_SEC_1 THEN
        RAISE_APPLICATION_ERROR(-20915, '9999. SECURITY VIOLATION');
END;
```

Figure 8 Example INSTEAD OF DELETE Trigger

```
CREATE OR REPLACE TRIGGER EAGLE.AA_INS_IO
INSTEAD OF INSERT
ON EAGLE.VIEW_AA
FOR EACH ROW
DECLARE
V_EIAC VARCHAR2(10);
V_RFID VARCHAR2(30);
V_RFID1 VARCHAR2(30);
E_SEC_1 EXCEPTION;
BEGIN
  BEGIN
  SELECT EAGLE.ZUID.ZEIACUID, EAGLE.ZUID.ZRFIDUID
  INTO V_EIAC, V_RFID
  FROM EAGLE.ZUID
  WHERE
  USER = EAGLE.ZUID.ZEIDUID AND
  :NEW.EIACODXA = EAGLE.ZUID.ZEIACUID;
EXCEPTION
WHEN NO_DATA_FOUND THEN
  RAISE E_SEC_1;
END;
IF :NEW.USERIDZU IS NULL THEN
  V_RFID1 := V_RFID;
ELSE
  V_RFID1 := :NEW.USERIDZU;
END IF;
IF :NEW.EIACODXA = V_EIAC AND
ISNULL(V_RFID1, '') = ISNULL(V_RFID, '') THEN
  INSERT INTO EAGLE.AA VALUES (
  :NEW.EIACODXA,
  :NEW.LSACONXB,
  :NEW.ALTLCNXB,
  :NEW.LCNTYPXB,
  :NEW.SERDESAA,
```

Figure 9 Example INSTEAD OF INSERT Trigger (Sheet 1 of 2)

```
:NEW.MAXTTRAA,  
:NEW.PERCENAA,  
:NEW.AHAVAAA,  
:NEW.INHAVAAA,  
:NEW.OMAMDTAA,  
:NEW.TMAMDTAA,  
:NEW.OPMTTRAA,  
:NEW.TEMTTRAA,  
:NEW.NUOPLOAA,  
:NEW.CREWSZAA,  
:NEW.TOSYSUAA,  
:NEW.RCMLOGAA,  
V_RFID1);  
  ELSE  
    RAISE E_SEC_1;  
  END IF;  
EXCEPTION  
  WHEN E_SEC_1 THEN  
    RAISE_APPLICATION_ERROR(-20915, '9999. SECURITY VIOLATION');  
END;
```

Figure 9 Example INSTEAD OF INSERT Trigger (Sheet 2 of 2)

```
CREATE OR REPLACE TRIGGER EAGLE.AA_UPD_IO
INSTEAD OF UPDATE
ON EAGLE.VIEW_AA
FOR EACH ROW
DECLARE
V_EIAC VARCHAR2(10);
V_RFID VARCHAR2(30);
V_RFID1 VARCHAR2(30);
E_SEC_1 EXCEPTION;
BEGIN
  BEGIN
  SELECT EAGLE.ZUID.ZEIACUID, EAGLE.ZUID.ZRFIDUID
  INTO V_EIAC, V_RFID
  FROM EAGLE.ZUID
  WHERE
  USER = EAGLE.ZUID.ZEIDUID AND
  :NEW.EIACODXA = EAGLE.ZUID.ZEIACUID;
EXCEPTION
WHEN NO_DATA_FOUND THEN
  RAISE E_SEC_1;
END;
IF :OLD.USERIDZU IS NULL THEN
  V_RFID1 := V_RFID;
ELSE
  V_RFID1 := :NEW.USERIDZU;
END IF;
IF :NEW.EIACODXA = V_EIAC AND
ISNULL(V_RFID1, ' ') = ISNULL(V_RFID, ' ') THEN
  UPDATE EAGLE.AA SET
  EAGLE.AA.MAXTTRAA = :NEW.MAXTTRAA,
  EAGLE.AA.PERCENAA = :NEW.PERCENAA,
  EAGLE.AA.ACHAVAAA = :NEW.ACHAVAAA,
  EAGLE.AA.INHAVAAA = :NEW.INHAVAAA,
  EAGLE.AA.OMAMDTAA = :NEW.OMAMDTAA,
  EAGLE.AA.TMAMDTAA = :NEW.TMAMDTAA,
  EAGLE.AA.OPMTTRAA = :NEW.OPMTTRAA,
```

Figure 10 Example INSTEAD OF UPDATE Trigger (Sheet 1 of 2)

```
EAGLE.AA.TEMTTRAA = :NEW.TEMTTRAA,  
EAGLE.AA.NUOPLOAA = :NEW.NUOPLOAA,  
EAGLE.AA.CREWSZAA = :NEW.CREWSZAA,  
EAGLE.AA.TOSYSUAA = :NEW.TOSYSUAA,  
EAGLE.AA.RCMLOGAA = :NEW.RCMLOGAA,  
EAGLE.AA.USERIDZU = V_RFID1  
WHERE  
EAGLE.AA.EIACODXA = :NEW.EIACODXA AND  
EAGLE.AA.LSACONXB = :NEW.LSACONXB AND  
EAGLE.AA.ALTLCNXB = :NEW.ALTLCNXB AND  
EAGLE.AA.LCNTYPXB = :NEW.LCNTYPXB AND  
EAGLE.AA.SERDESAA = :NEW.SERDESAA;  
ELSE  
  RAISE E_SEC_1;  
END IF;  
EXCEPTION  
  WHEN E_SEC_1 THEN  
    RAISE_APPLICATION_ERROR(-20915, '9999. SECURITY VIOLATION');  
END;
```

Figure 10 Example INSTEAD OF UPDATE Trigger (Sheet 2 of 2)

## 1.6 AUDITING USER ACTIONS

EAGLE provides the ability to audit insert, update, and delete transactions on its database. The ability to turn this functionality on or off is limited to the EAGLE userid. This is accomplished by setting the column CHGLOGZS in Table ZSYSTEM to a 1.

Once this column is turned on (set to a value of 1) all INSERTS, UPDATES and DELETES performed on the LSAR database, except for temporary tables, will be logged in a table called ZCHANGE. Pre-change and post-change images of the row affected are stored in the ZCHANGE row. If the transaction is an insert command the pre-change portion of the row will be blank and post-change will contain the inserted row image of the transaction. If the transaction is a delete command the pre-change portion of the row will contain the image of the transaction and the post-change will be blank. If the transaction is an update command the pre-change portion of the row will contain the image of the row prior to the update and the post-change portion of the row will contain the image of the row after the update.

If change logging is enabled, table ZCHANGE will grow rapidly in size. The EAGLE database administrator must develop a plan to archive this data on a regularly scheduled basis. EAGLE provides two methods of doing this. One is through the use of the FULLFILE export utility by selecting only the ZCHANGE table and exporting it, storing the exported file outside of the EAGLE database server, and truncating the table. The other method would be to use the AdHoc discipline to select a specific range of data to extract and store offline. The ZCHANGE records can then be deleted for the selected range. The EAGLE DBA may wish to explore other methods including the use of third party tools. When change logging is enabled, for each insert, update, and or delete a second row is built in the database. While this is not a great impact, it is never the less a second I/O operation.

The Database Audit Trail Discipline allows each user to peruse the audit table. See the EAGLE Workbook for details on this discipline.

The audit functionality provided by EAGLE is not a replacement for standard Oracle auditing but rather an extension to it. The standard Oracle functionality of auditing does not support auditing at the row level. It does, however, create other useful statistics. For a complete audit trail, the EAGLE database administrator should enable standard Oracle auditing and EAGLE auditing.

## 1.7 SECURITY EXAMPLES

Users may access all End Items for which they are approved. A user will be able to view all data related to End Items they are approved for by selecting 'ALL' for End Item in the Select End Item window. For example, assume the database contains 10 End Items and a user is approved for 5 of these End Items. If 'ALL' is selected in the End Item Select window, EAGLE will allow access to data only for the 5 End Items the user is approved for.. The selection of data is further limited by select teaming ID as described below.

Users are assigned a teaming ID, for example TEAM00 or TEAM01. This ID is used to determine which records the user can modify. Users are only allowed to modify records that match their teaming ID. Users are also assigned a select teaming ID, for example % or TEAM01. Users assigned select teaming IDs of % are allowed to view all data. Users assigned select teaming IDs of TEAM01 are only allowed to view records that match their select teaming ID.

The following examples assume that the users have access to the End Item that is being queried/modified. Also, the user is not logged in as SU or EAGLE.

### Example 1

Teaming ID	= TEAM01
Select Teaming ID	= TEAM01
XB Record Owner	= TEAM00
JA Record Owner	= TEAM01
JB Record Owner	= TEAM01

The user will not be able to access the JA record using the Transportation finder because the finder queries XB and the user does not own the XB record. The user would have to use AdHoc to modify the record. The only way this data could have been entered into the database would have been with AdHoc or Fullfile. The user can copy JA and JB.

