CA
Identity Manager 12.5
Security Target

Version 2.0
June 21, 2010

Prepared for:
CA
100 Staples Drive
Framingham, MA 01702

Prepared by:
Booz Allen Hamilton
Common Criteria Testing Laboratory
900 Elkridge Landing Road, Suite 100
Linthicum, MD 21090-2950

Table of Contents

List of Figures

List of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation. This ST targets Evaluation Assurance Level 3 (EAL3).

1.1.1 ST Identification

ST Title:	CA Identity Manager Security Target 12.5
ST Version:	2.0
ST Publication Date:	June 21, 2010
ST Author:	Booz Allen Hamilton
Keywords:	Access Control, Enterprise Resource Management, Enterprise Resource Planning, Enterprise Identity Management, Identity Access Management, Identity Administration, Identity Manager, Single Sign-On

1.1.2 Document Organization

Table 1-1: ST Organization outlines the chapters and sections of the Identity Manager ST. This table is to be used by the reader as a quick reference guide for chapter descriptions and document navigation. The *Chapter* column identifies the chapter name, where as the *Section* column lists the sections within the chapter. Finally, the *Description* column provides a brief description of the topics covered in each respective *Chapter*.

Chapter	Section	Description
1. ST Introduction	Security Target, TOE, and CC Identification Security Target Organization Conformance Claims Conventions, Terminology, and Acronyms Security Target Overview	Provides introductory and identifying information for the Identity Manager ST.
2. Conformance Claims	CC version CC claims PP claims Package claims	Provides an overview of the claims against which the TOE is being made for the evaluation.
3. Security Problem Definition	Threats Organizational Security Policies Assumptions	Provides the security environment description in terms of Assumptions,

	Security Objectives	Threats, Objectives (both for the TOE and the Operational Environment), and Operational Security Policies.
4. Extended Security Functional Requirements	Extended SFRs for the TOE Extended SFRs for the Operational Environment	Identifies the extended security requirements for the TOE and Operational Environment.
5. Extended Security Assurance Requirements	N/A	Identifies the extended security requirements for the evaluation.
6. Security Functional Requirements	N/A	Provides the TOE security functional requirements that will be subject to evaluation.
7. Security Assurance Requirements	N/A	Identifies the security assurance requirements that will be used to perform the development and evaluation for the TOE work products.
8. TOE Summary Specification	Physical Boundary Logical Boundary	Provides a description of the scope of the evaluation for the TOE. Also describes the functions provided by the TOE to satisfy the security functional requirements.
9. TOE Summary Specification Rationale	N/A	Provides a summary mapping between the Security Functional Requirements for the TOE and the TOE's capabilities as described in the TOE Summary Specification.
10. Security Problem Definition Rationale	Security Objectives Rationale EAL3 Justification Strength of Function Rationale Requirement Dependency Rationale Security Functional Requirements Rationale	Provides a rationale for the chosen EAL, any deviations from CC Part 2 with regards to SFR dependencies, a strength of function rationale, and a mapping of threats to assumptions, objectives, and SFRs.
11. Assurance Measures	N/A	Identifies the items used to satisfy the Security Assurance Requirements for the evaluation.

Table 1-1: ST Organization

1.1.3 Terminology

This section defines the terminology used throughout this ST. The terminology used throughout this ST is defined in Table 1-2: Terminology Definitions. This table is to be used by the reader as a quick reference guide for terminology definitions.

Terminology	Definition
Account Template	A preconfigured set of privileges which can be assigned to an endpoint user's account on an endpoint during provisioning.
Admin Role	A subset of available administrative activities that can be defined and assigned to a TOE user.
Administrator	A TOE user who is assigned as an administrator of a group is able to control the membership of that group.
Connector	A piece of code that translates provisioning commands issued by Identity Manager into commands that can be interpreted by an endpoint.
Endpoint	A computer on the enterprise network that can have its accounts managed by Identity Manager. This can be system-based (i.e. a UNIX endpoint) or application-based (i.e. an LDAP endpoint)
Endpoint User	A user on the enterprise network that interacts with endpoints managed by the TOE. If an endpoint user has the ability to interact with the TOE, then they are also considered a TOE user.
Identity Manager	An integrated identity management platform that automates the creation, modification, suspension or deletion of user identities and their access to enterprise resources.
Management Console	The administrative interface which is used only in the initial configuration of Identity Manager.
Policy	A collection of one or more conditions for a role that combine to determine whether or not a user is assigned that role and what their scope within it is.
Provisioning	The act of using Identity Manager to create or modify user accounts on an endpoint as if an administrator on that endpoint was directly configuring it.
Provisioning Role	A set of account templates that are applied to a set of endpoints which can be defined and assigned to end users.
TOE User	Any trusted user on the TOE. All TOE users are capable of some administrative functionality (self-management at the very least).
User	A generic term to refer to all individuals belonging to an IT enterprise environment. All users are at the very least endpoint users, but can potentially be TOE users as well.
User Console	The administrative interface which is used to configure Identity Manager during its operation.
Workflow	The process of requiring approval to changes made in the configuration of Identity Manager.

Table 1-2: Terminology Definitions

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 1-3: Acronym Definitions. This table is to be used by the reader as a quick reference guide for acronym definitions.

Acronym	Definition
ACL	Access Control List

AES	Advanced Encryption Standard
CC	Common Criteria
CS	Connector Server
DB	Database
IM	Identity Manager
IT	Information Technology
JIAM	Java Identity and Access Management
LDAP	Lightweight Directory Access Protocol
OS	Operating System
PP	Protection Profile
RDBMS	Relational Database Management System
SOAP	Simple Object Access Protocol
ST	Security Target
TEWS	Task Execution Web Services
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function
XML	Extensible Markup Language

Table 1-3: Acronym Definitions

1.1.5 References

[1] CA Identity Manager 12.5 Release Notes
[2] CA Identity Manager 12.5 Implementation Guide
[3] CA Identity Manager 12.5 Installation Guide
[4] CA Identity Manager 12.5 Configuration Guide
[5] CA Identity Manager 12.5 Administration Guide
[6] CA Identity Manager 12.5 Provisioning Guide

1.2 TOE Reference

CA Identity Manager ® 12.5

NOTE: *The TOE must have the IMr12.5CommonCriteriaPatch applied to be in the evaluated configuration.*

1.2.1 TOE Overview

This Security Target (ST) defines the Information Technology (IT) security requirements for the CA Identity Manager. CA Identity Manager (IM) is an integrated identity management platform that automates the creation, modification, suspension or deletion of user identities and their access to enterprise resources. Through these functions, Identity Manager manages diverse user populations on a range of enterprise systems, from mainframes to web applications over a single tool. Identity Manager also provides TOE

users with the functionality to manage and delegate Password Management, Provisioning/Deprovisioning, and Identity Administration to the level deemed necessary.

The TOE:

- Provides a platform for access control to enterprise resources
- Propagates the creation, modification, and removal of enterprise user accounts across a heterogeneous collection of managed resources
- Offers delegated administration and user self-service to distribute management privileges based on organizational needs

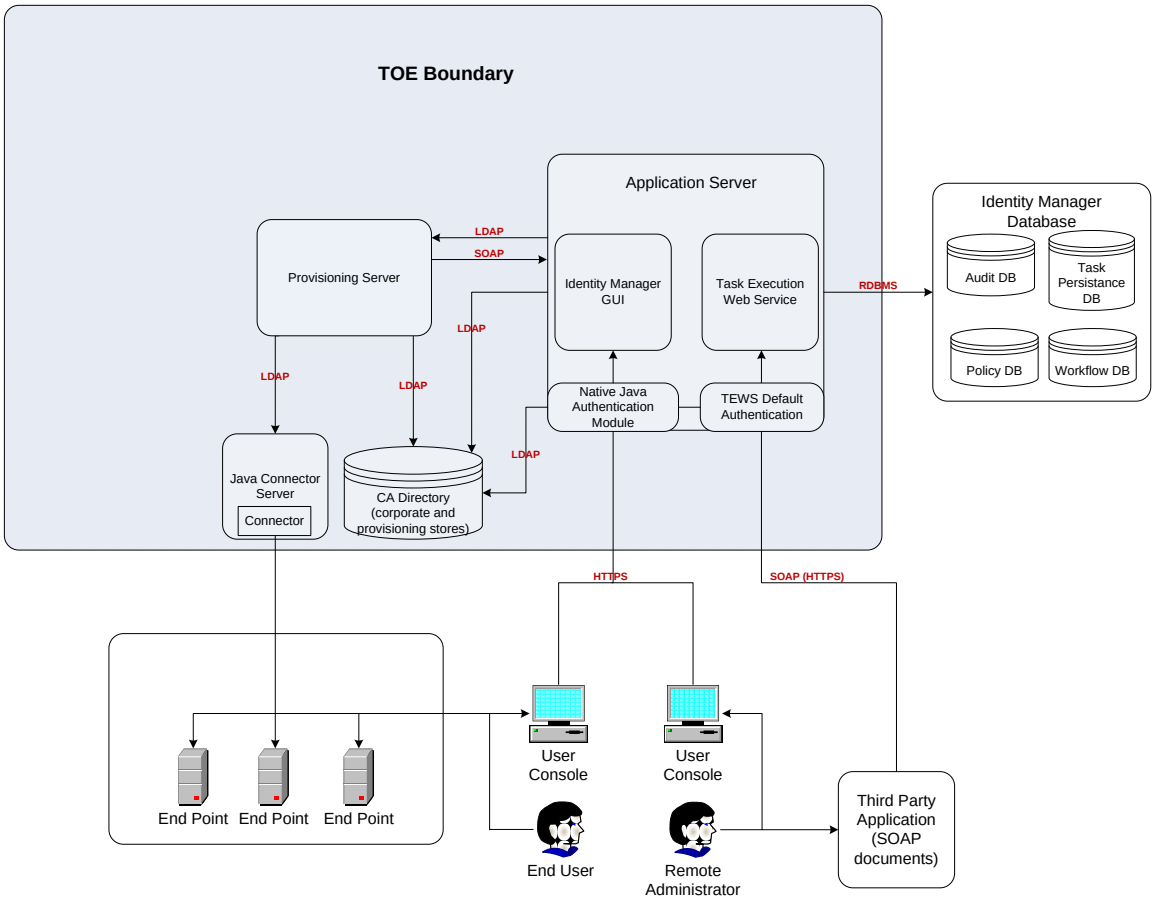

Figure 1-1: Major Components of CA Identity Manager

From this diagram, it can be seen that the TOE works in the following manner:

TOE users who wish to access the TOE via the Application Server for management functions authenticate to the Web-based User Console. This is accomplished via the Native Java Authentication Module, which does a credential check against the CA Directory corporate store. Once authenticated, the TOE user interacts with the User Console, which shows available functions by interacting with the Identity Manager GUI. Changes made via the User Console are propagated to the Identity Manager Database or to the Provisioning server, depending on the changes made. Throughout this process, the

remote communication is secured by encrypting HTTP data with an AES encryption of TLS.

If a remote TOE user wishes to execute a web service application that manages the TOE (for example, a macro that modifies the privileges of dozens of users automatically), they execute the third party SOAP application and the TEWS Default Authentication verifies the TOE user running the application has privileges to do so. Once approved, the Task Execution Web Service parses the SOAP application into a collection of commands and executes them all automatically.

NOTE: *In the evaluated configuration, the TEWS interface is expected to only be used by a user that has been assigned all privileges of the TOE.*

While most aspects of provisioning can be managed remotely, others require the presence of a TOE user accessing the Provisioning Server locally. In the evaluated configuration, the only provisioning aspects that will be used are the ones that can be managed remotely via the User Console. Provisioning is the process by which endpoint users, groups, or roles have their privileges defined on external servers, databases, or enterprise applications. By using the TOE to perform provisioning, it allows TOE users to modify configuration settings of multiple types of servers without issuing commands on those servers manually. This ensures that TOE users have a centralized point to control endpoint user access to internal servers, preventing redundancy, and easily identifying who should be able to do what on what machines.

The translation from provisioning commands in the TOE into information that reconfigures the endpoints is accomplished via connectors, which are hosted either on the connector server or on the managed endpoints themselves. However, only the former will be subject to evaluation because no trust can be placed on a connector running on an agent in the operational environment. There are different types of connectors for different types of endpoints. The role of the connector is to translate commands issued by the TOE into commands that modify the endpoints so that configuration changes can be made to the endpoints without accessing them directly. Once the provisioning has been accomplished, the settings on the managed endpoint will be changed as if they were performed by a human agent, and endpoint users can then access these endpoints with the privileges they were granted on them. Because these privileges are part of the environment once assigned, they continue to be enforced in the event of a service interruption of the TOE.

1.3 TOE Description

1.3.1 Physical Boundary

The following minimum components are required for the system that will host the Identity Manager servers:

Hardware Components

- CPU – one of the following:
  - Intel Core 2 Duo (or equivalent), 2 GHz
  - Dual-core SPARC, 1.5 GHz
- Memory: 4 GB
- Available disk space: 5 GB

Software Components

- Operating System: one of the following
  - Windows Server 2003 SP2
  - Windows Server 2003 R2 SP2
  - Windows Server 2008 (32-bit)
  - Solaris 9
  - Solaris 10
- ODBC Database: one of the following
  - Oracle 10g R2
  - Microsoft SQL Server 2005
- Application Server: JBoss 4.2.3 or WebLogic 9.2.3 for Solaris 10

> ***Note:*** *These hardware requirements take into account the requirements of the Application Server that must be installed on the system where Identity Manager is installed. The Provisioning Server will run on the same machine and the requirements for the Application Server are sufficient to accomplish this.*

In the evaluated configuration, the TOE will consist of one machine running the Application Server and another running the Provisioning Server, which includes the Java Connector Server. CA Directory will be used for the combined User Store and Provisioning Directory, and a third party application server is required to be installed on the Application Server prior to Identity Manager's installation.

In addition to the environmental components listed above, the following non-TOE software is required to run the TOE:

- TLS v1.0 implementation
- Transport standards HTTP, and FTP implementations
- SMTP implementation
- Web browser software

1.3.1.1 Identity Manager Application Server

The CA Identity Manager Application Server is a J2EE Enterprise Application that provides secure web-based TOE user interfaces, exposes identity management focused web services and provides the business rules engine (workflow). The exposed interfaces provide for centralized delegated administration of all identity management and provisioning tasks. CA Identity Manager Server also includes connectivity to the Provisioning Server.

After a third party application server such as JBoss is installed, the Identity Manager Installer is used to install all the software on the same system.

1.3.2 Logical Boundary

The logical boundaries of the TOE are described in the terms of the security functionalities that the TOE provides to the systems that utilize this product for endpoint user access control to protected resources.

The logical boundary of the TOE will be broken down into six security classes: Security Audit, Cryptographic Support, User Data Protection, Identification and Authentication, Security Management, and Trusted Path/Channel. Listed below are the security functions with a listing of the capabilities associated with them:

- Identification and Authentication – Defines password policies which control strength of per mutational access to the TOE. Provides identification and authentication of TOE users, as well as, determines the activities that require different which forms of authentication.
- Security Management – Controls how the modification of the TOE's behavior, TOE users, and other components are performed.
- Security Audit – Defines the events which are defined as auditable and the information that is written to the audit log as a result of these events.
- Data Protection – Determines what aspects of the TOE users are able to see and, within those aspects, the scope of control they have over other users.
- Cryptographic Communication – The ability to protect the confidentiality and integrity of data communicated between the TOE user, the TOE, and the external data stores.
- Provisioning – The ability to apply account templates on remote endpoints to endpoint users, modifying the configuration of those endpoints as if they were managed locally.

- Workflow – The ability to require explicit authorization of activities performed on the TOE after they are chosen by a TOE user but before they are executed by the TOE.

1.3.3 TOE Policies

Listed below are the three policies which the TOE uses to enforce access control to protected resources. They are defined in terms of the subjects upon which the policy is enforced, the attributes used the objects protected by the policy, and the operations the subjects perform on the objects.

User Policy – The User Policy is the policy by which the TOE allows or denies access to administrative functionality of the TOE. By performing an operation against an object governed by the User Policy, the subject is executing a task.

- Subjects – TOE users (with username, assigned roles, allowed tasks, and scope of control over these tasks as relevant security attributes).
- Objects – Configuration items used by the TOE and stored in the Identity Manager Database, user store, and endpoints. These include but are not limited to groups, roles, endpoint configurations, account templates, and user data (with ACL as relevant security attribute).
- Operations – The ability to create, modify, view, or delete the specified objects.

Workflow Policy – The Workflow Policy is the policy by which the TOE defines a delegated approval model for tasks performed by the User Policy. When a User Policy task (such as assigning a TOE user to a group) requires approval at one or more points in the process, the Workflow Policy is used.

- Subjects – TOE users (with username, allowed tasks, and assigned workflow approval role as relevant security attributes).
- Objects – Tasks governed by the User Policy for which workflow is enabled (with the workflow enable status, points in the task which require approval, and default approver as relevant security attributes).
- Operations – The ability to approve or reject the task (based on the subject being the object's default approver) and the ability to delegate this ability to another TOE user (based on whether or not the subject is assigned the delegation task).

Web Service Policy – The Web Service Policy is the policy by which the TOE allows or denies software agents the ability to perform automated instantiations of User Policy Tasks. The Web Service Policy is implemented by the Task Execution Web Service (TEWS).

- Subjects – Web Service applications that are run against the TOE (with the username and password of the user running the application as relevant security attributes).

- Objects – Configuration items used by the TOE and stored in the Identity Manager Database, User Store, and endpoints. These include but are not limited to groups, roles, endpoint configurations, account templates, and user data (with web service enable status and web service ACL as relevant security attributes).
- Operations – The ability to create, modify, view, or delete the specified objects.

1.4 Components of the TOE

An Identity Manager implementation, based on Figure 1-1, will include the following components:

Components:

- Servers
  - Identity Manager Application Server
  - Identity Manager Provisioning Server
- User Store
  - CA Directory (user store and provisioning directory)
- Connectors
  - Java Connector Server
  - Connectors
- Application Server Components
  - Identity Manager GUI – remote User Console
  - Task Execution Web Service (TEWS) Module
  - Native Java Authentication Module
  - TEWS Default Authentication Module

NOTE: *The TOE must have the IMr12.5CommonCriteriaPatch applied to be in the evaluated configuration.*

In addition, the following components belong to the operational environment:

- Third Party Applications (SOAP Documents)
- Managed Endpoints
- Databases
  - Policy Database
  - Task Persistence Database
  - Workflow Database
  - Audit Database

The TOE components provide the platform for the CA Identity Manager to manage the user populations on enterprise systems.

1.4.1 TOE Servers

The Identity Manager implementation for evaluation includes the following two types of servers:

- **Identity Manager Application Server**
  The Identity Manager Application Server executes tasks within Identity Manager. The J2EE Identity Manager application includes the Identity Manager Management Console and the Identity Manager User Console. It is also the primary interface to the environmental data stores, which assist in auditing and applying the tasks that are executed.

  The Application Server is ultimately responsible for determining the privileges available to a TOE user and only allowing that individual to access the parts of the TOE that they have been authorized to access.

- **Identity Manager Provisioning Server**
  The Provisioning Server manages accounts on endpoint systems. In the evaluated configuration, Identity Manager will support provisioning, so this is a required component.

  Note: *The Provisioning Directory must be installed on a CA Directory Server before installing the Provisioning Server. In the evaluated configuration, this Provisioning Directory will be the same logical CA Directory Server instance as the corporate user store.*

  The Provisioning Server is the server that manages additional accounts that are assigned to an endpoint user. When a provisioning role is assigned to an endpoint user, the Provisioning Server creates accounts on endpoints that meet the requirements of the role. For example, if a provisioning role is assigned to a user that includes an LDAP account template, the Provisioning Server assigns an LDAP account to the user. Basic management of provisioning roles and activities are accomplished through administrative use of the User Console. The Provisioning Server contains a Provisioning Manager GUI that allows for advanced management of provisioning functionality, but these features will not be subject to evaluation.

1.4.2 TOE User Directory

An Identity Manager implementation must include a user store that contains the identities that Identity Manager maintains. It is used for the purposes of authenticating to the TOE and delivering information to the internal security model, which then authorizes access to protected data. Typically, this is an existing user store that an enterprise utilizes to store information about its users, such as employees and customers. In the evaluated configuration, this will be an instantiation of CA Directory.

When provisioning is used (as it is in the evaluated configuration), Identity Manager also requires a provisioning directory that includes global users, which are associated with accounts on endpoints such as LDAP, Oracle, and SAP.

To provide options for managing users and automatic provisioning of additional accounts for those users, Identity Manager coordinates two user stores:

- The Identity Manager corporate directory, the user store maintained by Identity Manager. Typically, this is an existing store that contains the user identities that a company needs to manage.

  The user store can be an LDAP directory or a relational database.

  In the Management Console, the admin installing the TOE must create an Identity Manager Directory object to connect to the user store and to describe the user store objects that Identity Manager will maintain.

- The Provisioning Directory, the user store maintained by the Provisioning Server. It is an instance of CA Directory and includes global user accounts, which associate users in the Provisioning Directory with accounts on endpoints such as LDAP, Oracle, and SAP.

  Only some users have a corresponding global user account. The users are known as endpoint users. When a user receives a provisioning role, the Provisioning Server creates a global user in the Provisioning Directory, designating them as an endpoint user.

In the evaluated configuration, these two user stores will be the same logical instance of CA Directory. The corporate directory would traditionally be regarded as a component of the operational environment. However, the setup of the TOE will incorporate this directory into the TOE in order to manage provisioning. This is why the corporate directory cannot be considered to be part of the environment in this situation.

1.4.3 TOE Databases

The evaluated configuration for Identity Manager will include the following data stores to support Identity Manager functionality:

- Policy Database
- Task Persistence Database
- Workflow Database
- Audit Database

For more information regarding these databases, refer to the Identity Manager Database section.

1.4.4 Connectors

A connector is the software interface to an endpoint. The Provisioning Server uses the connector to communicate with the endpoint. It translates Provisioning Server actions into changes on the endpoint, such as "Create a new dba level account on an Oracle endpoint."

Examples of endpoints are LDAP server, Oracle database, or SAP enterprise software.

Connectors work with multiple endpoints. For example, if there are many UNIX workstation endpoints in the environment, there could be one UNIX connector on the Connector Server that is able to manage these workstations from a centralized point. Another connector might handle all connectors that request Windows accounts.

A Connector Server is a Provisioning Server component that manages connectors. All connectors will have a component that runs on and the Connector Server. However, some connectors also have a component that must be present on the managed endpoint in order for provisioning to be accomplished. For this evaluation, all connectors are within the scope, but the communication between connector components on the Connector Server and those that also run remotely will not be evaluated.

There are two types of connector servers:

- The Java Connector Server (JCS) manages connectors written in Java
- The C++ Connector Server (CCS) manages connectors written in C++

Note that for this evaluation, the JCS is the only connector service which is within the scope of the evaluation. The CCS has been listed only for informational purposes and will not be evaluated. The TOE has no assurance of the integrity of these agents. Because they are installed on systems that are outside the TOE boundary, an administrator has no capability to protect these agents from modification.

1.4.5 Identity Manager Management Console

The Identity Manager system administrator’s responsibilities include the following:

- Creating an Identity Manager directory
- Configuring an Identity Manager environment
- Using the system manager (superuser) account to configure initial TOE users and their authority
- Enabling custom features for initial use

To configure an Identity Manager environment, use the Management Console, a Web-based application.

The management console is divided into the following two sections:

- Directories—Use this section to create and manage Identity Manager directories, which describe user stores to Identity Manager.
- Environments—Use this section to create and manage Identity Manager environments, which control the management and configuration of a directory.

The Management Console is only used for initial configuration of the TOE. Once operational, the TOE management functions are handled by TOE user usage of the User Console.

1.4.6 Provisioning in an Identity Manager Environment

Provisioning can be configured for an Identity Manager environment to provide accounts in other systems, called endpoints, to endpoint users. Accounts provide endpoint users with access to additional resources, such as an email account. These access rights are granted by assigning provisioning roles, which are created through Identity Manager. Provisioning roles are associated with account templates that define accounts that endpoint users can receive.

When a provisioning role is assigned to an endpoint user, they receive the accounts defined by the account templates in the role. This can be applied to a newly-created account or to modify a pre-existing one. The account templates also define how user attributes are mapped to accounts. The accounts exist in managed endpoints defined by the policies.

Once the roles have been provisioned, the managed endpoints retain their configuration information as if they were configured directly on the system as opposed to by the TOE, and an endpoint user can interact directly with these endpoints using the provisions issued by the TOE, even if the TOE is not running at that time.

1.4.7 Task Execution Web Service

The Task Execution Web Service (or TEWS) is a web service API for Identity Manager which allows TOE users to develop applications that effectively automate a series of Identity Manager administrative commands. For example, a "hire employee" application could be written which creates endpoint user accounts and assigns the proper privileges to them via the Connector Server without requiring a TOE user to make each of those assignments individually.

In the evaluated configuration, the TEWS interface requires a third party application to authenticate users prior to granting access to the interface. The user will then provide their identity to the TOE for identification and to determine access control restrictions. Thus, TEWS interface will only identify a user that has already been granted access to the

interface by the third party application. The identity provided to the TOE does not have to match the one provided to the third party application, nor will the identity provided to the TOE be authenticated. Therefore, the TEWS interface is expected to only be used by a user that has been assigned all privileges of the TOE in the evaluated configuration.

NOTE: *Although it was not validated through the evaluation, the vendor has asserted that the TOE can have the TEWS interface protected by CA SiteMinder. This would allow for TOE users to have their TOE identity be authenticated by SiteMinder, and then have all actions on the TOE be associated with their validated TOE identity.*

1.5 TOE Security Environment

The TOE is an application installed on a Windows or Solaris OS and controls access to resources through identification, authentication, and authorization. Once access privileges to managed endpoints have been established, the TOE relies on the endpoints themselves to enforce those privileges. For example, if the TOE is used to give an endpoint user certain privileges on an LDAP server, those privileges will be issued by the TOE via the connector service, the connector will configure the LDAP server as if the privileges had been granted there manually.

Each Identity Manager environment requires one or more superusers to customize the initial roles and tasks using the User Console. Once a superuser creates the initial roles and tasks, that manager can grant administrative privileges to TOE users in that environment. These TOE users become managers of TOE data such as users, groups, and endpoints. Once sufficient privileges have been distributed across TOE users such that all relevant operational needs can be managed during initial configuration, the superusers are disabled.

1.6 Excluded from the TOE

Requires separate installation, not part of IM by standard installation

- WorkPoint Designer – This is a separate product which provides an optional graphical frontend for workflow management, which is currently managed via IM GUI.
- SiteMinder (for authentication) – SiteMinder is a separately licensed product that is not installed with Identity Manager by default. If the TEWS interface is used, SiteMinder is one potential application that can be deployed in the environment which can be used to protect it.
  NOTE: *Although it was not validated through the evaluation, the vendor has asserted that the TOE can have the TEWS interface protected by CA SiteMinder. This would allow for TOE users to have their TOE identity be authenticated by SiteMinder, and then have all actions on the TOE be associated with their validated TOE identity.*

- Graphical Identification and Authorization (GINA) – GINA is a separate component used for password reset when integrated with Microsoft Windows. Password reset is already facilitated by the TOE via public tasks.
- Cube Browser – This is only used when GINA is also used.
- Credential Provider – This is the Windows Vista equivalent of GINA.
- PAM Extensions – PAM (pluggable authentication module) extensions are separately installed components which would allow users to authenticate to the TOE using password policies derived from other machines. The TOE boundary requires the presence of an LDAP directory for user and provisioning storage, so native authentication against the LDAP directory is the expected method of authentication.
- IAM Report Server and Reporting Database – This is separately installed and configured executable, Tomcat server, and database which is optionally used to provide more in-depth audit review functionality.
- Password Synchronization Agents (PSAs) – PSAs are separate endpoint agents which reside on machines in the environment.
- C++ Connector Server (including remote agents) – C++ connectors are separate endpoint agents on environmental machines. This differs from Java connectors which reside within the TOE boundary and issue commands remotely.
- Audit Settings File – This file is used for exporting audit data to SiteMinder or some other source beyond the default Audit DB.
- Directory Structure – A directory structure is defined when there are multiple Organizations which are being integrated using IM. Based on the TOE including an LDAP directory, the directory structure is derived from the existing structure of that LDAP directory. As a result, the administrator can opt out of configuring this.
- Directory Configuration File – A directory configuration file is used during initial setup of the TOE to specify the organizational structure which Identity Manager will use. In the case of the evaluated configuration, the LDAP directory which is part of the TOE boundary is used to define this structure. As a result, the administrator can opt out of configuring this.
- SPML – SPML is an optional service used to provision using SPML requests rather than the IM GUI.

Installed by default but can be disabled following installation

- System Manager role – This is a superuser role which can be removed (or assigned to no users) following installation of the TOE. This is excluded to encourage administrative separation of duties.
- Access Role Manager role – Access roles are only used for SiteMinder integration, which is excluded. This role can be configured to not be assigned to any user or it can be removed altogether.
- Access roles – access roles are only used for SiteMinder integration, which is excluded.

Included as part of the product but cannot be disabled

- etautil – This utility is a local interface to Provisioning Server. It is not required for use because duplicate functionality is made available via the IM GUI.
- Provisioning Manager GUI – As of release 12 of IM the Provisioning Manager GUI features are part of the IM GUI as well. In future releases this GUI will be removed altogether. Because it is recommended that administration be done from a single point, this functionality is not being examined.
- Management Console – The Management Console is only used for initial installation of the TOE. Its responsibilities include setting up a directory configuration file, configuring an environment, and assigning an initial administrator. Once operational, the IM GUI should be used to perform management functions.

1.7 TOE Type

CA Identity Manager 12.5 provides the following: System Access Control.

2 Conformance Claims

2.1 CC Version

This ST is compliant with *Common Criteria for Information Technology Security Evaluation*, CCMB-2007-09-004, Version 3.1 Revision 2, September 2007

2.2 CC Part 2 Extended

This ST and Target of Evaluation (TOE) is Part 2 extended for EAL3 to include all applicable NIAP and International interpretations through 13 May 2008.

2.3 CC Part 3 Conformant Plus Flaw Remediation

This ST and Target of Evaluation (TOE) is Part 3 conformant plus flaw remediation for EAL3 to include all applicable NIAP and International interpretations through 13 May 2008.

2.4 PP Claims

This ST does not claim Protection Profile (PP) conformance.

2.5 Package Claims

This TOE has a package claim of EAL 3.

2.6 Package Name Conformant or Package Name Augmented

This ST and Target of Evaluation (TOE) is conformant to EAL package claims augmented with ALC_FLR.1 and ASE_TSS.2**.**

2.7 Conformance Claim Rationale

There is no Conformance Claim rationale for this ST.

3 Security Problem Definition

3.1 Threats

The TOE itself has threats and the TOE is also responsible for addressing threats to the environment in which it resides. The assumed level of expertise of the attacker for all the threats is unsophisticated. The following are threats addressed by the TOE.

T.ACCESS	TOE users could gain electronic access to protected resources by attempting to establish a connection that they are not permitted to perform.
T.ADMIN_ERROR	A TOE user may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.
T.AUDIT_COMPROMISE	A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a TOE user's action.
T.EAVESDROPPING	A malicious user could eavesdrop on network traffic to gain unauthorized access to TOE data.
T.MASK	Users whether they be malicious or non-malicious, could gain unauthorised access to the TOE by bypassing identification and authentication countermeasures.
T.MASQUERADE	A user may masquerade as an TOE user or an authorized IT entity to gain access to data or TOE resources.
T.UNAUTH	Users could gain unauthorised access to the TOE or its data stores by bypassing identification and authentication requirements.

3.2 Organizational Security Policies

There are no Organizational Security Policies that apply to the TOE.

3.3 Assumptions

The specific conditions listed in this section are assumed to exist in the environment in which the TOE is deployed. These assumptions are necessary as a result of practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

3.3.1 Personnel Assumptions

A.ADMIN One or more TOE users will be assigned to install, configure and manage the TOE.

A.PATCHES Users responsible for management of the operational environment exercise due diligence to update the TOE with the latest patches and patch the Operational Environment (e.g., OS and database) so they are not susceptible to network attacks.

A.NOEVIL TOE users are not careless, wilfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.

3.3.2 Physical Assumptions

A.LOCATE The TOE and the endpoints the TOE will monitor and manage are located on a network that is isolated from any other network. No connections exist to other networks.

3.4 Security Objectives

3.4.1 Security Objectives for the TOE

The following security objectives are to be satisfied by the TOE.

O.ACCESS The TOE will provide measures to authorize TOE users to access specified resources once the user has been authenticated. TOE user authorization is based on access rights configured by other TOE users with the ability to configure them.

O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.

O.AUTH The TOE will provide measures to uniquely identify all TOE users and will authenticate the claimed identity prior to granting a user access to the resources protected by the TOE.

O.FILESYS The security features offered by the TOE protects the confidentiality of TOE data that is stored in the database.

O.MANAGE The TOE will provide TOE users with the resources to manage and monitor user accounts, resources, and security information relative to the TOE.

O.ROBUST_ADMIN_GUIDANCE The TOE will provide TOE users with the necessary information for secure delivery and management.

O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.

O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a TOE user's logical access to the TOE and to explicitly deny access to specific TOE users when appropriate.

3.4.2 Security Objectives for the operational environment of the TOE

The following security objectives for the Operational environment of the TOE must be satisfied in order for the TOE to fulfill its security objectives.

OE.ADMIN One or more TOE users will be assigned to install, configure and manage the TOE.

OE.AUDIT The operational environment will provide a secure mechanism by which audit data can be reviewed.

OE.FILESYS The security features offered by the underlying Operating System and Database protect the files used by the TOE and access to the audit records.

OE.NOEVIL No TOE users are careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.

OE.LOCATE The TOE will be located on an isolated network with no connections to other networks.

4 Extended Security Functional Requirements

4.1 Extended Security Functional Requirements for the TOE

There are no extended Security Functional Requirements for the TOE in this ST.

4.2 Proper Dependencies

Because there are no extended Security Functional Requirements for the TOE, there are no additional dependencies which must be followed.

5 Extended Security Assurance Requirements

There are no extended Security Assurance Requirements in this ST.