### Example 2

Teaming ID	= TEAM01
Select Teaming ID	= %
XB Record Owner	= TEAM00
JA Record Owner	= TEAM01
JB Record Owner	= TEAM01

The user will be able to select data from the XB record using the Transportation finder because they have a select teaming ID that allows them to view all data. The user can access and modify both the JA and JB records. The user can copy JA and JB.

### Example 3

Teaming ID	= TEAM01
Select Teaming ID	= TEAM01
XB Record Owner	= TEAM01
JA Record Owner	= TEAM00

The user will be able to query the XB record on the Transportation finder but would not see the associated JA data in the JA screen because they do not own the JA record.

**Example 4**

Teaming ID	= TEAM01
Select Teaming ID	= %
XB Record Owner	= TEAM01
JA Record Owner	= TEAM00

The user will be able to query the XB record on the Transportation finder and see the associated JA data in the JA screen. The user would not be able to modify the JA record because they are not the owner.

**Example 5**

Teaming ID	= TEAM01
Select Teaming ID	= TEAM01
XB Record Owner	= TEAM01
JA Record Owner	= TEAM01
JB Record Owner	= TEAM00

The user will be able to query the XB record on the Transportation finder and update the non-key fields in the JA record. The user would not be able to delete the JA record because a child record (JB) is not owned by them. The user can not perform a key field change because they do not have privileges on table JB. The user would not be able to see data in the JB table.

**Example 6**

Teaming ID	= TEAM01
Select Teaming ID	= %
XB Record Owner	= TEAM01
JA Record Owner	= TEAM01
JB Record Owner	= TEAM00

The user will be able to query the XB record on the Transportation finder and update the non-key fields in the JA record. The user would not be able to delete the JA record because a child record (JB) is not owned by them. The user can not perform a key field change because they do not have privilege on table JB. The user would be able to see data in the JB table, but not modify it.

**Example 7**

Teaming ID = TEAM00  
Select Teaming ID = %  
XB Record Owner = TEAM01

The user will be able to query the XB record on the Transportation finder because select teaming ID of % has select access on all records. The user will be able to insert a JA record in this scenario. The user can not perform a XB copy or XB key field change in AdHoc.

**Example 8**

Teaming ID = TEAM00  
Select Teaming ID = TEAM00  
XB Record Owner = TEAM01

The user will not be able to query the XB record on the Transportation finder because select teaming ID of TEAM00 does not have access to records owned by TEAM01.

**Example 9**

Teaming ID = TEAM00  
Select Teaming ID = %  
XB Record Owner = TEAM01  
JA Record Owner = TEAM01

The user will be able to query the XB record on the Transportation finder because select teaming ID of % has select access on all records. The user will be able to see the JA record but not update it because they do not own it. The user can not perform a JA copy or JA key field change in AdHoc.

**Example 10**

Teaming ID = TEAM00  
Select Teaming ID = %  
XB Record Owner = TEAM01  
JA Record Owner = TEAM00  
JB Record Owner = TEAM01

The user will be able to query the XB record on the Transportation finder because select teaming ID of % has select access on all records. The user will be able to update non-key fields in the JA record. The user would not be able to delete the JA record because a child record (JB) is not owned by them. The user can not perform a JA key field change because they do not have privilege on table JB.

**Example 11**

Teaming ID	= TEAM00
Select Teaming ID	= TEAM00
XB Record Owner	= NULL
JA Record Owner	= NULL
JB Record Owner	= TEAM01

The user will be able to query the XB record on the Transportation finder because the record does not have an owner. This could only happen if security was turned off in table ZSYSTEM and then turned on after data was entered or if the user was logged in as SU or EAGLE when the XB was created. The user will be able to change the JA record since no one owns it. After the JA record is changed it would be assigned a record owner of 'TEAM00', because they were the first one to touch it with security.

Note: Using the EAGLE disciplines as EAGLE or SU will not effect the record owner, unless an owner is passed to the database. The only way to pass an owner to the database in EAGLE is to use AdHoc or Fullfile. For example, an XA record added using the LCN Maintenance discipline, logged in as EAGLE will set the row owner to NULL.

# ***SECTION 2***



# ***EAGLE SECURITY MAINTENANCE***

---



## SECTION 2 EAGLE SECURITY MAINTENANCE

## 2.0 INTRODUCTION

This section provides instructions for maintaining EAGLE security through the EAGLE Security Maintenance Discipline. Only the Database Administrator (EAGLE) or Security Administrator accounts can perform EAGLE security maintenance. See Section 1 of this manual for a detailed discussion of the EAGLE Security System design.

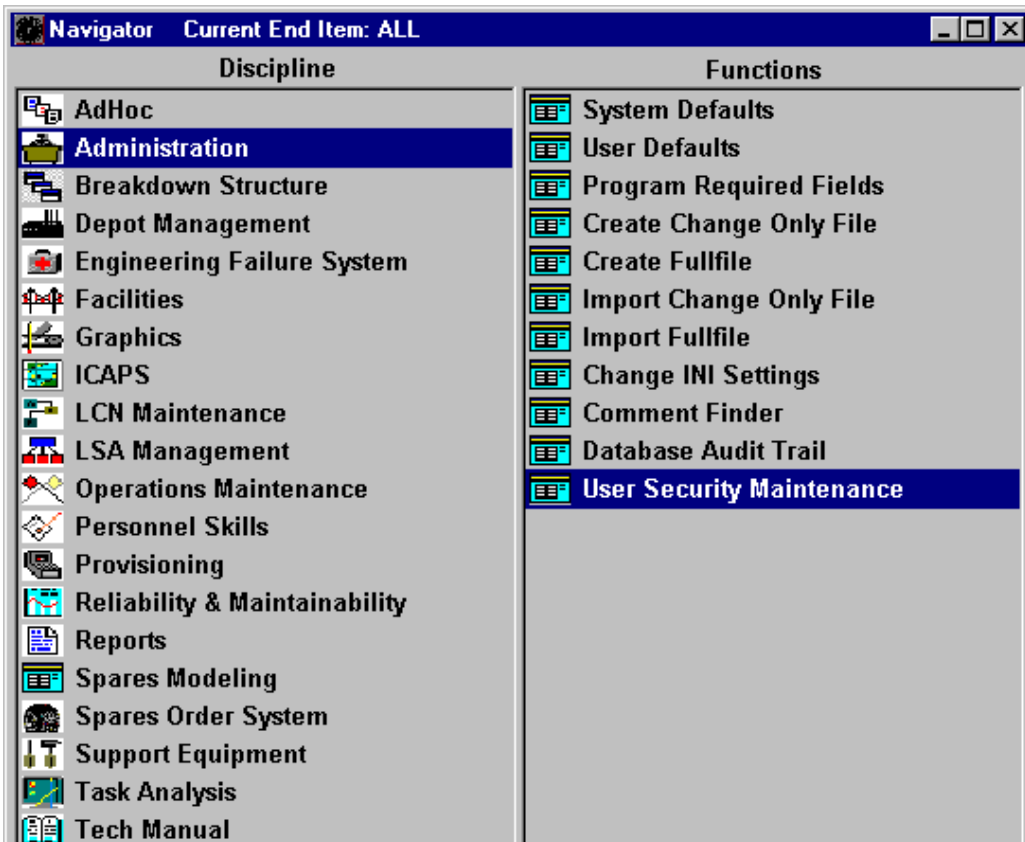


Figure 11 Accessing User Security Maintenance From the Navigator

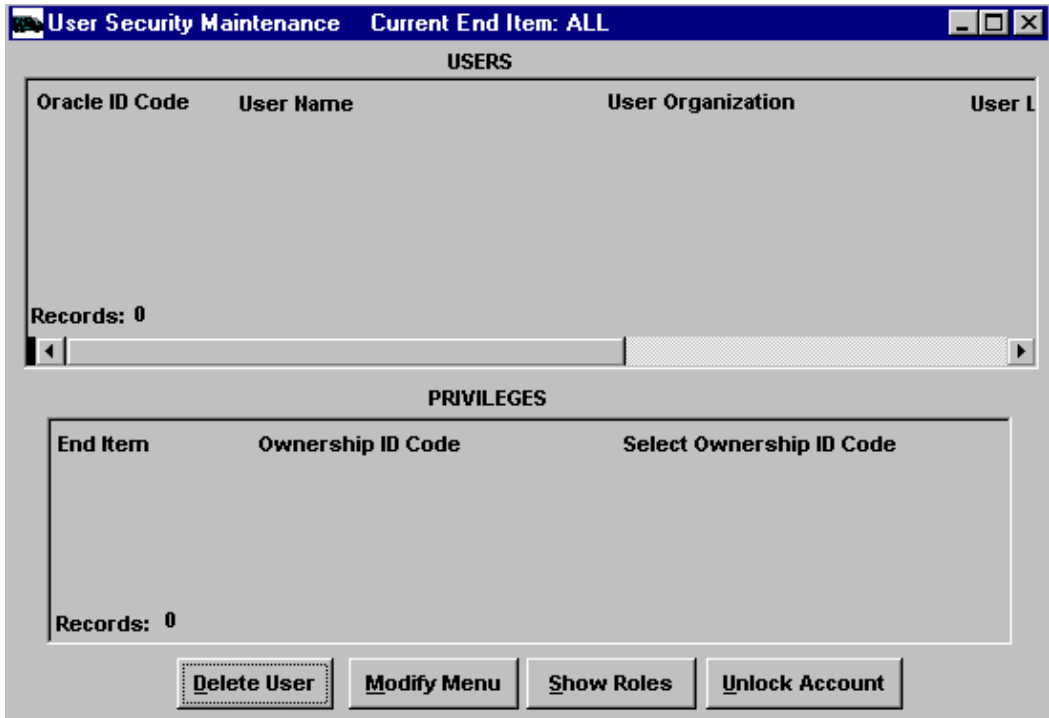


Figure 12 User Security Maintenance Window

## 2.1 ADDING USERS

In order to establish a new user account, data must be established in the Primary Security Table (TABLE ZID). This is accomplished by executing the User Security Maintenance function from the Administration discipline of the EAGLE Navigator window (see Figure 11). From the User Security Maintenance window shown in Figure 12, perform the following steps:

- Choose **Reports/Process>>New User** from the Menu Bar or the **New** button from the Function Specific Toolbar. The Create New User window shown in Figure 13 is displayed.
- Enter the Oracle ID, Oracle Password, and select the desired User Role from the User Role drop down list box (mandatory entries)
- Enter the User Name, User Organization, User Location, and User Phone (optional entries)

Note: The user will be required to change the Oracle Password during the initial login.

**Create New User** Current End Item: ALL

**User Information**

Oracle ID:  Oracle Password:

User Role: **EAGLE USER ROLE**

User Name:  User Organization:

User Location:  User Phone:

**PRIVILEGES**

End Item	Ownership ID Code	Select Ownership ID Code

Records: 0

**Add User**

Figure 13 Create New User Window

Next, privileges must be established in the EAGLE Userid Table, Table ZID, as follows:

- Choose the End Item from the End Item drop down list box or enter the desired End Item if it does not currently exist
- Choose the Ownership ID Code from the Ownership ID Code drop down list box or enter the desired Ownership ID Code if it does not currently exist
- Choose the Select Ownership ID Code from the Select Ownership ID Code drop down list box or enter the desired Select Ownership ID Code if it does not currently exist
- If the user is to have access to additional End Items, choose the **Insert** button on the Main Toolbar and repeat the previous three (3) steps
- Choose the **Add User** button on the Create New User window.
- Answer **Yes** or **No** when prompted "Do you want the ability to customize the user's Navigator?"

Note: A "User Successfully Created" message indicates that the user account has been successfully created.

- Choose the **OK** button to acknowledge.
- Close the Create New User window to return to the User Security Maintenance window.
- Choose the **Execute** button on the Main Toolbar to display all user accounts

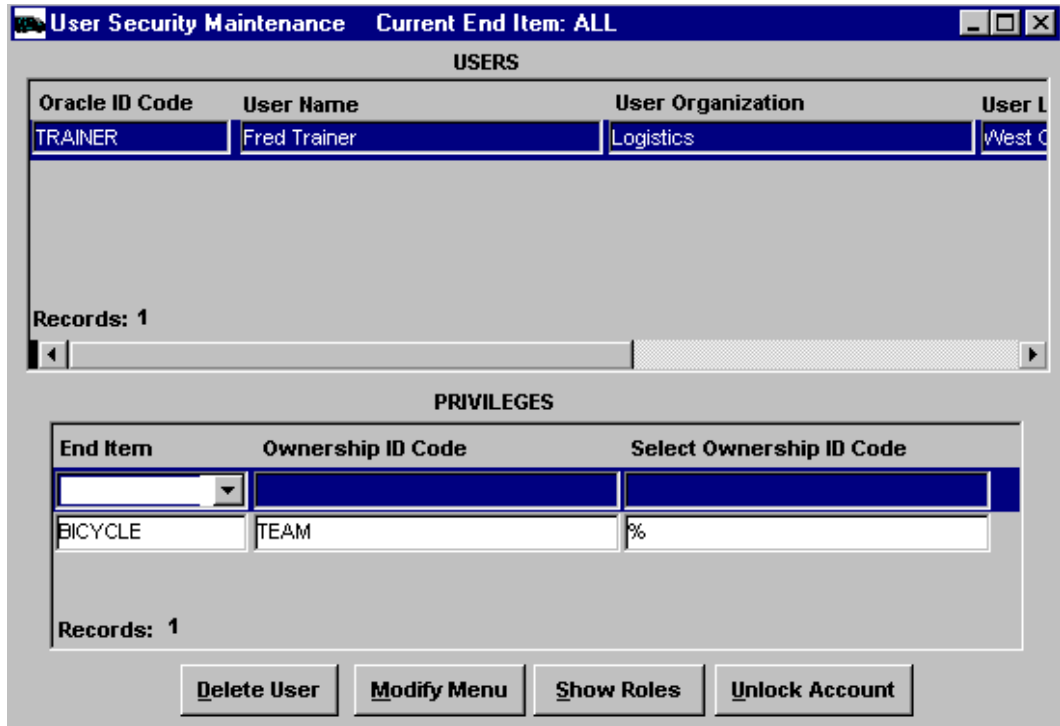


Figure 14 Granting Privileges to Existing User Accounts

## 2.2 GRANTING PRIVILEGES TO EXISTING USER ACCOUNTS

Privileges may be granted to existing user accounts in a manner similar to granting privileges when establishing new user accounts. From the User Security Maintenance Window shown in Figure 14, choose the User to whom privileges are to be granted and perform the following steps.

Note: Figure 14 shows the User Security Maintenance window *after* a new privileges record has been inserted.

- Activate the lower portion of the User Security Maintenance window by clicking on it (within the box below the "PRIVELEGES" heading)
- Choose the **Insert** button on the Main Toolbar to insert a new Privileges record
- Choose the End Item from the End Item drop down list box or enter the desired End Item if it does not currently exist
- Choose the Ownership ID Code from the Ownership ID Code drop down list box or enter the desired Ownership ID Code if it does not currently exist

- Choose the Select Ownership ID Code from the Select Ownership ID Code drop down list box or enter the desired Select Ownership ID Code if it does not currently exist
- If the user is to have access to additional End Items, choose the **Insert** button on the Main Toolbar and repeat the previous three (3) steps
- Choose the **Save** button on the Main Toolbar

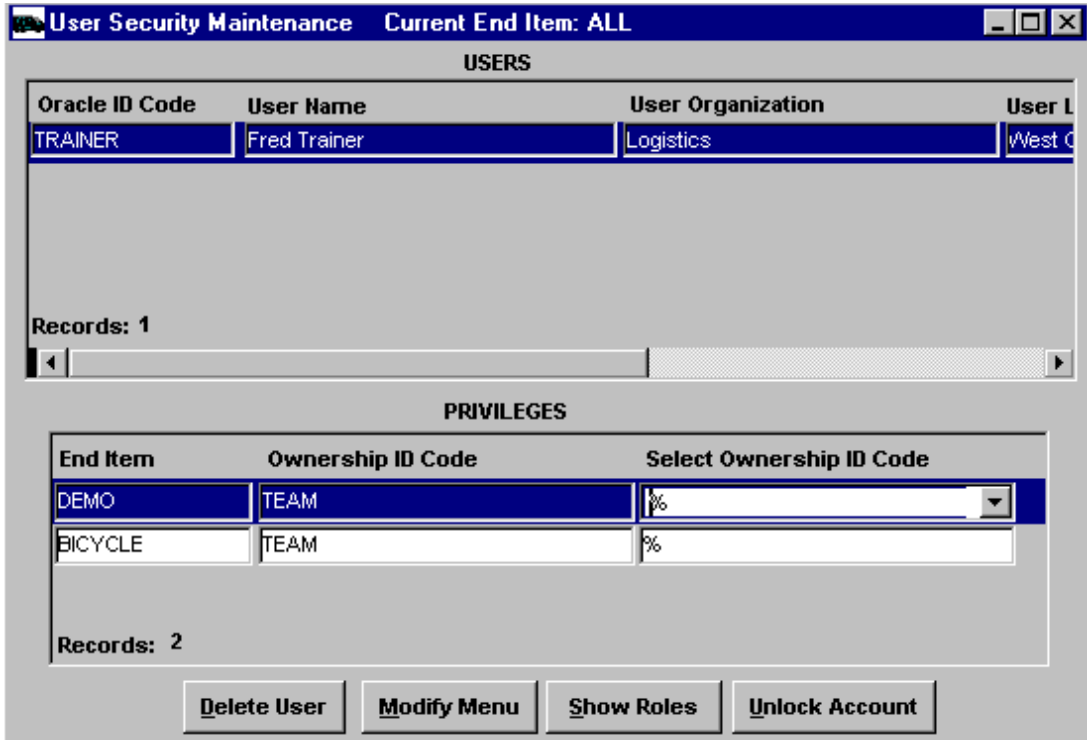


Figure 15 Revoking Privileges From Users

### 2.3 REVOKING PRIVILEGES FROM USERS

Selected privileges may be revoked from individual user accounts. From the User Security Maintenance window shown in Figure 15, perform the following steps:

- Choose the User for whom privileges are to be revoked
- Choose the Privilege to be revoked

- Choose the **Delete** button on the Main Toolbar
- Choose the **Save** button on the Main Toolbar to complete the deletion

Note: Privileges are not deleted until the **Save** button is chosen. To discontinue the deletion of privileges without saving, choose the **Execute** button on the Main Toolbar and respond “**No**” when prompted to save changes.