6 Security Functional Requirements

6.1 Security Functional Requirements for the TOE

The following table provides a summary of the Security Functional Requirements implemented by the TOE.

Security Function	Security Functional Components
User Data Protection (FDP)	FDP_ACC.1(1) Subset Access Control
	FDP_ACC.1(2) Subset Access Control
	FDP_ACC.1(3) Subset Access Control
	FDP_ACF.1 (1) Security Attribute Based Access Control
	FDP_ACF.1 (2) Security Attribute Based Access Control
	FDP_ACF.1 (3) Security Attribute Based Access Control
	FDP_IFC.1 Information Flow Control
	FDP_IFF.1 Simple Security Attributes
Identification and Authentication (FIA)	FIA_ATD.1 User Attribute Definition
	FIA_SOS.1 Verification of Secrets
	FIA_UAU.2 User authentication before any action
	FIA_UID.2 User Identification Before Any Action
Security Audit (FAU)	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
Security Management (FMT)	FMT_MSA.1 Management of Security Attributes
	FMT_MSA.3 Static attribute initialization

Security Function	Security Functional Components
	FMT_MTD.1(1) Management of TSF Data
	FMT_MTD.1(2) Management of TSF Data
	FMT_MTD.1(3) Management of TSF Data
	FMT_SMF.1 Specification of Management Functions
	FMT_SMR.2 Restrictions Security Roles
	FMT_REV.1 Revocation
Cryptographic Support (FCS)	FCS_CKM.1 Cryptographic Key Generation
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1 Cryptographic Operation

Table 6-1: Security Functional Requirements for the TOE

6.1.1 Class FAU: Security Audit

6.1.1.1 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;
b) All auditable events for the [***not specified***] level of audit; and
c) [***all operations listed in Table [6-2] Auditable Events***].

AddGrantorOnAdminRoleEvent
AddGrantorOnProvisioningRoleEvent
AddGroupAdminEvent
AddGroupAdminGroupEvent
AddGroupToGroupEvent
AddToGroupEvent
AssignAdminRoleEvent
AssignProvisioningRoleEvent
CertificationNonCertifiedActionCompletedNotificationEvent
CertificationNonCertifiedActionPendingNotificationEvent
CertificationRequiredFinalReminderNotificationEvent
CertificationRequiredNotificationEvent
CertificationRequiredReminderNotificationEvent
CertificationStatusCertifiedEvent
CertificationStatusInCertificationEvent
CertificationStatusNotCertifiedEvent
CertificationStatusRequiresCertificationEvent
CertifyRoleEvent

CreateAdminRoleEvent
CreateAdminTaskEvent
CreateGroupEvent
CreateIdentityPolicySetEvent
CreateLAHDefinitionEvent
CreatePasswordPolicyEvent
CreateProvisioningRoleEvent
CreateProvisioningUserAuditEvent
CreateProvisioningUserNotificationEvent
CreateUserEvent
DeleteAdminRoleEvent
DeleteAdminTaskEvent
DeleteGroupEvent
DeleteIdentityPolicySetEvent
DeleteLAHDefinitionEvent
DeletePasswordPolicyEvent
DeleteProvisioningRoleEvent
DeleteProvisioningUserAuditEvent
DeleteUserEvent
DisableUserEvent
EnableUserEvent
ExternalTaskEmptyEvent
ExternalTaskGroupEvent
ExternalTaskOrgEvent
ExternalTaskUserEvent
ForgottenPasswordAuditEvent
ForgottenPasswordEvent
ForgottenUserIDEvent
GenericAuditEvent
ModifyAdminRoleEvent
ModifyAdminTaskEvent
ModifyGroupEvent
ModifyIdentityPolicySetEvent
ModifyLAHDefinitionEvent
ModifyPasswordPolicyEvent
ModifyProvisioningRoleEvent
ModifyProvisioningUserAuditEvent
ModifyUserEvent
RemoveFromGroupEvent
RemoveGrantorOnAdminRoleEvent
RemoveGrantorOnProvisioningRoleEvent
RemoveGroupAdminEvent
RemoveGroupAdminGroupEvent
RemoveGroupFromGroupEvent
ResetPasswordEvent
RevokeProvisioningRoleEvent
SelfRegisterUserEvent

SetPrimaryObjectAuditEvent
SynchronizeUserAccountsEvent
SynchronizeUserEvent
SynchronizeUserProvisioningRolesAddAccountsEvent
SynchronizeUserProvisioningRolesDeleteAccountsEvent
UserAttributeAddValueEvent
UserAttributeRemoveValueEvent
ViewAdminRoleEvent
ViewAdminTaskEvent
ViewGroupEvent
ViewLAHDefinitionEvent
ViewPasswordPolicyEvent
ViewProvisioningRoleEvent
ViewUserEvent

Table 6-2: Auditable Events

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [***contents as listed in Table 6-3a through 6-3e***].

Column	Column type	Description
ID	number	Id of audited event
PARENT_TS_OID	varchar2(100)	Id of the parent task session (nested task cases)
PARENT_EVENT_OID	varchar2(100)	Id of the parent event
AUDIT_TIME	date	Timestamp of the audit item
TASKSESSION_OID	varchar2(100)	Id of the task session
ADMIN_DN	varchar2(512)	User DN of the administrator executing the task containing the event
ADMIN_NAME	varchar2(255)	Friendly Name of the administrator executing the task containing the event
TASK_NAME	varchar2(255)	Friendly name of the task
TASK_TAG	varchar2(255)	Unique tag name of the task
TASK_DESCRIPTION	varchar2(4000)	Task Description
TASK_PRIORITY	number	Scheduled priority of the task
STATE	varchar2(100)	Current task's state
ENVNAME	varchar2(100)	Friendly Name of the IM Environment for the task
ENV_OID	varchar2(100)	ID of the IM Environment for the task

Table 6-3a: Audit Record Columns for Task Sessions

Column	Column type	Description
ID	number	Id of the event
TASKSESSION_ID	number	Id of the task session to which the event belongs
PARENT_TS_OID	varchar2(100)	Id of the parent task session (nested task cases)
PARENT_EVENT_OID	varchar2(100)	Id of the parent event
AUDIT_TIME	Date	Timestamp of the audit item
EVENT_OID	varchar2(100)	
ADMIN_DN	varchar2(512)	User DN of the administrator executing the task containing the event
ADMIN_NAME	varchar2(255)	Friendly Name of the administrator executing the task containing the event
EVENT_NAME	varchar2(255)	Name of the event
EVENT_DESCRIPTION	varchar2(4000)	Description of the event
EVENT_STATE	varchar2(100)	Event's current state
ENVNAME	varchar2(100)	Friendly Name of the IM Environment for the task
ENV_OID	varchar2(100)	ID of the IM Environment for the task

Table 6-3b: Audit Record Columns for Audit Events

Column	Column type	Description
ID	number	ID of audited event
PARENT_EVENT_ID	number	Id of the parent event
AUDIT_TIME	date	Timestamp of the attribute change
OBJECT_TYPE	varchar2(100)	Managed Object Type associated with the event
OBJECT_NAME	varchar2(255)	Name of the managed object instance associated with the event.

Table 6-3c: Audit Record Columns for Event Objects

Column	Column type	Description
ID	number	ID of audited event
PARENT_EVENT_ID	number	ID of event which caused relationship change
AUDIT_TIME	Date	Timestamp of the attribute change
OBJECT_TYPE	varchar2 (100)	Managed Object Type of the relationship associated with the event
OBJECT_DN	varchar2 (512)	Unique name of the managed object instance used in the relationship associated with the event
CONTAINER_TYPE	varchar2 (100)	Relationship object container type (if applicable)
OBJECT_NAME	varchar2 (255)	Friendly name of the relationship managed object
CONTAINER_NAME	varchar2 (255)	Friendly name of the relationship object's container
CONTAINER_DN	varchar2 (512)	Unique name of the relationship object's container
OPERATION	varchar2 (50)	Relationship operation (i.e. assign/remove)

Table 6-3d: Audit Record Columns for Object Relationship Changes

Column	Column type	Description
ID	number	ID of audited event
PARENT_OBJECT_ID	number	ID of object for which attributes have been changed
AUDIT_TIME	date	Timestamp of the attribute change
ATTRIBUTE_ NAME	varchar2(255)	Name of managed object's attribute
ATTRIBUTE_ OLDVALUE	varchar2(4000)	Original value of managed object's attribute
ATTRIBUTE_ NEWVALUE	varchar2(4000)	New value of managed object's attribute

Table 6-3e: Audit Record Columns for Object Attribute Changes

Application Note: *A Task Session refers to an instance of a task. One or more events will occur during the execution of this task. Events will affect objects,*

Dependencies:	FPT_STM.1 Reliable time stamps

6.1.1.2 FAU_GEN.2 User identity association

Hierarchical to:	No other components.
FAU_GEN.2.1	For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.
Dependencies:	FAU_GEN.1 Audit data generation FIA_UID.1 Timing of identification
Application Note:	*This requirement is accomplished via AuditProfileAttribute Elements.*

6.1.2 Class FCS: Cryptographic Support

All cryptography for this product has only been asserted as tested by the vendor. The testing of the specific cryptographic algorithms will not be tested as part of this evaluation.

6.1.2.1 FCS_CKM.1 Cryptographic Key Generation

Hierarchical to:	No other components.
FCS_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [***AES***] and specified cryptographic key sizes [***256 bits***] that meet the following: [***FIPS PUB 197***].
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction

6.1.2.2 FCS_CKM.4 Cryptographic Key Destruction

Hierarchical to:	No other components.
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***overwrite***] that meets the following: [***no standard***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]

6.1.2.3 FCS_COP.1 Cryptographic Operation

Hierarchical to: No other components.

FCS_COP.1.1 The TSF shall perform [***encryption of TOE user sessions, encryption of directory or environment export, encryption of new TOE user passwords in the task session for a create task, encryption of database fields marked as requiring encrypt on write***] in accordance with a specified cryptographic algorithm [***AES***] and cryptographic key sizes [***256 bits***] that meet the following: [***FIPS PUB 197***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

6.1.3 Class FDP: User Data Protection

6.1.3.1 FDP_ACC.1(1) Subset Access Control

Hierarchical to: No other components.

FDP_ACC.1.1(1) The TSF shall enforce the [***User Policy***] on [***TOE users who are assigned the tasks as specified in Table 6-8***].

Admin Role	**Available Tasks (operation + object)**
Admin Role Manager	Create Admin Role Create Admin Task Delete Admin Role Delete Admin Task Modify Admin Role Modify Admin Role Members/Administrators Modify Admin Task View Admin Role View Admin Role Members/Administrators View Admin Task
Certify Manager	Certify User
Certification Process Manager	Begin Certification Process End Certification Process Send Certification Reminder Notification Send Final Certification Reminder Notification

Delegation Manager	A Delegation Manager can act on behalf of another user (the delegator) and delegate work item approval to tasks a third user (the delegate). This role has scope over all users.
Group Manager	Create Group Delete Group Modify Group View Group
Password Manager	Reset User Password View User
Provisioning Role Manager	Create Provisioning Role Delete Provisioning Role Modify Provisioning Role Modify Provisioning Role Members/Administrators View Provisioning Role View Provisioning Role Members/Administrators
Security Manager	Enable/Disable User Reset User Password View User
Self Delegator	Out of Office In addition, a Self Delegator (the delegator) can specify that another user (the delegate) be allowed to approve tasks in the delegator's work list. This role has scope over all users.
Self Manager	Change My Password Modify My Groups Modify My Profile View My Roles View My Submitted Tasks View My Work List
User Manager	Create User Delete User Modify User View User Modify Group Members
(no role)	Create Endpoint Delete Endpoint Modify Endpoint View Endpoint Create Account Template Delete Account Template Modify Account Template View Account Template Create Identity Policy Set Delete Identity Policy Set Modify Identity Policy Set

	View Identity Policy Set Synchronize Users Create Password Policy Delete Password Policy Modify Password Policy View Password Policy Connection Management Select Box Data View Submitted Tasks Create Logical Attribute Handler Delete Logical Attribute Handler Modify Logical Attribute Handler View Logical Attribute Handler Approve Modify Admin Role Membership Approve Create Group Approve Delete Group Approve Modify Group Membership Approve Modify Provisioning Role Membership Approve Self Registration Approve Create User Approve Delete User Approve Modify User

Table 6-4: Default Admin Tasks

Dependencies: FDP_ACF.1 Security attribute based access control

Application Note: *The admin role is defined as the subject, this is based on the process within the TOE that determines the login context of the operations performed. Specific information regarding this process will be discussed in the FSP/TDS.*

Application Note: *Not all default tasks are mapped to pre-existing roles by default. In order to use these tasks, a TOE user with the Create Admin Role task authority must create a role that contains one or more of the tasks and assign one or more users to it.*

Application Note: *Custom admin roles with access to a set of tasks may also be specified. TOE users who are assigned custom roles will only have access to tasks specified in the definition of these roles Defining and assigning custom roles is how the tasks associated to (no role) can be performed.*

6.1.3.2 FDP_ACC.1(2) Subset Access Control

Hierarchical to: No other components.

FDP_ACC.1.1(2) The TSF shall enforce the **[*Workflow Policy***] on [***all TOE users***].

Application Note: *For each approval operation that is allowed for a task, a reject operation is also allowed.*

Dependencies: FDP_ACF.1 Security attribute based access control

6.1.3.3 FDP_ACC.1(3) Subset Access Control

Hierarchical to: No other components.

FDP_ACC.1.1(3) The TSF shall enforce the [***Web Service Policy***] on [***all TOE users***].

Dependencies: FDP_ACF.1 Security attribute based access control

Application Note: *For the Web Service Policy, the subjects are TEWS applications, and the objects and operations are identical to those in the User Policy. TEWS applications are executed on behalf of a user whose identity is supplied by the user after a separate identity has been validated via an appropriate third party mechanism.*

6.1.3.4 FDP_ACF.1 (1) Security Attribute Based Access Control

Hierarchical to: No other components.

FDP_ACF.1.1 (1) The TSF shall enforce the [***User Policy***] to objects based on the following: [

- ***The allowed task(s) assigned to an admin role***
- ***The admin role(s) assigned to a TOE user***
- ***The scope of the admin role(s) assigned to a TOE user***].

FDP_ACF.1.2 (1) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- ***A TOE user is implicitly granted access to a task if he/she has an admin role attribute that authorizes the task.***
- ***A TOE user is explicitly granted access to the data a task uses based on the scope determined by the policy which determines their membership .***

]

Application Note: *To clarify, policies which determine whether or not a TOE user belongs to a role can differ in terms of the scope offered by that policy. For example, one policy for the Manage Users task may give a TOE user the ability to manage all users if that TOE user meets the membership condition, while a second policy could be more limited, such as giving a TOE user the ability to only manage users which belong to a certain group. As a result, both the role and the means by which the role was granted combine to enforce access of subjects to objects within the TOE.*

FDP_ACF.1.3 (1) The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [***none***].

FDP_ACF.1.4 (1) The TSF shall explicitly deny access of subjects to objects based on the [***following: an identity policy is defined that makes multiple admin roles mutually exclusive. If a TOE user who has one role is assigned a second, the assignment will be explicitly denied.***].

Application Note: *Identity policies are used in compliance support.*

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

6.1.3.5 FDP_ACF.1 (2) Security Attribute Based Access Control

Hierarchical to: No other components.

FDP_ACF.1.1 (2) The TSF shall enforce the [***Workflow Policy***] to objects based on the following: [***the ACL for the workflow activity***].

FDP_ACF.1.2 (2) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- ***Workflow is enabled for the object***
- ***The TOE user is authorized to process the workflow***
- ***The TOE user has not delegated the workflow to another TOE user***]

FDP_ACF.1.3 (2) The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [***none***].

FDP_ACF.1.4 (2) The TSF shall explicitly deny access of subjects to objects based on the [***none***].

6.1.3.6 FDP_ACF.1 (3) Security Attribute Based Access Control

Hierarchical to: No other components.

FDP_ACF.1.1 (3) The TSF shall enforce the [***Web Service Policy***] to objects based on the following: [***the taskContext value of the POST request submitted by the web service application***]

Application Note: *TEWS applications are executed on behalf of a user whose identity is supplied by the user after a separate identity has been validated via an appropriate third party mechanism.*

FDP_ACF.1.2 (3) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- ***the taskContext value is a valid username/password combination stored in the TOE user store***

- ***the taskContext value is part of an ACL for TEWS operations***
- ***the task to be invoked by the web service has Enable Web Services selected as one of its properties***]

Application Note: *Enable Web Services can be toggled for a task by a TOE user with Modify Admin Task privileges.*

Application Note: *View Submitted Tasks and View My Submitted Tasks cannot be invoked by TEWS.*

Application Note: *To clarify, policies which determine whether or not a TOE user belongs to a role is derived from their Operational Environment session when accessing TEWS. A TOE user being granted access to the TEWS interface via a third-party access manager will be recognized by the TSF as having the identity on the TOE as provided by the user. This user's identity is then assigned role and scope in the same manner as defined by the User Policy.*

FDP_ACF.1.3 (3) The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [***none***].

FDP_ACF.1.4 (3) The TSF shall explicitly deny access of subjects to objects based on the [***none***].

6.1.3.7 FDP_IFC.1 Information Flow Control

Hierarchical to: No other components.

FDP_IFC.1.1 The TSF shall enforce the [***workflow policy***] on [***tasks associated with workflow processes***].

Dependencies: FDP_IFF.1 Simple security attributes

6.1.3.8 FDP_IFF.1 Simple Security Attributes

Hierarchical to: No other components.

FDP_IFF.1.1 The TSF shall enforce the [***workflow policy***] based on the following types of subject and information security attributes: [

- ***the status of workflow enabled or disabled on the TOE***
- ***a task or an event within a task is associated with one or more workflow processes***
- ***at least one TOE user is associated with approval or delegation of a workflow process for the task or event***]

FDP_IFF.1.2	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [***for a given workflow process, at least one TOE user is authorized to approve a process or delegate its approval to another TOE user with the same authorization***].
FDP_IFF.1.3	The TSF shall enforce the [***no additional information flow control SFP rules***].
FDP_IFF.1.4	The TSF shall explicitly authorise an information flow based on the following rules: [***a workflow process template identifies the TOE users which are able to approve the process***].
FDP_IFF.1.5	The TSF shall explicitly deny an information flow based on the following rules: [***none***].
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation

6.1.4 Class FIA: Identification and Authentication

6.1.4.1 FIA_ATD.1 User Attribute Definition

Hierarchical to: No other components.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: **[*username, password, password verification question, password verification answer, e-mail address assigned roles, scope of role membership, enabled/disabled state*]**

Dependencies: No dependencies.

6.1.4.2 FIA_SOS.1 Verification of Secrets

Hierarchical to: No other components.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **[*defined Identity Manager password policies*].**

Dependencies: No dependencies.

6.1.4.3 FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.2 Timing of identification

Application Note: *If the TOE user account is set to the disabled state, no TSF-mediated actions will be allowed on behalf of that TOE user.*

Application Note: *Public tasks require the TOE user to identify using their username or e-mail address and require a pre-defined security verification question to be answered as an alternate form of authentication. They include the following:*

- *Self Registration: The Self Registration task allows TOE users to register at a web site without outside involvement. Users can enter profile information, set a password, and subscribe to groups.*

- *Forgotten Password: The Forgotten Password task provides a TOE user with a temporary password that he/she can use to log into Identity Manager. When the user logs in, she is immediately prompted to enter a new password.*
- *Forgotten Password Reset: The Forgotten Password Reset task enables a TOE user to reset a password after Identity Manager verifies his identity. In the default Forgotten Password Reset task, a TOE user must provide a user ID and answer three verification questions. Once Identity Manager verifies a TOE user's identity, a screen where the TOE user can enter a new password is presented.*
- *Forgotten User ID: The Forgotten User ID task allows a TOE user to retrieve a forgotten user ID. In the default task, a TOE user must provide an email address and answer one verification question to receive an email containing the TOE user's ID.*

6.1.4.4 FIA_UID.2 User Identification Before Any Action

Hierarchical to: FIA_UID.1

FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: No dependencies.

6.1.5 Class FMT: Security Management

6.1.5.1 FMT_MSA.1 Management of Security Attributes

Hierarchical to:	No other components.
FMT_MSA.1.1	The TSF shall enforce the [***User Policy***] to restrict the ability to [***query, modify, or delete***] the security attributes [***tasks, roles, scope, workflow ACL, web service ACL, workflow enabled state, web service enabled state***] to [***TOE users with the role and the appropriate scope associated with the tasks: Create/Modify/Delete Admin Roles, Modify Users, Modify Admin Role Members/Administrators, Create/Modify Admin Tasks***].

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

6.1.5.2 FMT_MSA.3 Static Attribute Initialization

Hierarchical to: No other components.

FMT_MSA.3.1 The TSF shall enforce the [***User Policy***] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [***TOE user with any of the available tasks specified in the table below***] to specify alternative initial values to override the default values when an object or information is created.

Role	**Available Tasks**
Admin Role Manager	Create Admin Role Create Admin Task
Certify Manager	Certify User
Group Manager	Create Group
Provisioning Role Manager	Create Provisioning Role Create Owner Policies for Provisioning Roles
User Manager	Create User
(no role)	Create Endpoint Create Account Template Create Identity Policy Set Create Password Policy Select Box Data Create Logical Attribute Handler Create Explore and Correlate Definition

Table 6-5: Tasks Capable of Data Creation

Application Note: *A TOE user that is assigned one or more admin roles which appear in this table may only override default values for the tasks which are associated with their role(s).*

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

6.1.5.3 FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

FMT_MTD.1.1(1) The TSF shall restrict the ability to [***query***] the [***data in the Identity Manager Database and User Store, configuration of the remote endpoints***] to [***TOE users with any of the following available tasks:***

Admin Role	**Available Tasks**
Admin Role Manager	View Admin Role View Admin Role Members/Administrators View Admin Task
Delegation Manager	A Delegation Manager can act on behalf of another user (the delegator) and delegate work item approval to tasks a third user (the delegate). This role has scope over all users.
Group Manager	View Group
Password Manager	Reset User Password View User
Provisioning Role Manager	View Provisioning Role View Provisioning Role Members/Administrators
Security Manager	View User
Self Manager	View My Roles View My Submitted Tasks View My Work List
User Manager	View User
(no role)	View Endpoint View Account Template View Identity Policy Set View Password Policy View Submitted Tasks View Logical Attribute Handler View Explore and Correlate Definition

Table 6-6: Tasks Authorized to Query TSF Data

]

Application Note: *A TOE user that is assigned one or more admin roles which appear in this table may only query TOE data as part of the tasks which are associated with their role(s).*

Application Note: *If a TOE user views the configuration of the Provisioning Server or Application Server, this is accomplished by querying the Identity Manager database and user stores.*

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1(2) The TSF shall restrict the ability to [***modify***] the [***data in the Identity Manager Database and User Store, configuration of the remote endpoints***] to [***TOE users with any of the following available tasks:***

Admin Role	**Available Tasks**
Admin Role Manager	Create Admin Role Create Admin Task Modify Admin Role Modify Admin Role Members/Administrators Modify Admin Task Reset Admin Role Owners
Certify Manager	Certify User
Certification Process Manager	Begin Certification Process End Certification Process Send Certification Reminder Notification Send Final Certification Reminder Notification
Delegation Manager	Delegate Work Items
Group Manager	Create Group Modify Group
Password Manager	Reset User Password
Provisioning Role Manager	Create Provisioning Role Modify Provisioning Role Modify Provisioning Role Members/Administrators Reset Provisioning Role Owners
Security Manager	Enable/Disable User Reset User Password
Self Delegator	Out of Office
Self Manager	Change My Account Change My Password Modify My Groups

	Modify My Profile
User Manager	Create User Modify User Modify Group Members
(no role)	Delete Endpoint Modify Endpoint Delete Account Template Modify Account Template Delete Identity Policy Set Modify Identity Policy Set Synchronize Users Delete Password Policy Modify Password Policy Connection Management Modify Logical Attribute Handler Modify Explore and Correlate Definition

Table 6-7: Tasks Authorized to Modify TSF Data

]

Application Note: *A TOE user that is assigned one or more admin roles which appear in this table may only modify TOE data as part of the tasks which are associated with their role(s).*

Application Note: *If a TOE user changes the configuration of the Provisioning Server or Application Server, this is accomplished by inserting or updating information in the Identity Manager database.*

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1(3) The TSF shall restrict the ability to [***delete***] the [***data in the Identity Manager Database and User Store, configuration of the remote endpoints***] to [***TOE users with any of the following assigned roles:***

Role	**Available Tasks**
Admin Role Manager	Delete Admin Role Delete Admin Task
Group Manager	Delete Group
Provisioning Role Manager	Delete Provisioning Role
User Manager	Delete User
(no role)	Delete Explore and Correlate Definition

Table 6-8: Roles Authorized to Delete TSF Data

]

Application Note: *A TOE user that is assigned one or more roles which appear in this table may only delete TOE data as part of the tasks which are associated with their role(s).*

Application Note: *If a TOE user removes a configuration of the Provisioning Server or Application Server, this is accomplished deleting information in the Identity Manager database and user stores.*

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

6.1.5.4 FMT_REV.1 Revocation

Hierarchical to: No other components.

FMT_REV.1.1 The TSF shall restrict the ability to revoke ***[access to tasks, role assignments, scope assignments]*** associated with the [***TOE users***] under the control of the TSF to [***TOE users with access to the "Modify User" task and sufficient scope to manage the target TOE user(s)***].

FMT_REV.1.2 The TSF shall enforce the rules [***upon first new page load following completion of the change of role, change of TOE user's role scope, or removal of role from the TOE user***].

Dependencies: FMT_SMR.1 Security roles

Application Note: *Completion of the change of role is not necessarily immediate. For example, the process could be subject to workflow, in which case no action would be taken until the final step was approved.*

6.1.5.5 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [***as described in Table 6-8, other admin tasks as created by TOE users with Create Admin Task privileges***].

Dependencies: No dependencies.

6.1.5.6 FMT_SMR.2 Restrictions on Security Roles

Hierarchical to: FMT_SMR.1 Security Roles

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.2.1 The TSF shall maintain the roles [***Admin Role Manager, Certify Manager, Certification Process Manager, Delegation Manager, Group Manager, Password Manager, Provisioning Role Manager, Provisioning Synchronization Manager, Security Manager, Self Delegator, Self Manager, User Manager, custom roles as defined by TOE user with Create Admin Role task privilege***].

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions **[*TOE users with a certain role can only perform tasks authorized to that particular role*]** are satisfied.

6.2 Security Functional Requirements for the Operational Environment

There are no security functional requirements for the Operational Environment in this ST beyond the extended requirements.

6.3 Operations Defined

The notation, formatting, and conventions used in this security target (ST) are consistent with version 3.1 of the Common Criteria for Information Technology Security Evaluation. All of the components in this ST are taken directly from Part 2 of the CC. Font style and clarifying information conventions were developed to aid the reader.

The CC permits four functional component operations:, assignment, iteration, selection, and refinement to be performed on functional requirements. These operations are defined in Common Criteria, Part 1 as:

6.3.1 Assignments Made

An assignment allows the specification of parameters and is specified by the ST author in [***italicized bold text***].

6.3.2 Iterations Made

An iteration allows a component to be used more than once with varying operations and are identified with the iteration number within parentheses after the short family name.

6.3.3 Selections Made

A selection allows the specification of one or more items from a list and is specified by the ST author in [***italicized bold text***].

6.3.4 Refinements Made

A refinement allows the addition of details and is identified with "Refinement:" right after the short name. Additions to the CC text are specified in ***italicized bold and underlined text***.

7 Security Assurance Requirements

This section identifies the Security Assurance Requirement components met by the TOE. These assurance components meet the requirements for EAL3 augmented with ALC_FLR.1 and ASE_TSS.2.

7.1 Security Architecture

7.1.1 Security Architecture Description (ADV_ARC.1)

ADV_ARC.1.1D: The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV_ARC.1.2D: The developer shall design and implement the TSF so that it is able to protect itself from tampering by un-trusted active entities.

ADV_ARC.1.3D: The developer shall provide a security architecture description of the TSF.

ADV_ARC.1.1C: The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.

ADV_ARC.1.2C: The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV_ARC.1.3C: The security architecture description shall describe how the TSF initialization process is secure.

ADV_ARC.1.4C: The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV_ARC.1.5C: The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.

ADV_ARC.1.1E: The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.2 Functional Specification with Complete Summary (ADV_FSP.3)

ADV_FSP.3.1D The developer shall provide a functional specification.

ADV_FSP.3.2D The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.3.1C The functional specification shall completely represent the TSF.

ADV_FSP.3.2C The functional specification shall describe the purpose and method of use for all TSFI.

ADV_FSP.3.3C The functional specification shall identify and describe all parameters associated with each TSFI.

ADV_FSP.3.4C For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.

ADV_FSP.3.5C For each SFR-enforcing TSFI, the functional specification shall describe direct error messages resulting from security enforcing effects and exceptions associated with invocation of the TSFI.

ADV_FSP.3.6C The functional specification shall summarize the SFR-supporting and SFR-non-interfering actions associated with each TSFI.

ADV_FSP.3.7C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.3.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

7.1.3 Architectural Design (ADV_TDS.2)

ADV_TDS.2.1D The developer shall provide the design of the TOE.

ADV_TDS.2.2D The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.

ADV_TDS.2.1C The design shall describe the structure of the TOE in terms of subsystems.

ADV_TDS.2.2C The design shall identify all subsystems of the TSF.

ADV_TDS.2.3C The design shall describe the behavior of each SFR non interfering subsystem of the TSF in detail sufficient to determine that it is SFR non-interfering.

ADV_TDS.2.4C The design shall describe the SFR-enforcing behavior of the SFR enforcing subsystems.

ADV_TDS.2.5C The design shall summarize the SFR-supporting and SFR-non interfering behavior of the SFR-enforcing subsystems.

ADV_TDS.2.6C The design shall summarize the behavior of the SFR-supporting subsystems.

ADV_TDS.2.7C The design shall provide a description of the interactions among all subsystems of the TSF.

ADV_TDS.2.8C The mapping shall demonstrate that all behavior described in the TOE design is mapped to the TSFIs that invoke it.

ADV_TDS.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_TDS.2.2E The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

7.2 Guidance Documents

7.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1D The developer shall provide operational user guidance.

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

7.3 Lifecycle Support

7.3.1 Authorization Controls (ALC_CMC.3)

ALC_CMC.3.1D The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.3.2D The developer shall provide the CM documentation.

ALC_CMC.3.3D The developer shall use a CM system.

ALC_CMC.3.1C The TOE shall be labeled with its unique reference.

ALC_CMC.3.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.3.3C The CM system shall uniquely identify all configuration items.

ALC_CMC.3.4C The CM system shall provide measures such that only authorized changes are made to the configuration items.

ALC_CMC.3.5C The CM documentation shall include a CM plan.

ALC_CMC.3.6C The CM plan shall describe how the CM system is used for the development of the TOE.

ALC_CMC.3.7C The evidence shall demonstrate that all configuration items are being maintained under the CM system.

ALC_CMC.3.8C The evidence shall demonstrate that the CM system is being operated in accordance with the CM plan.

ALC_CMC.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.2 CM Scope (ALC_CMS.3)

ALC_CMS.3.1D The developer shall provide a configuration list for the TOE.

ALC_CMS.3.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; the parts that comprise the TOE; and the implementation representation.

ALC_CMS.3.2C The configuration list shall uniquely identify the configuration items.

ALC_CMS.3.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item. Evaluator action elements:

ALC_CMS.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.3 Delivery Procedures (ALC_DEL.1)

ALC_DEL.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the consumer.

ALC_DEL.1.2D The developer shall use the delivery procedures.

ALC_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

ALC_DEL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.4 Identification of Security Measures (ALC_DVS.1)

ALC_DVS.1.1D The developer shall produce development security documentation.

ALC_DVS.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

7.3.5 Life-cycle Definition (ALC_LCD.1)

ALC_LCD.1.1D The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.

ALC_LCD.1.2D The developer shall provide life-cycle definition documentation.

ALC_LCD.1.1C The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_LCD.1.2C The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE. Evaluator action elements:

ALC_LCD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.6 Basic Flaw Remediation (ALC_FLR.1)

ALC_FLR.1.1D The developer shall document flaw remediation procedures addressed to TOE developers. Content and presentation elements:

ALC_FLR.1.1C The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.1.2C The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.1.3C The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.1.4C The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users. Evaluator action elements:

ALC_FLR.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.4 Security Target Evaluation

7.4.1 Conformance Claims (ASE_CCL.1)

ASE_CCL.1.1D The developer shall provide a conformance claim.

ASE_CCL.1.2D The developer shall provide a conformance claim rationale.

ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

7.4.2 Extended Components Definition (ASE_ECD.1)

ASE_ECD.1.1D The developer shall provide a statement of security requirements.

ASE_ECD.1.2D The developer shall provide an extended components definition.

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

ASE_ECD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components.

7.4.3 ST Introduction (ASE_INT.1)

ASE_INT.1.1D The developer shall provide an ST introduction.

ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C The ST reference shall uniquely identify the ST.

ASE_INT.1.3C The TOE reference shall identify the TOE.

ASE_INT.1.4C The TOE overview shall summarize the usage and major security features of the TOE.

ASE_INT.1.5C The TOE overview shall identify the TOE type.

ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE.

ASE_INT.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

7.4.4 Security Objectives (ASE_OBJ.2)

ASE_OBJ.2.1D The developer shall provide a statement of security objectives.

ASE_OBJ.2.2D The developer shall provide a security objectives rationale.

ASE_OBJ.2.1C The statement of security objectives shall describe the security objectives for the TOE and the security objectives for the operational environment.

ASE_OBJ.2.2C The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective.

ASE_OBJ.2.3C The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

ASE_OBJ.2.4C The security objectives rationale shall demonstrate that the security objectives counter all threats.

ASE_OBJ.2.5C The security objectives rationale shall demonstrate that the security objectives enforce all OSPs.

ASE_OBJ.2.6C The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.

ASE_OBJ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.4.5 Security Requirements (ASE_REQ.2)

ASE_REQ.2.1D The developer shall provide a statement of security requirements.

ASE_REQ.2.2D The developer shall provide a security requirements rationale.