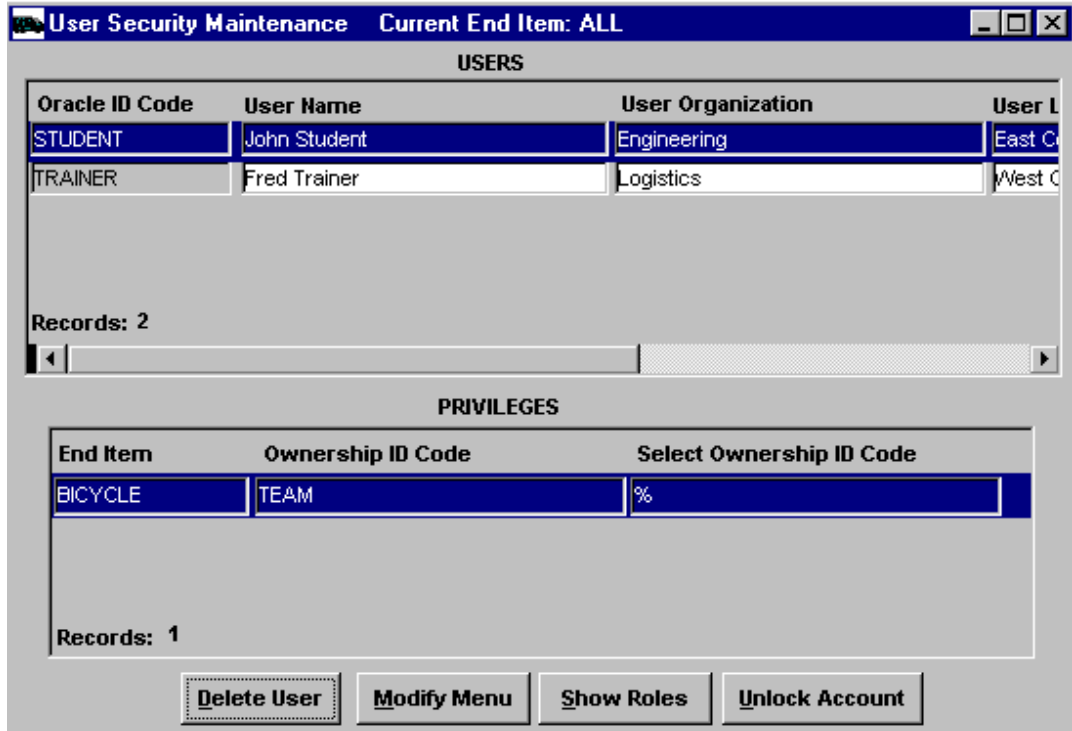


Figure 16 Deleting User Accounts

## 2.4 DELETING USER ACCOUNTS

User accounts may be deleted using the User Security Maintenance Window shown in Figure 16 as follows:

- Select the user account to be deleted from the User Security Maintenance Window
- Choose the **Delete** button located on the User Security Maintenance Window
- Choose the **Yes** button when prompted to save changes to delete the user account, or, choose the **No** button to discontinue the deletion. If you choose to discontinue the deletion, choose the **Execute** button on the Main Toolbar. When prompted to save changes, choose the **No** button once again. This will prevent the deletion from inadvertently occurring when other transactions are performed.

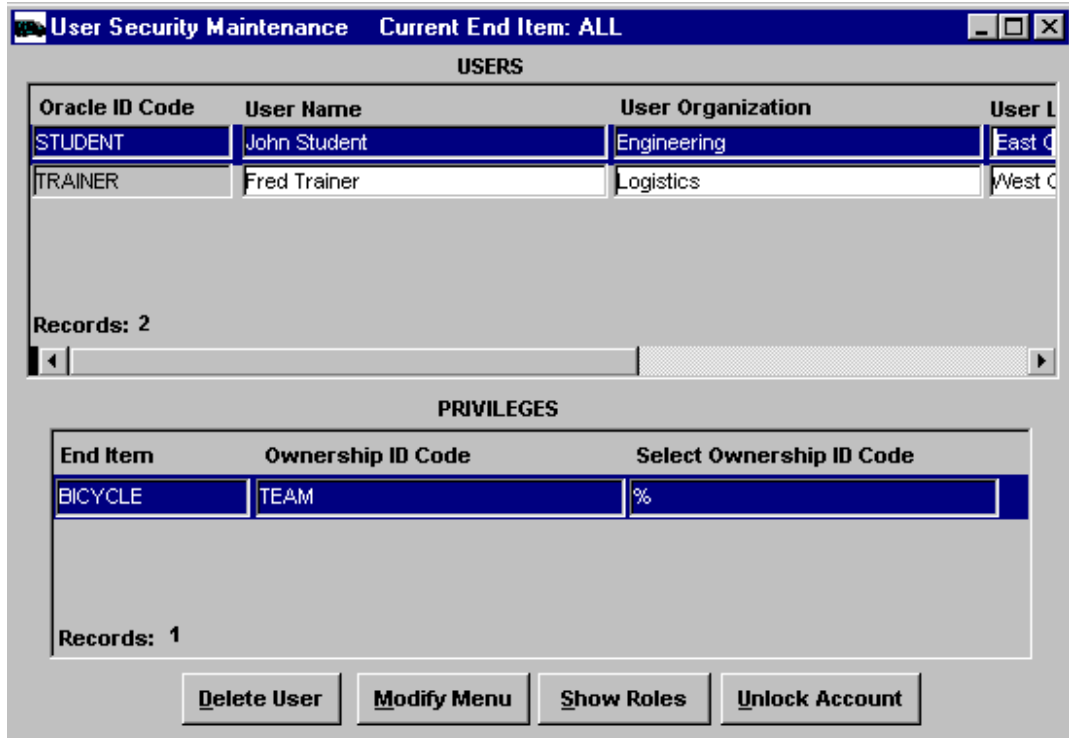


Figure 17 User Security Maintenance Window - Cloning User Accounts

## 2.5 CLONING USER ACCOUNTS

User account information for a new user may be cloned from the account information of an existing user without having to establish privileges for the new account. This can be especially useful when users have privileges on multiple End Items. To clone a user account, perform the following steps from the User Security Maintenance Window as shown in Figure 17:

- Choose the User Account to be cloned
- Choose **Reports/Process>>Clone User** from the Menu Bar or the **Clone** button from the Function Specific Toolbar. The Clone User window shown in Figure 18 is displayed.

**Clone User** Current End Item: ALL

**From User Information**

Oracle ID: STUDENT User Name: John Student

Oracle ID: STUDENT2 Oracle Password: \*\*\*\*\*

User Role: EAGLE\_USER\_ROLE

User Name: Alice Student User Organization: Engineering

User Location: East Coast User Phone: (999) 999-1234

Clone User

Figure 18 Clone User Window

- Enter the Oracle ID, Oracle Password, and select the desired User Role from the User Role drop down list box (mandatory entries) as shown in Figure 18
- Enter the User Name, User Organization, User Location, and User Phone (optional entries)
- Choose the **Clone User** button on the Clone User window
- Answer **Yes** or **No** when prompted "Do you want the ability to customize the user's Navigator?"

Note: A "User Successfully Created" message indicates that the user account has been successfully cloned.

- Choose the **OK** button to acknowledge that the user account has been successfully cloned
- Close the Clone User window to return to the User Security Maintenance Window
- Choose the **Execute** button on the Main Toolbar to retrieve the new user account information

Note: If additional privileges must be granted to the new user account or if existing privileges must be revoked, see paragraph 2.2 or paragraph 2.3 as required.

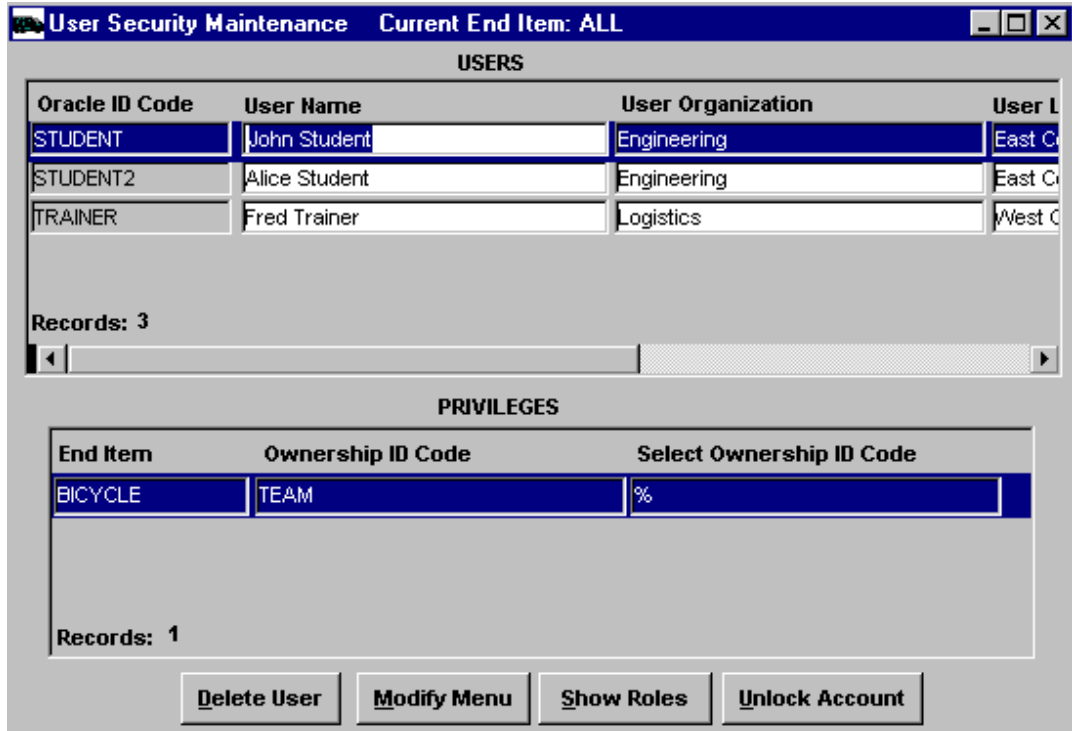


Figure 19 User Security Maintenance Window - Customizing the Navigator

## 2.6 CUSTOMIZING THE NAVIGATOR FOR INDIVIDUAL USER ACCOUNTS

Menu options displayed in the EAGLE Navigator are stored in table ZMENU. This data can be customized for each user account so that only menu options for which a user has a need to access will be displayed in the user's Navigator window. To customize a user's menu options, perform the following steps from the User Security Maintenance Window as shown in Figure 19:

- Choose the User Account for whom the menu options are to be customized
- Choose the **Modify Menu** button on the User Security Maintenance Window

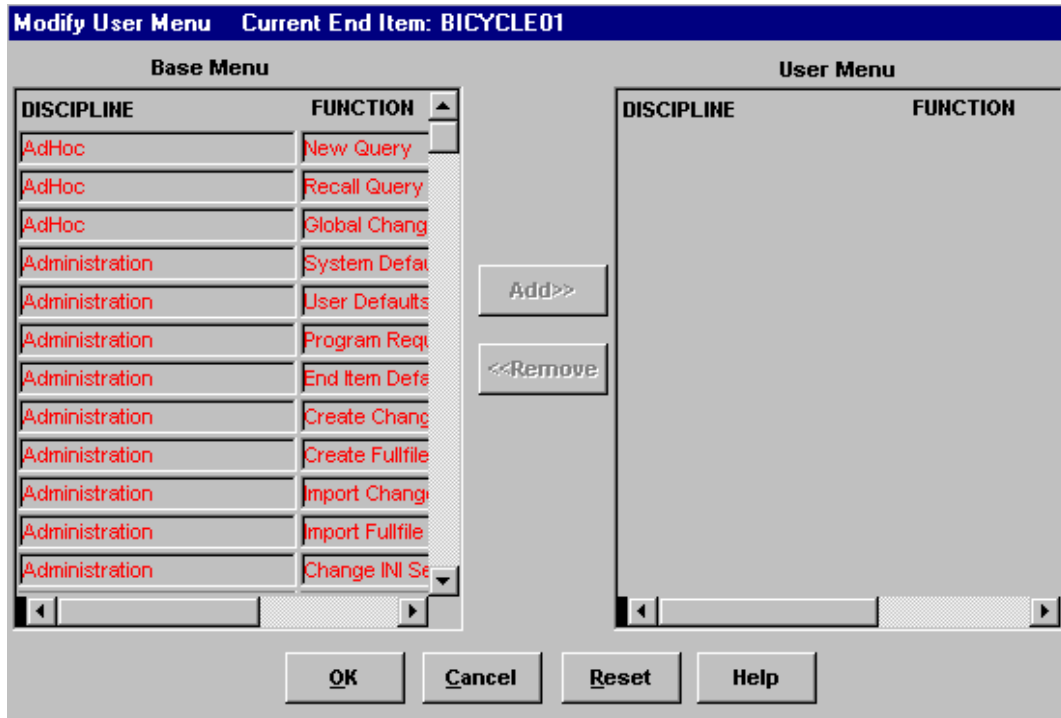


Figure 20 Modify User Menu Window

- If additional menu options are to be added, from the Modify User Menu window shown in Figure 20, select the menu option to be added from the Base Menu list and choose the **Add>>** button
- If menu options are to be withheld from the user's menu, select the menu item to be withheld from the User Menu list and choose the **<<Remove** button
- If all menu options in the Base Menu are to be added to the User's Menu, choose the **Reset** button
- Choose the **OK** button on the Modify User Menu window. The Modify User Menu window closes and the user's menus are now customized.

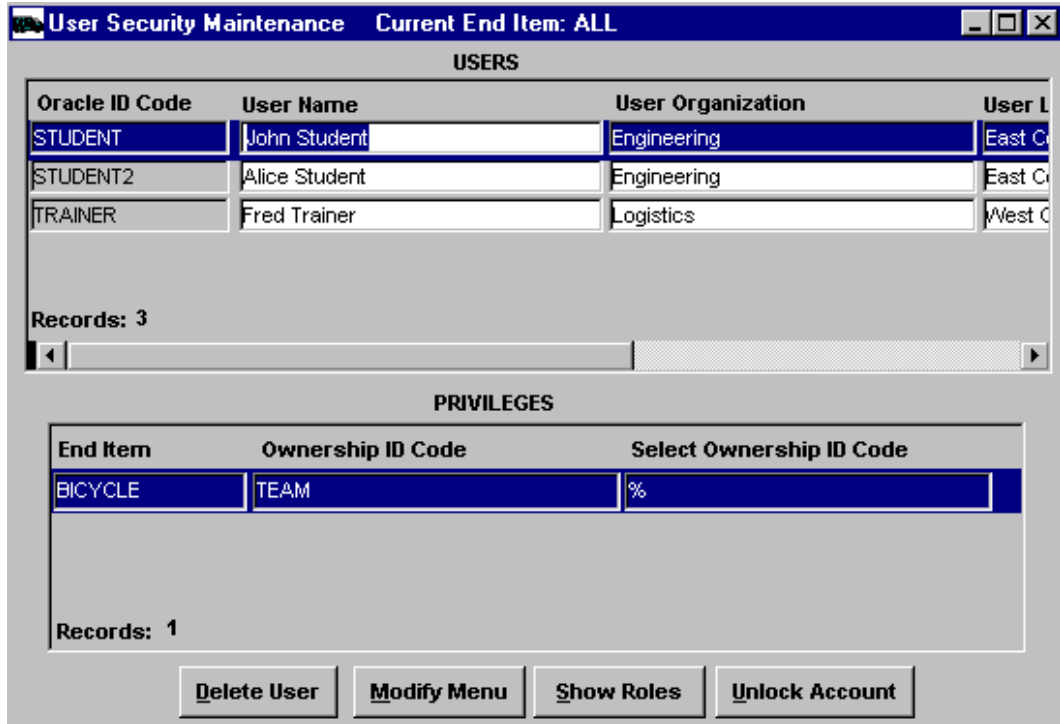


Figure 21 User Security Maintenance Window- Altering User Accounts

## 2.7 ALTERING USER ACCOUNTS

From the EAGLE User Security Maintenance Window, the security administrator can alter a user's account information by changing the user's Oracle Password, User Roll, User Name, Organization, Location, or Phone Number. This can be useful when users forget their password or when the user's responsibilities change. To alter a user's account information, from the User Security Maintenance window shown in Figure 21, perform the following steps:

- Choose the User Account to be altered
- Choose **Reports/Process>>Alter User** from the Menu Bar or the **Alter** button from the Function Specific Toolbar. The Alter User window shown in Figure 22 is displayed.