ASE_REQ.2.1C The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.2.2C All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.2.3C The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.2.4C All operations shall be performed correctly.

ASE_REQ.2.5C Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.2.6C The security requirements rationale shall trace each SFR back to the security objectives for the TOE.

ASE_REQ.2.7C The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE.

ASE_REQ.2.8C The security requirements rationale shall explain why the SARs were chosen.

ASE_REQ.2.9C The statement of security requirements shall be internally consistent.

ASE_REQ.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.4.6 Security Problem Definition (ASE_SPD.1)

ASE_SPD.1.1D The developer shall provide a security problem definition.

ASE_SPD.1.1C The security problem definition shall describe the threats.

ASE_SPD.1.2C All threats shall be described in terms of a threat agent, an asset, and an adverse action.

ASE_SPD.1.3C The security problem definition shall describe the OSPs.

ASE_SPD.1.4C The security problem definition shall describe the assumptions about the operational environment of the TOE.

ASE_SPD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.4.7 TOE Summary Specification (ASE_TSS.2)

ASE_TSS.2.1D The developer shall provide a TOE summary specification.

ASE_TSS.2.1C The TOE summary specification shall describe how the TOE meets each SFR.

ASE_TSS.2.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

ASE_TSS.2.2C The TOE summary specification shall describe how the TOE protects itself against interference and logical tampering.

ASE_TSS.2.2E The evaluator *shall confirm* that the TOE summary specification is consistent with the TOE overview and the TOE description.

ASE_TSS.2.3C The TOE summary specification shall describe how the TOE protects itself against bypass.

7.5 Tests

7.5.1 Analysis of Coverage (ATE_COV.2)

ATE_COV.2.1D The developer shall provide an analysis of the test coverage.

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that all TSFIs in the functional specification have been tested.

ATE_COV.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.5.2 Basic Design (ATE_DPT.1)

ATE_DPT.1.1D The developer shall provide the analysis of the depth of testing.

ATE_DPT.1.1C The analysis of the depth of testing shall demonstrate the correspondence between the tests in the test documentation and the TSF subsystems in the TOE design.

ATE_DPT.1.2C The analysis of the depth of testing shall demonstrate that all TSF subsystems in the TOE design have been tested.

ATE_DPT.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.5.3 Functional Tests (ATE_FUN.1)

ATE_FUN.1.1D The developer shall test the TSF and document the results.

ATE_FUN.1.2D The developer shall provide test documentation

ATE_FUN.1.1C The test documentation shall consist of test plans, expected test results and actual test results.

ATE_FUN.1.2C The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.3C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.4C The actual test results shall be consistent with the expected test results.

ATE_FUN.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.5.4 Independent Testing (ATE_IND.2)

ATE_IND.2.1D The developer shall provide the TOE for testing.

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_IND.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

ATE_IND.2.3E The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

7.6 Vulnerability Assessment

7.6.1 Vulnerability Analysis (AVA_VAN.2)

AVA_VAN.2.1D The developer shall provide the TOE for testing.

AVA_VAN.2.1C The TOE shall be suitable for testing.

AVA_VAN.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.2.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.2.3E The evaluator shall perform an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design and security architecture description to identify potential vulnerabilities in the TOE.

AVA_VAN.2.4E The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

8 TOE Summary Specification

The following sections identify the security functions of the TOE.

8.1.1 Identification and Authentication

Identity Manager includes native authentication for Identity Manager environments by default. Identity Manager users enter a valid username and password to log into an Identity Manager environment. Identity Manager authenticates the name and password against the user store that Identity Manager manages. This is accomplished using the out-of-the-box Java Authentication Module within the Application Server.

Identity Manager enables TOE users to create basic password policies that manage TOE user passwords by enforcing rules and restrictions governing password expiration, composition, and usage.

Identity Manager includes one console that should be protected by the TOE as well as the Task Execution Web Service (TEWS) interface which is protected by a third party application:

- *User Console:* Enables TOE users the ability to perform management tasks remotely in an Identity Manager environment. The scope of the allowed behaviors depends on the tasks assigned to the TOE user and the scope assigned to them.

- *Task Execution Web Service:* Enables TOE users to develop applications using the Identity Manager web service API. This gives TOE users the ability to automate certain tasks or execute large numbers of tasks in batches rather than execute them all manually.

  NOTE: *In the evaluated configuration, the TEWS interface is expected to only be used by a user that has been assigned all privileges of the TOE.*

Identity Manager also has a Management Console, which enables an individual to perform the initial configuration activities of the TOE prior to its operation. Since this is part of the installation process and not its operation, the Management Console is not part of the evaluated configuration.

Identity Manager includes native authentication, which protects the User Console by default.

8.1.1.1 User Console Authentication

The User Console is the user interface that TOE users use to manage objects such as users, groups, and endpoints in an Identity Manager environment, with a set of associated roles and tasks. After a TOE user authenticates to the User Console, that TOE user can only see tasks that he can perform in that environment. All TOE users have some authorized administrative functions, but this can be restricted to the point where the TOE

user can only perform self-management tasks on the User Console, such as changing their own password.

By default, Identity Manager protects access to the Management Console with native authentication. TOE users enter a valid username and password to log into an Identity Manager environment. Identity Manager authenticates the name and password against the user store that Identity Manager manages.

8.1.1.2 Provisioning Server

The provisioning server contains the core logic of the provisioning system. It communicates to target systems via connectors to add, delete and modify endpoint user accounts.

The Provisioning Server is the server that manages additional accounts that are assigned to an endpoint user. When a provisioning role is assigned to an endpoint user, the Provisioning Server creates accounts on endpoints that meet the requirements of the role. For example, if a user is assigned a provisioning role that includes an LDAP account template, the Provisioning Server assigns an LDAP account to the user, who is now designated as an endpoint user.

8.1.1.3 Task Execution Web Service

The Identity Manager Task Execution Web Service (TEWS) is a web service interface that allows third-party client applications to submit remote tasks to Identity Manager for execution. This interface implements the open standards of WSDL and SOAP to provide remote access to Identity Manager. Most Identity Manager tasks are supported.

A client application submits a remote task as an HTTP 1.x POST request to a web service URL in the Identity Manager environment. The body of the POST request is a SOAP document that conforms to the interface described in a task-specific WSDL document generated by Identity Manager. The WSDL document describes the metadata that the client application needs to prepare and submit a task request.

In the evaluated configuration, each TEWS request has to include a TaskContext value. This context includes the username of the TOE user who is nominally running the task. If the taskContext value does not identify the username of a user who is authorized to run TEWS the request will be rejected. In addition, the request will be also be rejected if the task to be performed does not have the "Enable Web Services" property enabled. These criteria define the Web Service Policy for the TOE that protects access to TEWS. No other method can be used to supersede this mechanism to explicitly allow or deny access to TEWS operations.

TEWS is configured in the Management Console prior to operation of the TOE.

The TEWS servlet will extract this information from the request and use it to identify the TOE user with the User Store before proceeding. If the operation succeeds, then the task is considered to be being run by the TOE user whose identity was supplied.

The credentials are sent over a secure channel, using TLS 1.0 with 128-bit AES as the symmetric key algorithm.

In the evaluated configuration, the TEWS interface requires a third party application to authenticate users prior to granting access to the interface. The user will then provide their identity to the TOE for identification and to determine access control restrictions. Thus, TEWS interface will only identify a user that has already been granted access to the interface by the third party application. The identity provided to the TOE does not have to match the one provided to the third party application, nor will the identity provided to the TOE be authenticated. Therefore, the TEWS interface is expected to only be used by a user that has been assigned all privileges of the TOE in the evaluated configuration.

NOTE: *Although it was not validated through the evaluation, the vendor has asserted that the TOE can have the TEWS interface protected by CA SiteMinder. This would allow for TOE users to have their TOE identity be authenticated by SiteMinder, and then have all actions on the TOE be associated with their validated TOE identity.*

8.1.1.4 Public Tasks

Public tasks are activities which can be performed by any TOE user with any level of privilege. Unlike the other features of the TOE, public tasks can be performed without password input. They are listed below:

- **Self Registration:** The Self Registration task allows TOE users to register at a corporate Web site without outside involvement. TOE users can enter profile information, set a password, and subscribe to groups.
- **Forgotten Password:** The Forgotten Password task provides a TOE user with a temporary password that he/she can use to log into Identity Manager. When the TOE user logs in, she is immediately prompted to enter a new password.
- **Forgotten Password Reset:** The Forgotten Password Reset task enables a TOE user to reset a password after Identity Manager verifies his identity.

  In the default Forgotten Password Reset task, a TOE user must provide a user ID and answer three verification questions. Once Identity Manager verifies a TOE user's identity, a screen where the TOE user can enter a new password is presented.
- **Forgotten User ID:** The Forgotten User ID task allows a TOE user to retrieve a forgotten user ID. In the default task, a TOE user must provide an email address and answer one verification question to receive an email containing the TOE user's ID.

8.1.1.5 Password Policies

A password policy is a set of rules and restrictions that determines how passwords are created and when they expire.

In a password policy, the following settings can be configured:

- Password composition--Specify the content requirements for new passwords. For example, a TOE user can configure settings that require TOE users to create passwords which are at least eight characters long and contain a number and a letter.
- Regular expressions--Provide an expression that determines the format of a valid password. A TOE user can specify whether passwords must match or must not match that format.
- Advanced password options--Specify actions that Identity Manager should take, such as making passwords lower case, before processing a password. The priority of a password policy can be specified if multiple password policies apply.

Note: *A TOE user configures a password policy in an Identity Manager environment; however, the policy applies to the user store associated with the environment. If a user directory is associated with multiple environments, a password policy defined in one environment may apply in other environments, as well.*

The TOE can enforce password policies (with complexity requirements) for the TOE users to meet the requirements of an organization password policy. The TOE by default does not enforce password policies or complexity requirements for TOE users' passwords until after a policy is created. TOE users are required to create and use strong/complex passwords as instructed by their organization's user guidance in accordance with the documented password policy. The organization's password policy can be configured by a TOE user with a role which has been assigned the create/modify/delete password policy task, and will be enforced by CA Identity Manager.

For the purposes of the evaluated configuration, a sufficiently secure password policy has been defined below:

- At least eight characters
- At least one number
- At least one capital and one lowercase letter

8.1.2 Security Management

8.1.2.1 Identity Manager Database

The evaluated configuration for Identity Manager will include the following data stores to support Identity Manager functionality:

- **Policy Database**
  Contains Identity Manager configuration information. TOE users performing management tasks such as Modify Admin Role modify the data in this database. TOE users performing display tasks such as View Admin Role query the data in this database. The TOE itself also queries the database to determine what operations are allowed for a TOE user.

- **Task Persistence Database**
  Maintains information about Identity Manager activities and their associated events over time. This allows the system to accurately track Identity Manager activities, even if the Identity Manager Server is restarted for some reason. The data in this database is added automatically as tasks are performed and queried through View Submitted Tasks.

- **Workflow Database**
  Stores workflow process definitions, jobs, scripts, and other data required by the Workflow Engine. As admin tasks are configured to utilize workflow, data is written to this database, which is then queried as admin tasks are executed or viewed by a TOE user.

- **Audit Database**
  Provides a historical record of operations that occur in an Identity Manager environment. This database is written to automatically as tasks are performed and is queried via a third party mechanism. Identity Manager has no capability to read the data in the Audit Database.

  Note: *The amount and type of information that Identity Manager stores in the audit database can be configured to accommodate the environment's storage constraints.*

When the Identity Manager Installer is run, Identity Manager configures a connection to a single database, called the Identity Manager Database, which contains separate data stores for each of these database types.

Note: *A data store for task persistence, workflow, or auditing can alternatively be created in a separate database and configure Identity Manager to connect to it. See the Installation Guide for more information.*

8.1.2.2 User Attributes

The list of attributes of a TOE user is configured in the "directory.xml" file used to create the directory object in IM's management console. This file contains the following security-relevant user attributes:

- Username
- Password

- Email address
- Password recovery question and answer
- Disabled flag

When the TOE installed in Windows, for example, this file will normally be found in C:\Program Files\CA\IAM Suite\Identity Manager\tools\directoryTemplates\.

The data attributes are associated with TOE users include fields such as a disabled flag and personal information. These values can be modified by other TOE users who are authorized to perform the Modify User task, or by the TOE users themselves (for their own individual data).

The user attributes that are used for identification and authentication on the TOE are username, password, and email address. In order to perform tasks which require authentication to the TOE, the TOE user must supply their username and password. Public tasks do not require authentication via password, but the TOE user must identify using their username or e-mail address and have their identity validated by a pre-defined security verification question to be answered as an alternate form of authentication (for example, if they forgot their username or password, a TOE user would need to identify with an email address so that the TOE knows how to communicate further with the TOE user). To recover their password, a TOE user must supply the correct answer to their recovery question.

If a TOE user does not require access to the TOE (for example, if they have left the organization or are away for an extended period of time), the disabled flag can be set to deny all authentication requests from that TOE user.

8.1.2.3 Password Management

Identity Manager includes several features for managing TOE user passwords:

- Password Policies--These policies manage TOE user passwords by enforcing rules and restrictions governing password expiration, composition, and usage.
- Password Managers--TOE users who have the Password Manager role can reset a password at the request of another TOE user.
- Self-service password management--Identity Manager includes several self-service tasks that allow TOE users to manage their own passwords. These tasks include:
    - Self Registration--TOE users specify a password when they register at a corporate Web site.
    - Change My Password--TOE users can modify their passwords without help from TOE users with the Password Manager role
    - Forgotten Password--TOE users can reset or retrieve a forgotten password after Identity Manager verifies their identity.

- Forgotten User ID--TOE users can retrieve a forgotten user ID after Identity Manager verifies their identity.

8.1.2.4 Groups

A TOE user with the "Create Group" task can create several types of groups, or a combination of these types:

- Static group--A list of TOE users who are added interactively
- Dynamic group--TOE users belong to the group if they meet an LDAP query

  Note: *The Dynamic Group Query field is not included in the Create Group task or other group tasks even if this field exists in the directory.xml for a group. A TOE user would include Dynamic Group Query field in the task by editing the associated profile screen.*
- Nested group--A group containing other groups

Note: *To view the static, dynamic, and nested groups to which a TOE user belongs, use the Groups tab for the User object. This tab appears in the View and Modify User tasks by default.*

8.1.2.4.1 Static Group

A collection of TOE users can be associated in a static group. The static group is managed by adding or removing individual TOE users from the group's membership list. Only TOE users authorized to create or manage groups can perform this operation. Prerequisite assigned tasks include Modify Group Members/Administrators, Modify My Groups, Create Group. To see the list of members for a group, use the Membership tab, which is included with the View and Modify Group tasks by default.

Note: *The Membership tab displays only the members who are explicitly added to the group. It does not display members who are added dynamically.*

8.1.2.4.2 Dynamic Group

A TOE user can create a dynamic group by defining an LDAP filter query using the Identity Manager User Console to dynamically determine group membership at runtime without having to search and add TOE users individually. This ability requires authorization to perform the Create Group task.

For example, if a TOE user wanted to generate a group that lists all U.S. employees of a fictional organization called NeteAuto, they could define an LDAP search filter similar to the following in the Dynamic Group Query field of the Identity Manager User Console:

ldap:///cn=Employees,o=NeteAuto,c=US??sub

They could also modify this query to locate employees outside the United States.

Note: *The Dynamic Group Query field is included in the task by editing the associated profile screen. It is not included by default in the Create Group task.*

8.1.2.4.3 Nested Group

Because the user store is an LDAP directory (CA Directory), a group can be added as a member of another group. The group is called a nested group.

The group containing the nested group is called a parent group. Members of the nested group become members of the parent group. However, members of the parent group do not become members of a nested group.

Nested groups are similar to email distribution lists where one list can be a member of another. With nested groups, groups and TOE users can be added as members in the group. By nesting a group in another group's membership list, all of the nested group's members would be included.

For example, if separate groups are created for the manufacturing, design, shipping, and accounting divisions of a company, a TOE user can construct a parent group for the entire company by nesting all the separate division groups as members of the company parent group. As a result, any changes they made to the manufacturing, design, shipping, and accounting nested groups would be automatically reflected in the nested group for the entire company. A group that is nested within another group can be dynamic and/or contain other nested groups.

Be aware of the following before creating a nested group:

- Only a TOE user with the Modify Group Members task can add or change nested groups from the group's static member list in the Identity Manager User Console.
- Only TOE users with the appropriate privileges can modify, add, or remove members from a group.

  For example, if parent Group A is created by nested groups B and C, the Group A administrator can only modify the members of Group A and not B and C. Groups B and C can only be modified by their appropriate administrators.

8.1.2.4.4 Group Administrators

On the Administrators tab of the Create or Modify Group tasks, a TOE user can specify TOE users and groups as administrators of a group. This ability requires authorization to

perform the Create Group or Modify Group Members/Administrators task. When a TOE user is assigned as a group administrator, make sure that the administrator has a role with appropriate scope for managing the group. For example:

1. Use Modify Group to assign a TOE user as an administrator of a group.
2. Assign that TOE user an admin role with group management tasks, such as Modify Group Members, or user management tasks with a Groups tab.
3. Check that the role has appropriate scope over the group.
    a. Use View Admin Role on the role that was assigned with group management tasks.
    b. On the Members tab, verify a policy exists with the following:

       A member rule that the group administrator meets

       A scope rule that includes the group

       A scope rule that includes some TOE users to be added to the group

When a TOE user assigns a group, only administrators of that group will be administrators of the group they are creating or modifying. Members of the TOE user group they specify will not have privileges to manage the group.