**Alter User** Current End Item: ALL

**From User Information**

Oracle ID:  User Name:

Oracle Password:

User Role:

User Name:  User Organization:

User Location:  User Phone:

Figure 22 Alter User Window

- Enter the required changes and choose the **Alter User** button on the Alter User window
- Choose the **OK** button to acknowledge the user was successfully altered
- Close the Alter User window
- Choose the **Execute** button on the Main Toolbar to retrieve the altered user information

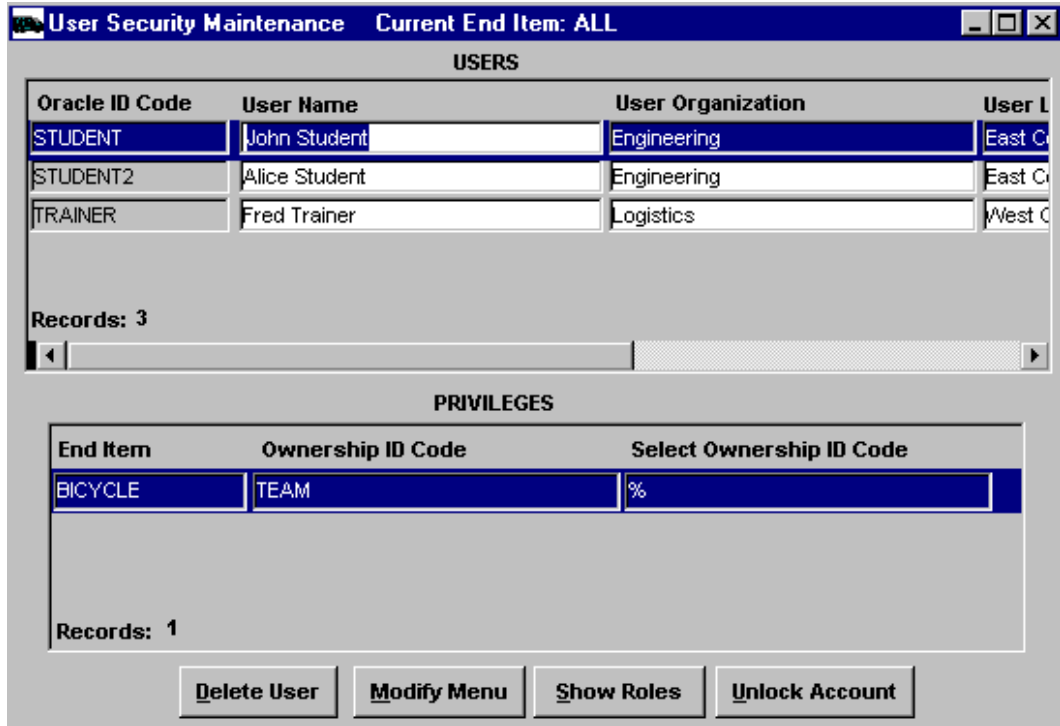


Figure 23 User Security Maintenance Window - Viewing User Roles

## 2.8 VIEWING USER ROLES

The role assigned to a user may be viewed by selecting the Show Roles button from User Security Maintenance window shown in Figure 23. This will display the User Roles window shown in Figure 24.

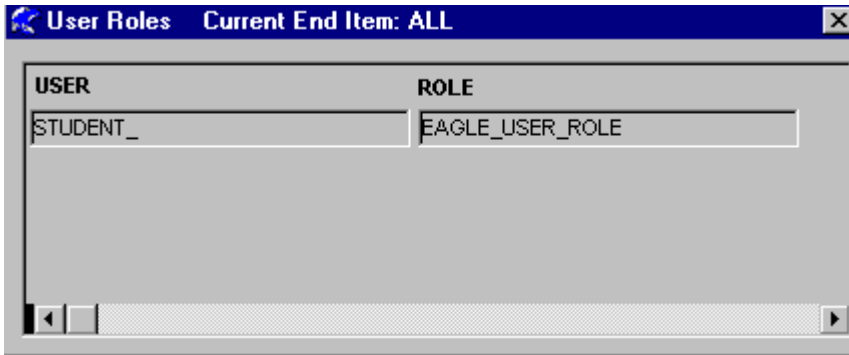


Figure 24 User Roles Window

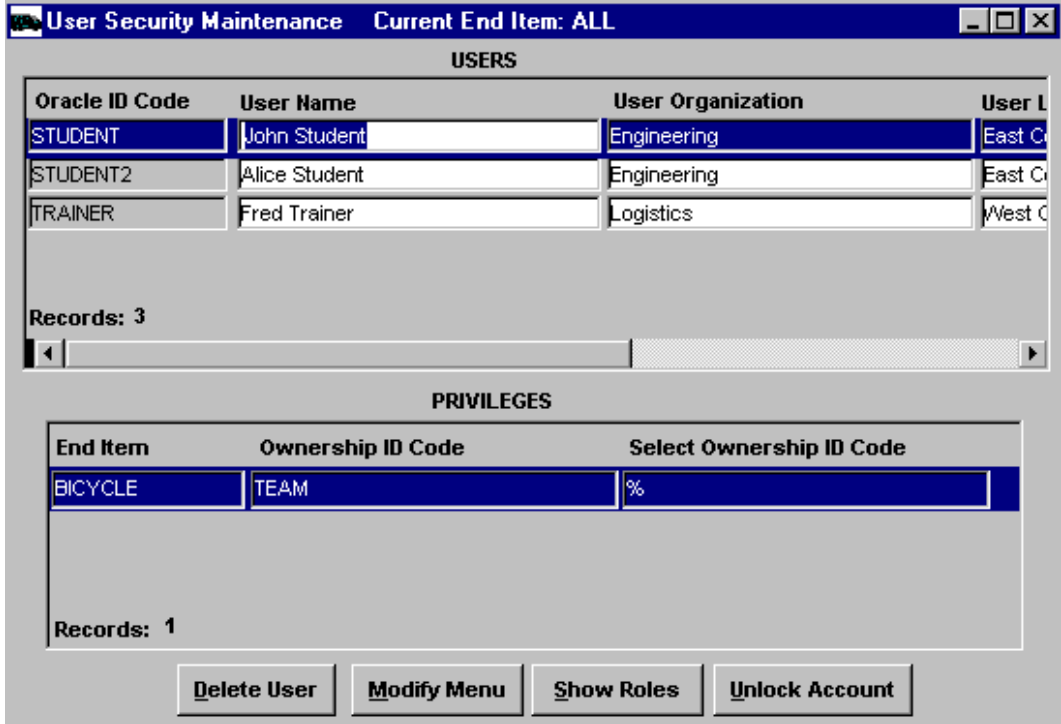


Figure 25 User Security Maintenance Window - Unlocking User Accounts

## 2.9 UNLOCKING USER ACCOUNTS

A user account will become locked if three consecutive login attempts fail. The security administrator can unlock user accounts by performing the following steps from the User Security Maintenance window shown in Figure 25:

- Choose the user account to unlock
- Choose the **Unlock Account** button on the User Security Maintenance window

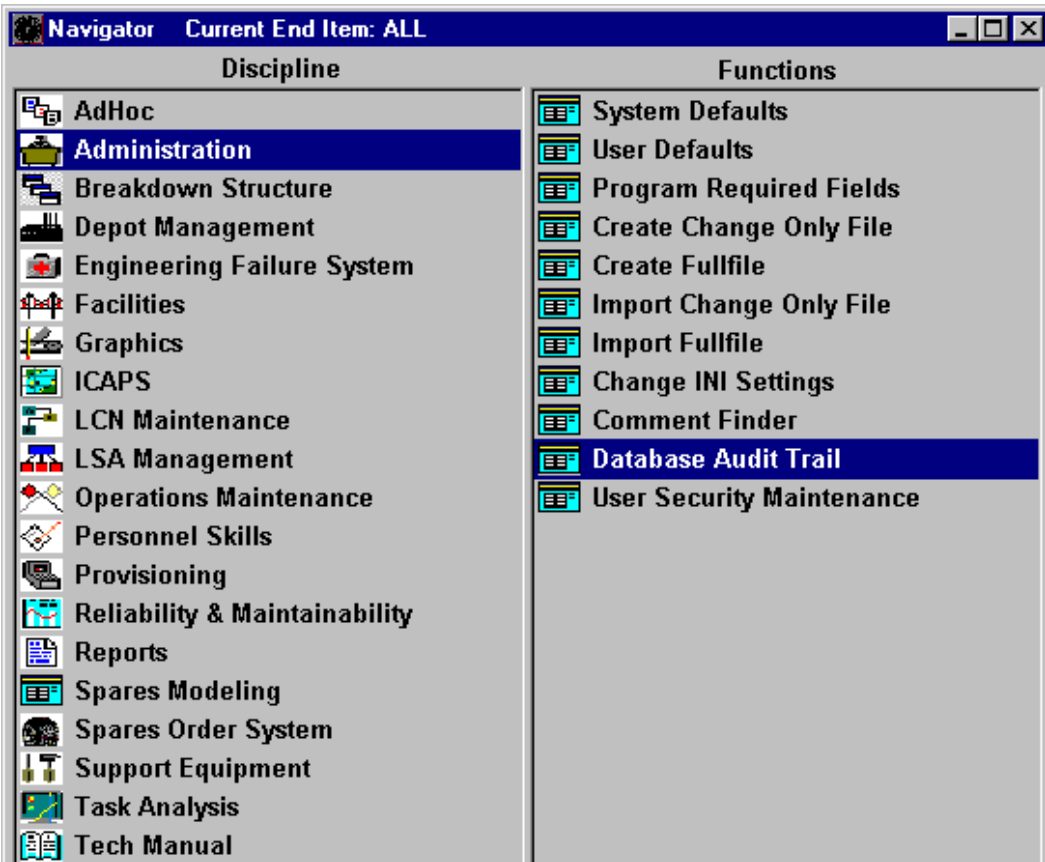


Figure 26 Accessing The Database Audit Trail From the Navigator

## 2.10 VIEWING DATABASE AUDIT TRAIL INFORMATION

EAGLE provides the ability to audit insert, update, and delete transactions on its database. This capability can be enabled/disabled as described in paragraphs 1.6 and 2.11. If the change logging feature is selected in the ZSYSTEM table, every change that is made to the database is recorded in a log table called ZCHANGE. This table may be viewed through the EAGLE application using the Database Audit Trail Function. To view the database audit trail, select the Database Audit Trail Function from the Administration Discipline using the Navigator as shown in Figure 27.