8.1.2.5 Compliance Support

Compliance is a corporate governance that includes a wide range of procedures that ensure a company and its employees comply with business policies. These compliance procedures often involve documenting, automating, and auditing the allocation of entitlements to applications and systems.

In the evaluated configuration, Identity Manager includes the following feature, which supports compliance management:

- Identity policies – A TOE user can create a compliance policy, a type of identity policy, which prohibits TOE users from having certain privileges if they have other privileges. For example, they can prohibit TOE users who can approve checks from issuing checks.

  Compliance policies enforce a segregation of duties in the evaluated configuration.

8.1.2.6 Default System Tasks

Identity Manager includes the following tasks that help TOE users to manage an Identity Manager environment:

- **View Submitted Tasks:** Allows TOE users to view the status of tasks in the environment.

- **Bulk Loader:** Uploads feeder files that are used to manipulate large numbers of managed objects simultaneously.

- **Select Box Data:** Allows TOE users to upload files that are used to populate options in fields, such as select boxes, in admin tasks.

- **Logical Attribute Handler:** Allows TOE users to manage logical attributes, which are used to display user store attributes (called physical attributes) in a user-friendly format on task screens.

- **Connection Management:** Configures the database server connection details in Identity Manager.

TOE users may want to track the status of Identity Manager tasks once they are submitted for processing. Identity Manager provides the following methods for viewing task status:

- View Submitted Tasks tab: This tab allows a TOE user to search for and display Identity Manager tasks that have been submitted for processing. TOE users can view task details at a high level or view additional levels of detail.

  The View Submitted Tasks tab is included in two default tasks:

  - View My Submitted Tasks: Allows TOE users with the "View My Submitted Tasks" task to search for and display information about tasks that they submitted for processing.

  - View Submitted Tasks: Allows TOE users with the "View Submitted Tasks" task to search for and display information about tasks that other TOE users have submitted for processing.

- User History tab: This tab, which is associated with TOE user tasks, such as View or Modify User, lets TOE users view the following information for a selected TOE user:

  - Tasks performed on the TOE user
  - Tasks performed by the TOE user
  - Workflow approvals by the TOE user

A task is an administrative function that a TOE user can perform in Identity Manager. Tasks include events, actions that Identity Manager performs to complete the task. A task

may include multiple events. For example, the Create User task may include events that create the TOE user's profile, adds the TOE user to a group, and assigns roles.

Identity Manager tasks and events can be associated with a workflow process, which determines how Identity Manager performs the required actions, and other custom business logic. Tasks may also be associated with other tasks, called nested tasks. In this case, Identity Manager processes the nested tasks with the original task.

The status of a task depends on the status of its associated events, workflow processes, nested tasks, and custom business logic.

8.1.2.7 Default Tasks

8.1.2.7.1 Default Self Service Tasks

Identity Manager includes the following self-service tasks that can be performed by authenticated TOE users:

- Change My Password--Allows TOE users to reset their password
- Modify My Profile—Allows TOE users to modify profile information, such as address and phone number
- Modify My Groups--Enables TOE users to subscribe to groups
- View My Roles--Displays a TOE user's roles
- View My Submitted Tasks--Displays Identity Manager tasks that the TOE user initiated

For tasks that can be performed without authentication, refer to **Public Tasks**.

8.1.2.7.2 Default Admin Tasks

When a TOE user authorized to perform the Create Admin Role task creates an admin role, they can include any of the default admin tasks, which appear in the user interface under these categories:

- **My Account:** The My Account tab provides these tasks for viewing or modifying the profile, roles, groups, and password of the current TOE user.
    - Change My Account – Lets a TOE user make a request to change the details of their own account.
    - Change My Password – Lets a TOE user change their own password.
    - Modify My Groups – Lets a TOE user join or leave any self-subscribing group (a group that doesn't require an explicit authorization or other membership prerequisite)

- Modify My Profile – Lets a TOE user change their own personal information.
- Out of Office – Lets a TOE user specify another TOE user to approve items on their work list in preparation for a planned absence from the organization.
- View My Roles – Displays a list of roles for which the TOE user is a member or an administrator.
- View My Submitted Tasks – Lets a TOE user view all tasks they have submitted (or a subset based on filters such as time period) or details about the events generated by those tasks.
- View My Work List – Displays a list of tasks assigned to the TOE user for workflow approval. By default, the following types of approval tasks are made available and can be assigned to the TOE user based on the definition of a workflow process:
  - Approve Modify Admin Role Membership
  - Approve Create Group
  - Approve Delete Group
  - Approve Modify Group Membership
  - Approve Modify Provisioning Role Membership
  - Approve Self Registration
  - Approve Create User
  - Approve Delete User
  - Approve Modify User

- **Admin Roles:** A TOE user can use the Admin Role Management category to create or manage admin roles and admin tasks. The category contains these tasks:
  - Create Admin Role – Used to create a role that contains admin tasks.
  - Create Admin Task – Used to create a custom task that can then be included in an admin role.
  - Delete Admin Role – Used to remove an admin role that exists on the TOE.
  - Delete Admin Task – Used to remove an admin task that exists on the TOE.
  - Modify Admin Role – Used to change the properties of an admin role such as its enabled/disabled state and the tasks which comprise the role.
  - Modify Admin Role Members/Administrators – Used to manually assign TOE users to be a member or administrator for a role.
  - Modify Admin Task – Used to modify an admin task that exists on the TOE.
  - Reset Admin Role Owners – Used to redefine the owner rules which determine who can modify the role.

- View Admin Role – Used to display the properties of an admin role.
- View Admin Role Members/Administrators – Used to display the TOE users who are members or administrators of a particular role.
- View Admin Task – Used to display the properties of an admin task.

- **Certification:** The Certification category manages the TOE user certification process. This category includes the following tasks:
  - Begin Certification Process – Allows the TOE user to select other TOE users that require certification. Once you select the TOE users, Identity Manager sets the TOE user's certification to REQUIRES CERTIFICATION, and sends an email to the appropriate certifier, notifying them that they have pending certifications.
  - End Certification Process – Allows the TOE user to disable TOE users which have not been certified.
  - Send Certification Reminder Notification – Allows the TOE user to send an email to certifiers to inform them they have pending certifications.
  - Send Final Certification Reminder Notification – This task is similar to the Send Certification Reminder Notification task except that the distributed email also warns the certifiers that no further notification will be given.
- **Endpoints:** The Endpoints category manages endpoints and account templates to apply management settings to remote endpoints via the Connector Server. This category includes the following tasks:
  - Create Endpoint – Allows the TOE user to create a new endpoint to which provisioning can be applied.
  - Delete Endpoint – Allows the TOE user to remove an endpoint from the set of available endpoints.
  - Modify Endpoint – Allows the TOE user to modify the information about the endpoint (such as the available applications on the endpoint and its network information).
  - View Endpoint – Displays the properties of a selected endpoint.
  - Create Account Template – Allows the TOE user to create an account template to apply application configuration rules to endpoints as part of the provisioning process.
  - Delete Account Template – Allows the TOE user to remove an account template.
  - Modify Account Template – Allows the TOE user to modify the properties of an account template.
  - View Account Template – Displays the properties of an account template.
- **Endpoint Accounts:** The Endpoint Accounts category allows a TOE user to create, modify, view, or delete endpoint user accounts for each endpoint type supported by the TOE. This can be used to directly manage endpoint user accounts when account templates are not sufficient.

- **Groups:** The Groups category creates or manages groups. This category includes these tasks:
    - Create Group – Allows the TOE user to create a group, specifying information about the group and its initial members and administrators.
    - Delete Group – Allows the TOE user to remove a group.
    - Modify Group – Allows the TOE user to modify properties of a group except for its members.
    - Modify Group Members – Allows the TOE user to modify the members or administrators of a group.
    - View Group – Allows the TOE user to view the information about a group, including its membership.
- **Policies:** The Policies category manages Identity Policy Sets and Password Policies. This category contains the following tasks:
    - Create Identity Policy Set – Allows the TOE user to configure a set of changes that can be made when a TOE user meets a certain condition. For example, this can be used to define entry conditions for granting automatic assignment to an admin role. It can also be used to identify admin roles which are mutually exclusive.
    - Delete Identity Policy Set – Allows the TOE user to remove a policy.
    - Modify Identity Policy Set – Allows the TOE user to modify information about the policy (such as its name), the TOE users who are allowed to modify it, and the changes it performs.
    - View Identity Policy Set – Allows the TOE user to view basic information about a policy, the owners of the policy, and the actions it can perform.
    - Synchronize User – Allows the TOE user to check all policies against one or more TOE users so that new policies can be applied and old policies can be removed.
    - Create Password Policy – Allows the TOE user to compose a set of restrictions that define password composition and expiration, as well as the priority the policy takes over others.
    - Delete Password Policy – Allows the TOE user to remove a password policy.
    - Modify Password Policy– Allows the TOE user to modify the restrictions and priority of a password policy.
    - View Password Policy – Allows the TOE user to view information about a given password policy.
- **Provisioning Roles:** A TOE user uses the Provisioning Roles category to create and manage provisioning roles, which can be assigned to endpoint users. This category contains the following tasks:
    - Create Provisioning Role – Allows the TOE user to create a provisioning role which can be assigned to endpoint users.

- Delete Provisioning Role – Allows the TOE user to remove a provisioning role.
- Modify Provisioning Role – Allows the TOE user to modify the account templates which comprise a provisioning role.
- Modify Provisioning Role Members/Administrators – Allows the TOE user to modify the membership of the provisioning role.
- Reset Provisioning Role Owners – Allows the TOE user to change the conditions which define ownership of a provisioning role.
- View Provisioning Role – Displays the properties of a provisioning role.
- View Provisioning Role Members/Administrators – Displays the membership of a provisioning role.

- **System:** A TOE user can use the System category to monitor the status of tasks and events that have been submitted for processing. This category involves the following tasks:
  - Connection Management – Allows the TOE user to configure the connection information to the external database server.
  - Select Box Data – Allows the TOE user to specify an XML file that can be used to populate dropdown menus in the User Console with custom data.
  - View Submitted Tasks – Displays the status of tasks which have been submitted for processing.
  - Create Logical Attribute Handler – Allows the TOE user to create logical attribute handlers, which are Java objects that process logical attribute data and are written using the Logical Attribute API. For example, when performing this task, a TOE user can create a logical attribute handler that can convert physical attribute data from the user store into logical attribute data that can be displayed on the task screen.
  - Delete Logical Attribute Handler – Allows the TOE user to delete existing logical attribute handlers. Deleting logical attribute handlers deletes the Java objects that process logical attribute data but does not affect the data store.
  - Modify Logical Attribute Handler – Allows the TOE user to modify existing logical attribute handlers.
  - View Logical Attribute Handler – Allows the TOE user to view the properties of existing logical attribute handlers.
- **Users:** The User category creates or manages TOE user accounts. This category includes these tasks:
  - Certify User – Allows the TOE user to certify the roles assigned to a selected TOE user. This is the mechanism by which the certification process can be accomplished.
  - Create User – Allows the TOE user to create a new account on the TOE. This includes assigning the new TOE user group and role membership if permissions allow.

- Delegate Work Items – Allows the TOE user to delegate work items from one TOE user to another.
- Delete User – Allows the TOE user to remove an account on the TOE.
- Enable/Disable User – Allows the TOE user to alter a TOE user's disabled state.
- Modify User – Allows the TOE user to modify another TOE user's personal information, as well as manage their endpoint user accounts if they have any. This includes endpoint password change, endpoint account enable/disable, and create/view/modify/delete endpoint account operations.
- Reset User Password – Allows the TOE user to change another TOE user's password without requiring input of the current password.
- View User – Displays a selected TOE user's profile.

Note: *Any task that appears in Identity Manager but is omitted from the tasks defined here is a task created by an Identity Manager TOE user. Depending on the admin roles assigned to the TOE user, they see only certain admin tasks. For example, if they have the default Group Manager role, they see five tasks under Groups. The two exception to this are the Access Roles tasks because Access Roles have been excluded from the evaluated configuration and the Provisioning Synchronization tasks because they are only used when the user store and provisioning directory are separate.*

8.1.2.8 Identity Manager Directories

An Identity Manager directory describes how objects such as users and groups are stored in the user directory and represented in Identity Manager. An Identity Manager directory is associated with one or more Identity Manager environments. The Identity Manager directory is created during the initial configuration of the TOE and is not modified during its operation.

8.1.3 Security Audit

8.1.3.1 Audit Capabilities

Identity Manager includes auditing capabilities that allow auditors to monitor the activity in an Identity Manager environment. This information is stored in an audit database. The amount and type of information that is stored in the audit database is configurable and in the evaluated configuration will be established as part of the installation.

The TOE does not include any native capabilities to review or access the audit records. The responsibility for this functionality is assumed by the operational environment.

Audit data provides a historical record of operations that occur in an Identity Manager environment. To audit data in Identity Manager, the following are needed:

- An auditing database
- An audit settings file

A complete list of the data in an audit record can be found in tables 6-3 and 6-4. An audit record logs the name and description of the event, the date/time it was created, the state of the event (which includes ultimate success and failure of the event in addition to other states that are logged during processing of the event), and the distinguished name of the TOE user which caused the event to happen. Because the audit settings file is loaded when the TOE starts up, auditing cannot be changed or disabled while the TOE is operational.

8.1.3.2 Audit Database

The Audit Database is the environmental store where all of the CA Identity Manager logging records are sent for storage and future analysis. For a listing of the data that is audited in the evaluated configuration, refer to tables 6-2, 6-3, and 6-4.

When the Identity Manager Installer is used, Identity Manager configures a connection to a single database, called the Identity Manager Database, and creates a separate database instance for auditing. During operation of the TOE, the Identity Manager Database will be used by the TOE, which will query, create, modify, and delete information based on the tasks that are performed by users. The operations that a TOE user can perform on the database depend on their admin role(s). For more information about which roles can perform which operations on the Identity Manager Database, refer to tables 6-10, 6-11, and 6-12. The next section (Data Protection) provides a more detailed discussion on how these admin roles apply to TOE users.

Note: *The Identity Manager Database also includes database instances for other Identity Manager functionality, including task persistence, workflow, and reporting. For scalability purposes, a new, separate instance of a database for auditing can be created.*

8.1.3.3 Audit Settings

Audit settings are configured in an audit settings file. An audit settings file determines the amount and type of information that Identity Manager audits. An audit settings file can be configured to do the following:

- Enable auditing for an Identity Manager environment.
- Enable auditing for some or all of the Identity Manager events generated by admin tasks.
- Record event information at specific states, such as when an event completes or is cancelled.
- Log information about attributes involved in an event. For example, attributes that change during a ModifyUserEvent event can be logged.
- Set the audit level for attribute logging.

The audit settings file is configured during initial setup of the TOE and statically loaded into memory, so is not part of the evaluated configuration. However, it's relevant to mention it because it identifies the auditing capabilities of Identity Manager following the configuration process. In addition, because the audit settings file determines the configuration for audit generation and is only configured during installation, auditing for the TOE is always active as long as the TOE itself is active.

8.1.3.4 Audit Events

When performing activities on the TOE, audit events are called which are used to generate an audit trail for use of the TOE. Table 6-2 lists all of the audit events which can be triggered in the evaluation of the TOE. Listed below is a mapping of the tasks which can be performed on the TOE to the audit event(s) that can be generated when that task is performed.

Task	Events Generated for the Task
Begin Certification Process	CertificationRequiredNotificationEvent CertificationStatusRequiresCertificationEvent
Certify User	CertificationStatusCertifiedEvent CertificationStatusInCertificationEvent CertifyRoleEvent ForgottenPasswordEvent ModifyUserEvent RemoveGrantorOnAccessRoleEvent RemoveGrantorOnAdminRoleEvent RemoveGrantorOnProvisioningRoleEvent ResetPasswordEvent RevokeAccessRoleEvent RevokeAdminRoleEvent RevokeProvisioningRoleEvent
Change My Password	ForgottenPasswordEvent ModifyUserEvent ResetPasswordEvent
Create Admin Role	CreateAdminRoleEvent
Create Admin Task	CreateAdminTaskEvent
Create Group	CreateGroupEvent AddGroupAdminEvent AddGroupAdminGroupEvent AddGroupToGroupEvent AddToGroupEvent
Create Identity Policy Set	CreateIdentityPolicySetEvent
Create Logical Attribute Handler	CreateLAHDefinitionEvent
Create Password Policy	CreatePasswordPolicyEvent
Create Provisioning Role	CreateProvisioningRoleEvent

Create User	CreateUserEvent AddGrantorOnAccessRoleEvent AddGrantorOnAdminRoleEvent AddGrantorOnProvisioningRoleEvent AddGroupAdminEvent AddToGroupEvent AssignAccessRoleEvent AssignAdminRoleEvent AssignProvisioningRoleEvent
Delete Admin Role	DeleteAdminRoleEvent
Delete Admin Task	DeleteAdminTaskEvent
Delete Group	DeleteGroupEvent
Delete Identity Policy Set	DeleteIdentityPolicySetEvent
Delete Logical Attribute Handler	DeleteLAHDefinitionEvent
Delete Password Policy	DeletePasswordPolicyEvent
Delete Provisioning Role	DeleteProvisioningRoleEvent
Delete User	DeleteUserEvent
Enable/Disable User	DisableUserEvent EnableUserEvent
End Certification Process	CertificationNonCertifiedActionCompletedNotificationEvent CertificationNonCertifiedActionPendingNotificationEvent CertificationStatusNotCertifiedEvent
External Task	ExternalTaskEmptyEvent
External Group Task	ExternalTaskGroupEvent
External Organization Task	ExternalTaskOrgEvent
External User Task	ExternalTaskUserEvent
Forgotten Password	ForgottenPasswordEvent
Forgotten User ID	ForgottenUserIDEvent
Modify Admin Role	ModifyAdminRoleEvent
Modify Admin Role Members/Administrators	AddGrantorOnAccessRoleEvent AssignAccessRoleEvent RemoveGrantorOnAccessRoleEvent RevokeAccessRoleEvent
Modify Admin Task	ModifyAdminTaskEvent
Modify Group	AddGroupAdminEvent AddGroupAdminGroupEvent AddGroupToGroupEvent AddToGroupEvent ModifyGroupEvent RemoveFromGroupEvent RemoveGroupAdminEvent RemoveGroupAdminGroupEvent RemoveGroupFromGroupEvent

Modify Group Members	AddGroupToGroupEvent AddToGroupEvent RemoveFromGroupEvent RemoveGroupFromGroupEvent
Modify Identity Policy Set	ModifyIdentityPolicySetEvent
Modify Logical Attribute Handler	ModifyLAHDefinitionEvent
Modify My Groups	AddGroupAdminEvent AddToGroupEvent RemoveFromGroupEvent RemoveGroupAdminEvent
Modify My Profile	ModifyUserEvent
Modify Password Policy	ModifyPasswordPolicyEvent
Modify Provisioning Role	ModifyProvisioningRoleEvent
Modify Provisioning Role Members/Administrators	AddGrantorOnProvisioningRoleEvent AssignProvisioningRoleEvent RemoveGrantorOnProvisioningRoleEvent RevokeProvisioningRoleEvent
Modify User	ModifyUserEvent AddGrantorOnAccessRoleEvent AddGrantorOnAdminRoleEvent AddGrantorOnProvisioningRoleEvent AddGroupAdminEvent AddToGroupEvent AssignAccessRoleEvent AssignAdminRoleEvent AssignProvisioningRoleEvent ForgottenPasswordEvent RemoveFromGroupEvent RemoveGrantorOnAccessRoleEvent RemoveGrantorOnAdminRoleEvent RemoveGrantorOnProvisioningRoleEvent RemoveGroupAdminEvent ResetPasswordEvent RevokeAccessRoleEvent RevokeAdminRoleEvent RevokeProvisioningRoleEvent
Reset Admin Role Owners	ModifyAdminRoleEvent
Reset Provisioning Role Owners	ModifyProvisioningRoleEvent
Reset User Password	ModifyUserEvent ForgottenPasswordEvent ResetPasswordEvent
Self Registration	SelfRegisterUserEvent
Send Certification Reminder Notification	CertificationRequiredReminderNotificationEvent

Send Final Certification Reminder Notification	CertificationRequiredFinalReminderNotificationEvent
Synchronize User	ModifyUserEvent AddGrantorOnAccessRoleEvent AddGrantorOnAdminRoleEvent AddGrantorOnProvisioningRoleEvent AddGroupAdminEvent AddToGroupEvent AssignAccessRoleEvent AssignAdminRoleEvent AssignProvisioningRoleEvent ForgottenPasswordEvent RemoveFromGroupEvent RemoveGrantorOnAccessRoleEvent RemoveGrantorOnAdminRoleEvent RemoveGrantorOnProvisioningRoleEvent RemoveGroupAdminEvent ResetPasswordEvent RevokeAccessRoleEvent RevokeAdminRoleEvent RevokeProvisioningRoleEvent
View Admin Role	ViewAdminRoleEvent
View Admin Role Members/Administrators	ViewAdminRoleEvent
View Admin Task	ViewAdminTaskEvent
View Group	ViewGroupEvent
View Identity Policy Set	(none)
View Logical Attribute Handler	ViewLAHDefinitionEvent
View My Roles	ViewUserEvent
View My Submitted Tasks	(none)
View Password Policy	ViewPasswordPolicyEvent
View Provisioning Role	ViewProvisioningRoleEvent
View Provisioning Role Members/Administrators	ViewProvisioningRoleEvent
View Submitted Tasks	(none)
View User	ViewUserEvent

Table 8-1 Audit Events by Task

8.1.4 Data Protection

Identify Manager utilizes Role-Based Access Control (RBAC) to authorize TOE users access to various functions based on that individual's admin role. Whether or not a TOE user is assigned to an admin role is governed by one or more policies for that role. For example, access to an admin role can be explicitly granted to a TOE user, or it can be assigned by virtue of that TOE user's title or membership within a certain group. In addition, each policy can then determine the scope of allowed operations within that particular task. The combination of the TOE user's role within the TOE and the policy that allows them membership into that role determines the extent to which they are able

to interact with a particular task. This defines the User Policy, which is used by the TOE in order to make access control decisions. No additional rules are used to explicitly allow TOE users access to tasks.

Identity Manager provides three types of roles, two of which will be subject to evaluation:

- User management roles are called *admin roles*.
- Application access roles are called *access roles*. Access roles will not be part of the evaluated configuration of the TOE because they are used only when the TOE is integrated with SiteMinder. Since SiteMinder integration is not being evaluated, access roles will not be either.
- Account assignment roles are called *provisioning roles*.

The TOE provides default admin roles if minimal customization is desired, but the capability to create new roles based on combinations of tasks is also available. The following table illustrates the characteristics common to these roles:

Role Characteristic	Details
Role Profile	Define a name and description for the role and set Enabled status.
Tasks	Include admin or access tasks.
Account Templates	Include account templates that define accounts that exist in endpoints (provisioning roles only).
Member Policies	For each member policy, define: • Member Rules -- Who can use the role • Scope Rules -- Which objects can a role member manage • Add Action -- What happens to the profile of a TOE user who becomes a member • Remove Action -- What happens to the profile of a TOE user who is removed as a member
Admin Policies	For each admin policy: • Admin Rules -- Who can manage the TOE users as members or administrators • Scope Rules -- Which TOE users can the TOE user manage as members or administrators • Add Action -- What happens to the profile of a TOE user who becomes an administrator

	• Remove Action -- What happens to the profile of a TOE user who is removed as an administrator
Owner Rules	Define who can modify the role.

Table 8-2: Global Role Characteristics

8.1.4.1 Admin Roles

In Identity Manager, user store objects (users and groups) are managed through admin roles. Admin roles are also used to manage the roles and tasks through which user store objects are managed. For example, admin roles are used to modify profile attributes of TOE users, give TOE users options for managing their own accounts, and to approve tasks that use workflow.

An Identity Manager environment is viewed through the Identity Manager user console. A TOE user's assigned admin roles determine what a TOE user sees in that console as shown in the following table:

Assigned Role Type	**Format of the Identity Manager User Console**
Roles for managing more than one type of object	The category list with one item for each type of object which can be managed
Roles for managing one type of object, such as Users	The tasks for that object (such as Modify User) *without* a category list
An approval role	The Work list screen Appears if the TOE user has tasks pending approval (for example, self-registering TOE users need approval)

Table 8-3: User Console Formats

Admin roles consist of admin tasks, which represent granular capabilities for managing objects. For example, a user object could be managed by using these admin tasks:

- Create User
- View User
- Modify User
- Reset User Password

Identity Manager comes with a variety of preconfigured admin roles. A list of these roles as well as the tasks that role can be performed is shown in the table below. Note that these tasks are taken from the set of default admin tasks listed in section **Default Admin Tasks**.

Role	Available Tasks
Admin Role Manager	Create Admin Role Create Admin Task Delete Admin Role Delete Admin Task Modify Admin Role Modify Admin Role Members/Administrators Modify Admin Task Reset Admin Role Owners View Admin Role View Admin Role Members/Administrators View Admin Task
Certify Manager	Certify User
Certification Process Manager	Begin Certification Process End Certification Process Send Certification Reminder Notification Send Final Certification Reminder Notification
Delegation Manager	Delegate Work Items This role has scope over all users.
Group Manager	Create Group Delete Group Modify Group View Group
Password Manager	Reset User Password View User
Provisioning Role Manager	Create Provisioning Role Delete Provisioning Role Modify Provisioning Role Modify Provisioning Role Members/Administrators Reset Provisioning Role Owners View Provisioning Role View Provisioning Role Members/Administrators
Security Manager	Enable/Disable User Reset User Password View User

Self Delegator	Out of Office In addition, a Self Delegator (the delegator) can specify that another user (the delegate) be allowed to approve tasks in the delegator's work list. This role has scope over all users.
Self Manager	Change My Account Change My Password Modify My Groups Modify My Profile View My Roles View My Submitted Tasks View My Work List
User Manager	Create User Delete User Modify User View User Modify Group Members
(no role)	Create Endpoint Delete Endpoint Modify Endpoint View Endpoint Create Account Template Delete Account Template Modify Account Template View Account Template Create Identity Policy Set Delete Identity Policy Set Modify Identity Policy Set View Identity Policy Set Synchronize Users Create Password Policy Delete Password Policy Modify Password Policy View Password Policy Connection Management Select Box Data View Submitted Tasks Create Logical Attribute Handler Delete Logical Attribute Handler Modify Logical Attribute Handler View Logical Attribute Handler Approve Create Group Approve Delete Group Approve Modify Group Membership Approve Modify Provisioning Role Membership Approve Self Registration Approve Create User Approve Delete User Approve Modify User

Table 8-4: Default Admin Tasks

Note that Identity Manager includes a System Manager role by default as well which has global superuser functionality. Following initial configuration of the TOE, this role is not operationally necessary and so it will be disabled in the evaluated configuration. Also note that the Access Role Manager role and any functionality related to access roles has been omitted as well. When a new TOE user is created, it has no roles by default. A TOE user with the ability to create TOE users must assign roles to the new TOE user and make scope assignments based on their intended authority. This is more secure than creating a TOE user with global roles and scope and forcing the creator to remove authority.

Admin tasks include *events*, actions that Identity Manager performs to complete the task. A task may include multiple events. For example, the Create User task may include events that create the TOE user's profile, adds the TOE user to a group, and assigns roles.

A TOE user with the ability to modify other TOE users can change the attributes, roles, and scope of other TOE users, provided they have the allowed scope to do so. This includes the ability to revoke roles assigned to that TOE user. If a TOE user has a role revoked while they are accessing the TOE, the revocation will take effect following the next page load.

8.1.4.2 Provisioning Roles

In Identity Manager, an authorized TOE user will provide additional accounts to endpoint users by using provisioning roles. Provisioning roles contain account templates, which define accounts that exist in managed endpoints, such as an email server. Once endpoint users are in Identity Manager, provisioning roles can be assigned to some of those users. The endpoint user receives the accounts defined by the templates in the role.

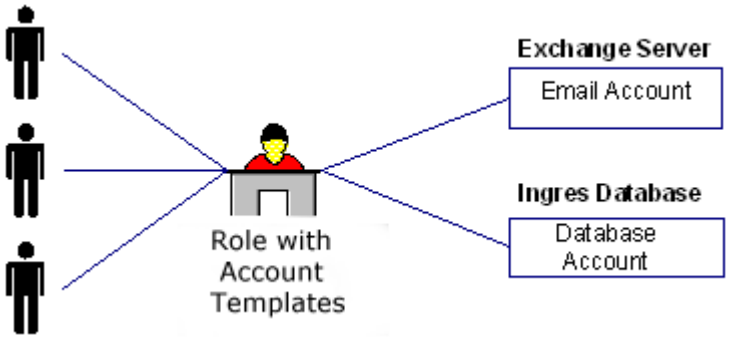

Figure 8-1: Account Templates

The account templates define the characteristics of the account. For example, an account template for an Oracle account might define the tablespaces owned by that account. The account templates also define how user attributes are mapped to accounts.

To be able to use provisioning roles, the Provisioning Server must be installed with the Identity Manager server. Account templates can then be created via the User Console and propagated to the Provisioning Server using the LDAP protocol.

There are no pre-defined provisioning roles; instead, they are created by TOE users who have authorization for the Create Provisioning Role task. These provisioning roles are based on the needs of the organization using Identity Manager. A provisioning role contains the following information:

- Role name
- Endpoint type(s) provisioned
- For each endpoint type, at least one account template (a UNIX endpoint type can have multiple account templates for different desired configurations on different machines in the role, so it's not a strict one-to-one relationship)
- Admin rules which control who can manage members and administrators of the role
- Owner rules which control who can modify the role

Note that an admin doesn't have to be an owner, and vice versa.

8.1.4.2.1 Account Template Overview

To simplify account management, it is possible create and maintain accounts using account templates, which are associated with one or more provisioning roles. Account templates contain the attributes that are used to create accounts. It is possible to define attributes using rule strings or values.

Using account templates, the following activities can be performed:

- Control what account attributes endpoint users have on an endpoint when their accounts are created
- Combine account attributes from different roles, so endpoint users have only one account, on a specific endpoint account attribute with all the necessary account attributes
- Create or update account attributes as endpoint users change roles
- Synchronize account attributes so endpoint users have only the attributes they need
- Perform queries to see which accounts are to be created, updated, or deleted during a synchronization operation
- Determine which account attributes can be synchronized with roles and which cannot

8.1.4.2.2 Account Template Attributes

Account templates include two types of attributes:

- *Capability attributes* represent account information, such as storage size, quantity, frequency limits, or group memberships.
- *Initial attributes* represent all information that is initially set for an account, such as account name, password, and account status and personal information such as name, address and phone numbers.

Accounts are considered in synchronized with their account templates when all the capability attributes are synchronized. These are attributes that differ from endpoint-type to endpoint-type such as group memberships, privileges, quotas, login-restrictions, which control what the endpoint user can do when logging into the account.

Other account attributes are not updated by synchronization. They are initialized from the account templates during account creation and can also be updated during propagation functions. The provisioning server provides two propagation functions (an immediate update of accounts at the time the account template is changed and an update of accounts at the time endpoint user attributes change).

8.1.5 Cryptographic Communication

The TOE can be configured to perform several different types of encryption. When operating in the evaluated configuration, the TOE uses AES encryption utilizing 256-bit keys. Keys are generated in accordance with FIPS PUB 197 as asserted by the vendor; however, the cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

The following are situations in which encryption is employed:

- When the contents of an environment or directory are dumped out, the credentials are encrypted in the exported version.
- When new TOE users are created, the cleartext password used during the creation has to be preserved for use in the synchronization of the various provisioned endpoints, since it is not recoverable from the directory. It is stored in encrypted form in the task session for the life of the create task.
- When objects that use the TOE's JDBCManagedObject implementation are written to the object store, any fields on the objects marked as requiring encrypt on write are encrypted before storage to the DB. They are decrypted on read. This includes such things as the passwords of connection objects.

- In order to encrypt communications between remote TOE users and the TOE itself, the environmental application server must be configured to utilize HTTPS.

Note that there is no specific key rollover process used, keys are simply generated prior to use and overwritten when no longer used.

While there are many other situations where Identity Manager can employ encryption, these situations are beyond the scope of the evaluation because components such as CA SiteMinder are required for them.

8.1.6 Other Identity Manager Components

8.1.6.1 Connectors

Identity Manager Connectors run as part of the wider Provisioning Server architecture and communicate with the systems managed in the evaluated configuration. A connector acts as a gateway to a native endpoint type system technology. Connectors manage the objects that reside on the systems. Managed objects include accounts, groups, and optionally, endpoint type specific objects.

Connectors are installed on the Connector Server and some components are installed on the Provisioning Server (for example, Server plug-in).

The Java Connector Server (Java CS) is a server component which handles hosting, routing to, and management of Java connectors. Java connectors are used in the evaluated configuration because they utilize the native APIs of endpoints to interact with them directly. This means that the untrusted endpoints do not require agents to be installed on them to act as a translator.

In order to verify that the only changes made to endpoint accounts are mediated by the TOE, Identity Manager is capable of running an Explore and Correlate process. This process compares the baseline configuration of the Provisioning Directory against the actual configuration of endpoints and corrects any deviations. This prevents system administrators from accessing endpoints directly in order to modify their configurations in a way that could potentially violate the rules set up on the TOE. For example, a user may be given administrative privileges directly on the endpoint while they belong to a group on the TOE that denies them these privileges.

8.1.6.2 User Store and Provisioning Directories

To provide options for managing users and automatic provisioning of additional accounts for those users, Identity Manager coordinates two user stores:

- The Identity Manager user store, the user store maintained by Identity Manager. Typically, this is an existing store that contains the user identities that an organization needs to manage.

  The user store can be an LDAP directory or a relational database. In the evaluated configuration, it will be an instance of CA Directory, which utilizes an LDAP interface.

  In the Management Console, an Identity Manager Directory object is created to connect to the user store and to describe the user store objects that Identity Manager will maintain.