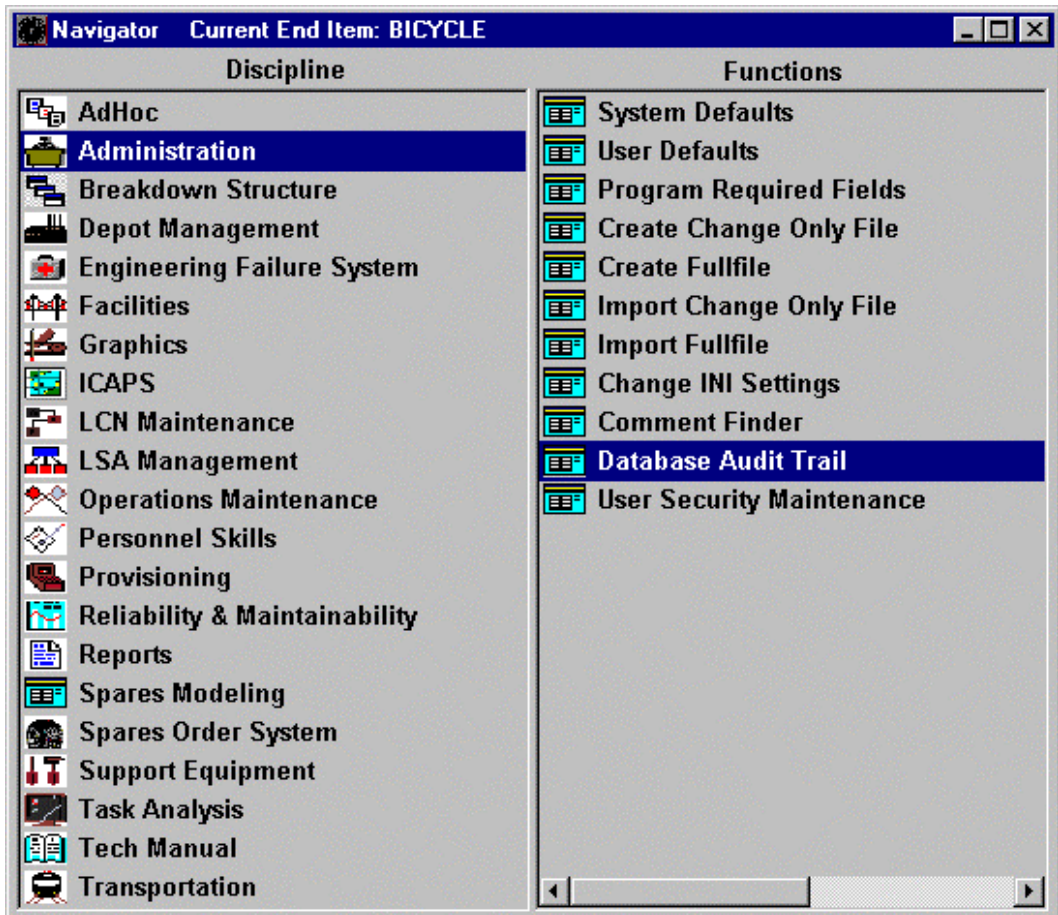


Figure 27 Navigator - Database Audit Trail

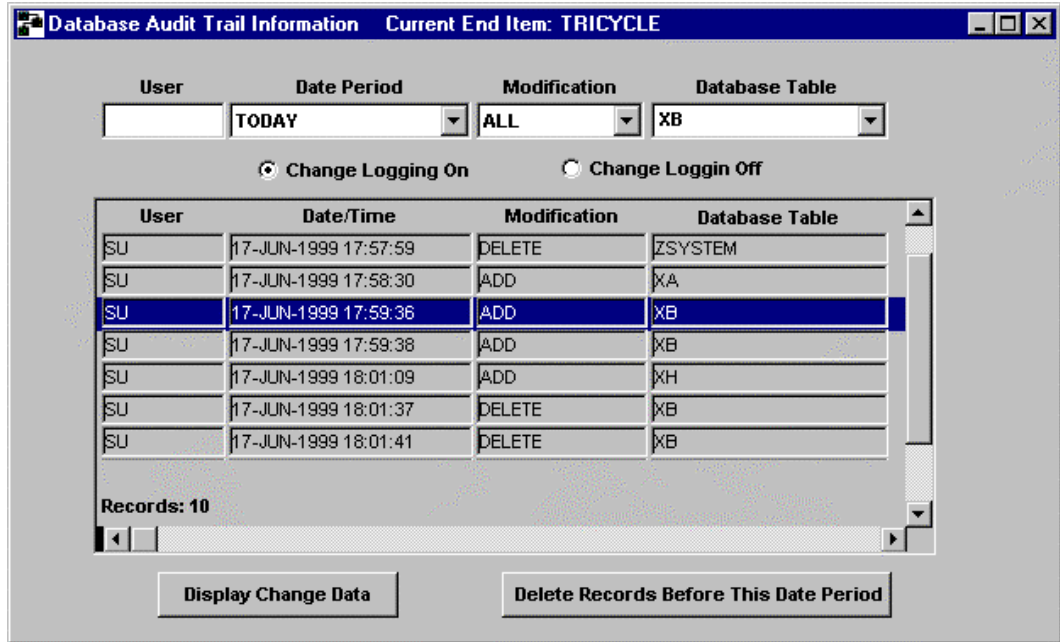


Figure 28 Database Audit Trail Information Window

STEP To view the database audit trail, perform the following steps as shown in Figure 28.

- Select the table for which the audit trail is to be viewed from the Database Table drop down list box
- Select the time period for which the audit trail is to be viewed from the Date Period drop down list box
- Select the Modification Type (Add, Change or Delete) from the Modification Type drop down list box
- Choose the **Execute** button on the Main Toolbar to retrieve change records

Once the changes are displayed, they can be viewed in more detail to find out exactly what was changed, added, or deleted. The **Display Change Data** button allows the user to view more detail about the change. The **Delete Records Before This Date Period** button allows cleanup to be performed on the change log file. This should be done periodically to prevent the change log file from growing to large.

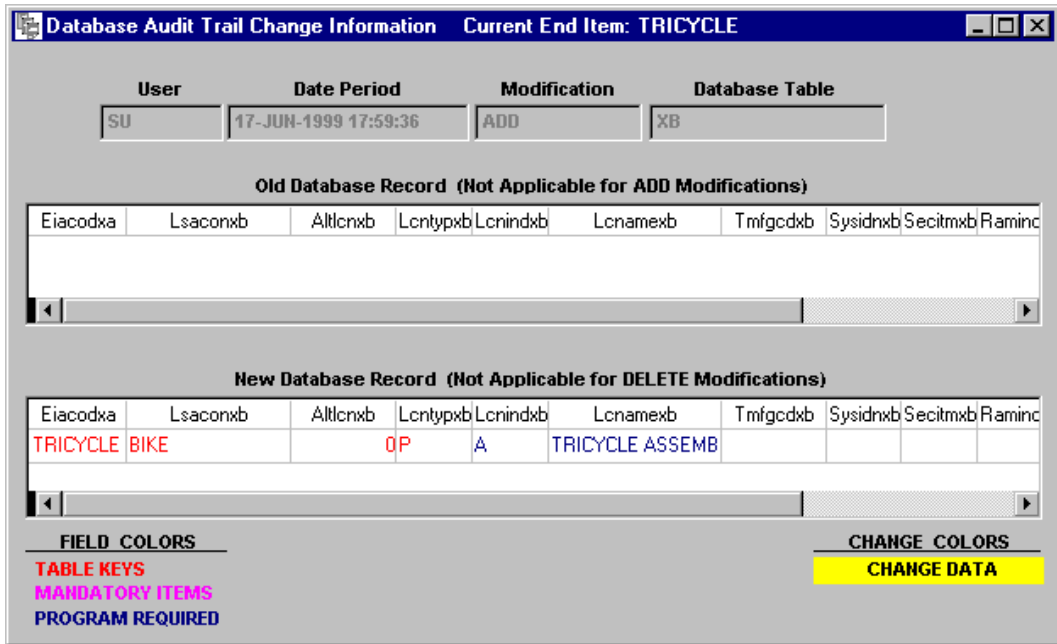


Figure 29 Database Audit Trail Change Information

STEP To view details on specific changes to the database perform the following steps as shown in Figure 29.

- Select the change record to be viewed (Figure 28)
- Choose the **Display Change Data** button
- When done viewing, close the Database Audit Trail Change Information Window
- Close the Database Audit Trail Information Window

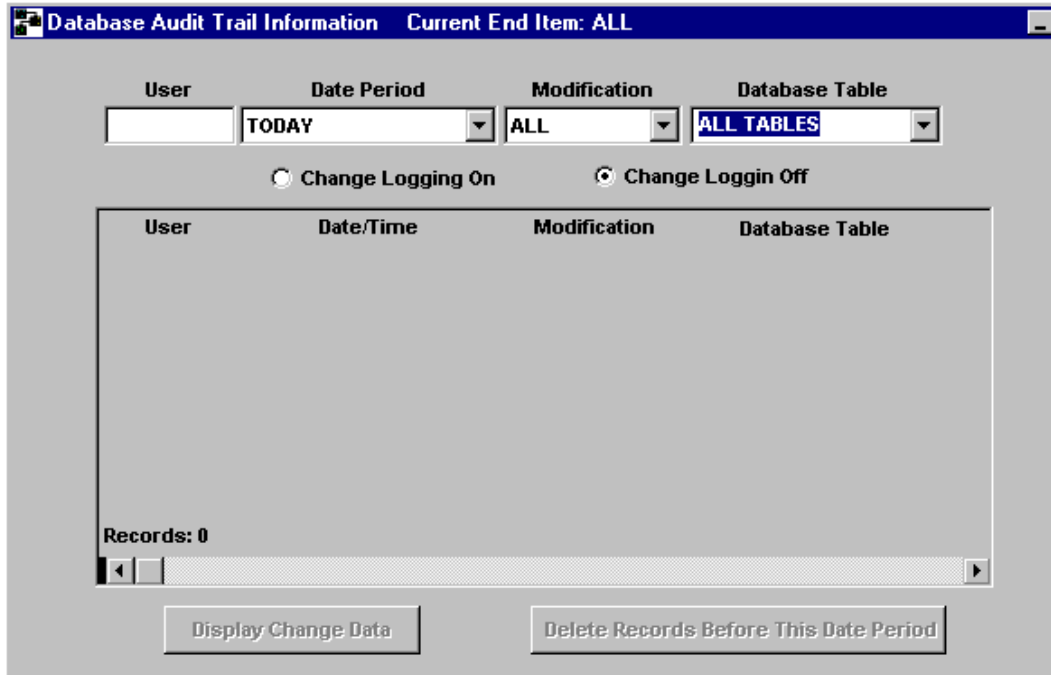


Figure 30 Database Audit Trail Information Window

## 2.11 ENABLING/DISABLING CHANGE LOGGING

Change logging can be enabled or disabled from the Database Audit Trail Information window shown in Figure 30 or from the System Defaults window shown in Figure 31.

To enable/disable change logging from the Database Audit Trail Information window, perform the following steps:

- Choose the appropriate radio button
- Choose the **Save** button on the Main Toolbar

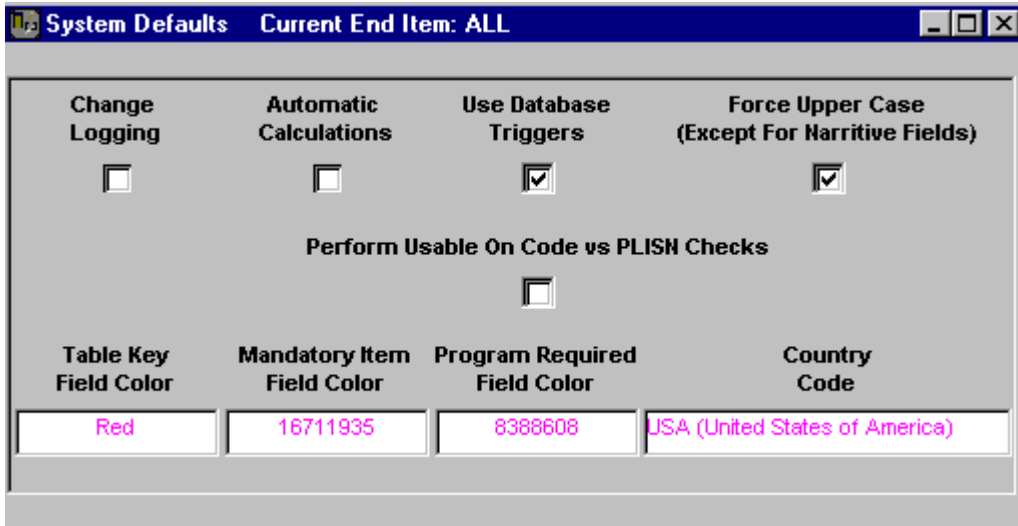


Figure 31 System Defaults Window

To enable/disable change logging from the System Defaults window, perform the following steps:

- Toggle the Change Logging checkbox on/off
- Choose the **Save** button on the Main Toolbar





# ***INDEX***

---



---

**INDEX**

- Adding Users, 2-4
  - Granting Privileges, 2-6
  - Revoking Privileges, 2-7
- Altering User Accounts, 2-14
- Auditing User Actions, 1-28
- Basic Concepts
  - Standard Oracle Objects, 1-5
  - User Connection Process, 1-10
- Change Logging
  - Enabling and Disabling, 2-22
- Cloning Users, 2-10
- Customizing the Navigator, 2-12
- Data Access and Control, 1-20
- Deleting Users, 2-9
- Disabling Change Logging, 2-22
- EAGLE Security Maintenance, 2-3
  - Adding Users, 2-4
    - Granting Privileges, 2-6
    - Revoking Privileges, 2-7
  - Altering User Accounts, 2-14
  - Change Logging
    - Disabling, 2-22
    - Enabling, 2-22
  - Cloning Users, 2-10
  - Customizing the Navigator, 2-12
  - Deleting Users, 2-9
  - Introduction, 2-3
  - Unlocking User Accounts, 2-18
  - Viewing Database Audit Trail Information, 2-19
  - Viewing User Roles, 2-16
- EAGLE Security Manager, 1-15
  - Random Password Generation and Encryption, 1-20
- Enabling Change Logging, 2-22
- Granting Privileges, 2-6
- Privileges
  - Granting, 2-6
  - Revoking, 2-7
- Random Password Generation and Encryption, 1-20
- Revoking Privileges, 2-7
- Security Examples, 1-28
- Special Login Accounts
  - Database Administrator (EAGLE), 1-4
  - Security Administrator (EAGLESA), 1-4
  - Superuser (SU), 1-5
- Standard Oracle Objects, 1-5
- Technical Support, 7
- Theory of Operation, 1-3
  - Auditing User Actions, 1-28
  - Basic Concepts, 1-5
    - Standard Oracle Objects, 1-5
    - User Connection Process, 1-10
  - Data Access and Control, 1-20
  - EAGLE Security Manager, 1-15
    - Random Password Generation and Encryption, 1-20
  - Introduction, 1-3
  - Presumptions, 1-4
  - Security Examples, 1-28
  - Security Needs Identified, 1-3
  - User Classification, 1-4
- Unlocking User Accounts, 2-18
- User Accounts. *See* Special User Accounts
  - Authorized Users, 1-5
- User Connection Process, 1-10
- Users
  - Adding, 2-4
  - Altering User Accounts, 2-14
  - Deleting, 2-9
  - Unlocking User Accounts, 2-18
  - Viewing User Roles, 2-16
- Viewing Database Audit Trail Information, 2-19
- Viewing User Roles, 2-16





***NOTES***

---











# Enhanced Automated Graphical Logistics Environment

Technical support is provided by the EAGLE Team of Raytheon Company. Phone support is available Monday through Friday from 8:00 a.m. to 4:30 p.m. Mountain Standard Time. EAGLE technical support personnel can be reached at (520) 663-6673. Training on the EAGLE product is available.

Are you ready for EAGLE? Join Team EAGLE and find out what it's like to soar. Give your logistics software product the EAGLE advantage. For more information on becoming part of Team EAGLE, contact:

Raytheon Company  
Team EAGLE  
(520) 663-6673  
email [raytheoneagle@west.raytheon.com](mailto:raytheoneagle@west.raytheon.com)