- The Provisioning Directory is the user store maintained by the Provisioning Server. In the evaluated configuration, it is the same instance of CA Directory that is used for the Identity Manager user store above. The Provisioning Directory includes endpoint user accounts, which associate users in the Provisioning Directory with accounts on endpoints such as LDAP, Oracle, and SAP.

Only some TOE users have a corresponding endpoint user account. When a TOE user receives a provisioning role, the Provisioning Server creates an endpoint user. When the TOE performs tasks related to individual users (such as provisioning or creating new user accounts) the user store will be used by the TOE, which will query, create, modify, and delete information based on the tasks that are performed.

Combined User Store and Provisioning Directory

If an instance of CA Directory is selected for both user stores, the directory functions as both a user store and provisioning directory. TOE users created in Identity Manager are stored in that directory, but that directory can still be modified as a provisioning directory. For example, an endpoint user attribute can be selected for use in the Provisioning Server for a specific user attribute used by Identity Manager.

The following figure shows the use of one directory for both the user store and provisioning directory.

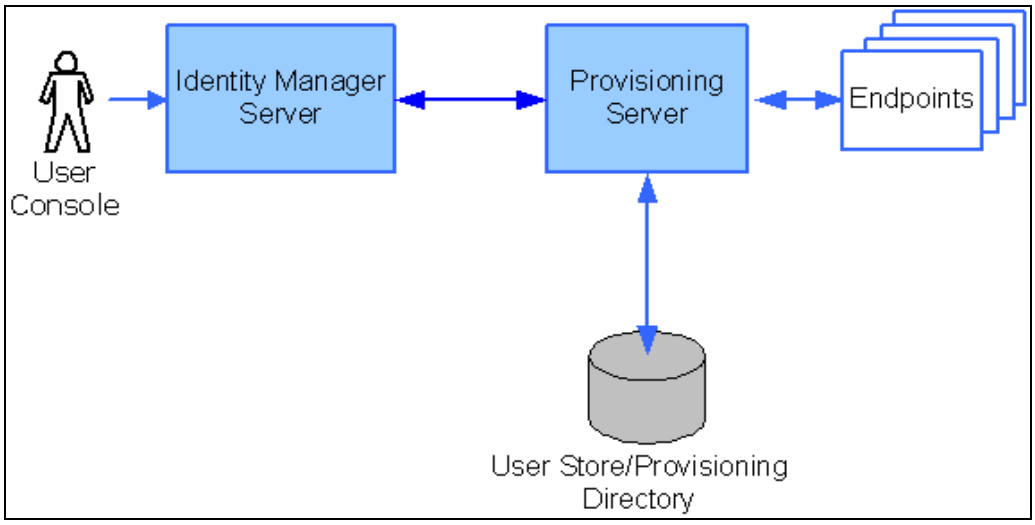

Figure 8-2: Combined User Store and Provisioning Directory

In this situation, changes to the user store/provisioning directory can be initiated by Identity Manager, the Provisioning Server, or an endpoint.

8.1.6.3 Provisioning Roles Management

In the User Console, the Provisioning Role Administrator or other authorized TOE user can create and manage provisioning roles by choosing Roles and Tasks and selecting a task under Provisioning Roles. Tasks exist for the standard operations, such as making a TOE user a member of a role and modifying or deleting a role.

The Provisioning Role Administrator or other authorized TOE user can create a provisioning role once the following role requirements have been determined:

- Which Identity Manager environment has endpoint users who need other accounts
- Which accounts will be associated with the role
- Who will be the members, administrators, and owners of the role

Although provisioning roles are managed in Identity Manager, some provisioning roles may have been created in Provisioning Manager or an external application. For any provisioning role created outside of Identity Manager, the Provisioning Role Administrator or other authorized TOE user can reset the role owner to be an Identity Manager administrator, so it can be managed in Identity Manager.

8.1.6.4 Workflow

A workflow process is one or more steps that must be performed before Identity Manager can complete a task that is under workflow control. To place a task under workflow control, a TOE user with the ability to create or modify an admin task can specify that workflow should be enabled for that task. The TOE user then applies a workflow process

which specifies which events in the task require approval and who can approve them. A job is a runtime instance of a workflow process.

A workflow process consists of one or more steps, called activities that must be performed in order to accomplish some business task, such as creating or modifying an employee TOE user account. Typically, a workflow process includes one or more manual activities which require an authorized TOE user, or participant, to approve or reject the task.

A participant is a person who is authorized to perform a workflow activity. In Identity Manager, participants are also called approvers, since they must approve or reject the task under workflow control. A participant resolver is a rule or set of criteria for determining who the participants are.

The individual manual activities in a workflow are called work items in Identity Manager. A work list is a workflow-generated list of approval tasks, or work items, that appears in the Identity Manager User Console of the participant authorized to approve the task.

The Workflow Policy determines how access to workflow is given. It's the combination of a task being enabled for workflow and a TOE user who can modify that task who is able to designate approvers. If the approvers have a delegation role, they may delegate their approval to another TOE user. Alternatively, a TOE user who can modify the task can reassign the approvers to another TOE user while they are active. Approvers are placed on an access control list (ACL) for the task, which determines the TOE users for which access to the workflow for that task is granted. These TOE users may delegate their approval ability if they have the appropriate admin role, allowing another TOE user to serve as an approver in their place. No additional rules are used to grant access to workflow approval. All TOE users who have not had access granted in this manner are automatically denied access.

To demonstrate the basic functionality of workflow, Identity Manager has a default workflow process called online requests, which is described below.

8.1.6.4.1 Work Lists and Work Items

A work list is a list of work items (or approval tasks) that appears in the User Console of the participant authorized to approve the task. Work items correspond to manual activities in a workflow process. Work items are represented as rows in the work list.

Work items can be added to a work list in the following ways:

- A participant resolver determining a list of approvers.
- Receiving delegated work items from another TOE user.
- Reassigning it to another TOE user.

Work items can be removed from a work list in the following ways:

- Completing (approving or rejecting) the work item.
- Reassigning it to another TOE user.
- Reserving it. This removes it from the work list of all other participants.

The information tabs that appear on a work item depend on whether the work item was generated by workflow under task-level or event-level control:

- **Profile** – Provides profile information about the object affected by the event (event-level only).
- **Task Details** – Provides detailed information for all events within the task (task-level only).
- **Approvers** – Lists all individual approvers and delegators for the task or event (task-level and event-level)

A TOE user's work list appears automatically when they log into the Identity Manager User Console if they have been assigned as a participant to approve tasks (or work items) initiated by other TOE users.

8.1.6.4.2 Reserving Work Items

A work item can be reserved by a delegator to "check it out" and remove it from the work list of other participants. Reserving a work item holds it for the TOE user performing the reservation. If multiple TOE users can act on a work item, reserving it is a way for one of the TOE users to claim ultimate authority over it.

If the reserving TOE user releases the work item, it becomes available again on the work list of other participants. If the reserving TOE user approves or rejects the work item, it is completed, and no longer available to other participants.

8.1.6.4.3 Reassignment and Reserved Work Items

If a TOE user has a work item reserved while it is reassigned, the TOE user keeps it reserved. But if the TOE user then releases that work item, he loses access to it.

A TOE user can reassign, reserve, or release another TOE user's work item, but cannot approve or reject another TOE user's work item. Only the assigned work item participant can do that.

8.1.6.4.4 Delegation and Reserved Work Items

While a delegation is active, either the delegate or the delegator may reserve a work item. A work item reserved by one TOE user cannot appear on another TOE user's work list.

For example, if a delegate has a work item reserved while the delegation is withdrawn, the delegate keeps the work item reserved. But if the delegate then releases that work item, he loses access to it.

If a TOE user who is a delegate is deleted while the delegate has a work item reserved, the delegate still retains the work item. If the delegate then approves the work item, auditing can no longer determine who delegated it.

If a delegate has a work item reserved while the delegation is withdrawn, the delegate retains access until the work item is completed or released.

8.1.6.4.5 Delegating Work Items

Work item delegation lets a TOE user (the delegator) specify that another TOE user (the delegate) be allowed to approve tasks in the delegator's work list. A delegator can assign work items to another approver during periods when the delegator is "out of the office." Delegators retain full access to their work items during the delegation period.

Delegated work items are not changed in any way. Logging indicates whether a work item was delegated.

Delegation works by allowing the delegate to "impersonate" the delegator and view the items on the delegator's work list. When viewing a work list, delegates see their own work items as well as the delegator's work items.

Delegation is not transitive. A delegate can only see work items that the delegator has assigned directly. For example, If TOE user A delegates work items to TOE user B, and TOE user B delegates work items to TOE user C, TOE user C can only see work items belonging to TOE user B, and not any work items that may have been work items delegated to TOE user B by TOE user A.

Note: *Workflow approval delegation must be enabled before work items can be delegated to another TOE user. By default, delegation is disabled.*

8.1.6.4.6 Delegating for Another User

TOE users with the Delegation Manager role can delegate work items from one TOE user (the delegator) to another. For example, a TOE user may be out of the office unexpectedly, or a TOE user may need to assign a large workload to multiple TOE users.

TOE users with the Self Delegator role can only delegate work items for TOE users over whom they have scope. Similarly, they can only add or remove TOE users they manage from the list of delegates.

8.1.6.4.7 Reassigning Work Items

Reassignment allows TOE users with the ability to manage admin tasks to change the assignees of a work item after it is created. An authorized TOE user can:

- View another TOE user's work list
- Add and remove work item assignees
- Change the reserve status of work items

  For example, a TOE user can reassign a work item or release a reserved work item from a TOE user who is not acting on it.

If a TOE user has a work item reserved while it is reassigned, the TOE user keeps it reserved. But if the TOE user then releases that work item, he loses access to it.

If a delegate has a work item reserved while the delegation is withdrawn, the delegate retains access until the work item is completed or released.

Reassignment is performed on the Work Item Approvers tab, which displays a list of current work item approvers (or assignees). When reassignment is performed, the open work item is assigned to all approvers in the list. Therefore, to reassign a work item to a new assignee, the current assignee must be explicitly removed.

8.1.6.5 Identity Manager Events

An Identity Manager task may represent several smaller and possibly asynchronous operations, called events. One or more events are generated whenever an Identity Manager task (other than a workflow approval task) is performed.

Identity Manager audits events, enforces customer-specific business rules associated with events, and, when events are mapped to workflow processes, requires approval for events.

A task is not completed until all of its component events have been completed and, if workflow is used, approved. For example, when revoking TOE user attributes, the modification task is not complete until the final event for that task is processed. If the TOE user has an existing session on the TOE, they can continue to interact with the TOE with the privileges they had at the time they authenticated. In this case, the modification takes effect the next time the TOE user logs in to the TOE.

How Identity Manager Determines Task Status

A task is an administrative function that a TOE user can perform in Identity Manager. Tasks include events, actions that Identity Manager performs to complete the task. A task may include multiple events. For example, the Create User task may include events that create the TOE user's profile, adds the TOE user to a group, and assigns roles.

Identity Manager tasks and events can be associated with a workflow process, which determines how Identity Manager performs the required actions, and other custom business logic. Tasks may also be associated with other tasks, called nested tasks. In this case, Identity Manager processes the nested tasks with the original task.

The status of a task depends on the status of its associated events, workflow processes, nested tasks, and custom business logic.

8.1.6.6 Admin Tasks and Events

If multiple events are generated for a task, and the events are mapped to workflow processes, all the workflow processes must be completed before Identity Manager can complete the task.

Generally, events are independent of other events. However, some tasks are associated with a primary event and one or more secondary events:

- A failure of a primary event results in the automatic rejection of all of its secondary events. For example, if a CreateUserEvent fails, there is no need for the AddToGroupEvent to occur for the TOE user. It also results in the cancellation of the associated task.
- A failure of a secondary event does not affect the success or failure of any other events executed for the task or the execution of the task itself. For example, in a Create User task, an AddToGroupEvent may be rejected, meaning that the new TOE user cannot be added to a particular group. But the TOE user can still be created (CreateUserEvent) and even be added to other groups.

8.1.7 TOE Protection

8.1.7.1 TP-1 TSF domain separation

The Application Server maintains individual sessions associated with TOE users once they authenticate. The TSF maintains the TOE user's identification (ie username) as part of a session to prevent interference between TOE user actions. A TOE user's access to the TOE and TOE data is also determined upon session establishment by being associated with a role which has specific functions that can be performed. The TOE user sessions are not locked. CA Identity Manager allows multiple TOE users to make changes simultaneously. The configuration changes last saved by any session are enforced.

The TSF when invoked by the underlying host OS maintains a security domain that protects it from interference and tampering by untrusted subjects in the TOE's Scope of Control.

The Application Server, Provisioning Server, and Connectors are passive devices in that they indirectly connect to networks via other devices e.g. network interface. These components' protected domains include the software of which they are comprised.
In addition to the Application Server, Provisioning Server, and Connectors specific software, other software files such as configuration files are also stored on disk. These files are configured during the installation and setup of the TOE, and therefore access to them is not in the scope of evaluation.

Any actions on TOE files or processes are determined by the Operating System (OS) which the TOE has been installed on. This is because the TOE's processes run at the kernel level and the files are protected by the OS authorization mechanisms. In the evaluated configuration, the TOE will be installed by an administrator utilizing the Unix "root" or Windows Administrator user provided by the respective OS. A user which can gain local access to a machine were the TOE components are installed, must authenticate to the OS with the same permissions provided by these accounts to affect the TOE's processes or files. This is because any user with lesser privileges which tries to perform an action on the TOE's processes or files at the OS's user space level will be sent to the OS kernel for authorization. Once the kernel's authorization mechanisms recognize that the local user does not have permission, the kernel will reject the request. The underlying assumption regarding the operation of the TOE components are that they are maintained in a physically secure environment.

The environment provides encryption of TOE data and a secure communication path between the remote TOE users and the Application Server. The encryption is performed using the AES algorithm with 256 bit keys, and is in accordance with FIPS PUB 197 as asserted by the vendor; however, the cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

9 TOE Summary Specification Rationale

This section identifies the security functions provided by the TOE mapped to the security functional requirement components contained in this ST. This mapping is provided in the following table.

Security Function	Security Functional Components
User Data Protection	FDP_ACC.1(1) Subset Access Control
	FDP_ACC.1(2) Subset Access Control
	FDP_ACC.1(3) Subset Access Control
	FDP_ACF.1 (1) Security Attribute Based Access Control
	FDP_ACF.1 (2) Security Attribute Based Access Control
	FDP_ACF.1 (3) Security Attribute Based Access

Security Function	Security Functional Components
	Control
	FDP_IFC.1 Information Flow Control
	FDP_IFF.1 Simple Security Attributes
Identification and Authentication	FIA_ATD.1 User Attribute Definition
	FIA_SOS.1 Verification of Secrets
	FIA_UAU.2 User Authentication Before Any Action
	FIA_UID.2 User Identification Before Any Action
Security Audit	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
Security Management	FMT_MSA.1 Management of Security Attributes
	FMT_MSA.3 Static Attribute Initialization
	FMT_MTD.1(1) Management of TSF Data
	FMT_MTD.1(2) Management of TSF Data
	FMT_MTD.1(3) Management of TSF Data
	FMT_SMF.1 Specification of Management Functions
	FMT_SMR.2 Restrictions on Security Roles
	FMT_REV.1 Revocation
Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1 Cryptographic Operation
Trusted Path/Channels	FTP_TRP.1 Trusted Path

Table 9-1: Security Functional Components

9.1.1 User Data Protection

The User Data Protection functions of the TOE and the Operational Environment enforce the FDP_ACC.1(1), FDP_ACC.1(2), FDP_ACC.1(3), FDP_ACF.1(1), FDP_ACF.1(2), FDP_ACF.1(3), FDP_IFC.1, and FDP_IFF.1 requirements.

When a TOE user attempts to access the TOE, Identity Manager uses their role information to determine the tasks available for them to access. Within these tasks, the TOE user's scope, determined by the policy that assigned them their role information or an explicit authorization, is used to determine the scope of control of the TOE user's available operations. Compliance support can be used to define mutually exclusive roles or preconditions for a TOE user being assigned a role. In this manner, explicit denial of operations can be established.

Independently of this process, TOE users can be delegated the ability to be workflow approvers for specific tasks. TOE users have workflow approver roles that determine

what types of tasks they can approve, and delegation of this approval is determined by role and scope information. Workflow is essentially an information flow that forces tasks to be approved at certain points before their execution. If workflow is enabled for the TOE and applies to a certain tasks, the information flow will apply to that task as long as it has at least one approver assigned to.

The third means by which TOE data is protected is via the Task Execution Web Service. An authorized TOE user can run a web service application to perform a batch of automated commands on the TOE. The credentials of the TOE user running the application are sent to the TOE and checked against a separate access control list (ACL) stored in the Identity Manager database before the web service application can be run.

Provisioning roles are the process by which endpoint user accounts are assigned to an endpoint. A provisioning role identifies the type of endpoint account that will be created (such as a Unix account). The privileges assigned to that account are based on account templates. For example, an account template can be defined for a DBA, attached to a Unix account provisioning role, and then this role can applied to all DBA endpoint users by assigning the role to them. Provisioning roles and account templates are created by a TOE user with the appropriate task privileges.

Once an endpoint user account has been assigned to an endpoint via provisioning, it's the responsibility of that endpoint to protect its data from unauthorized access. A TOE user who accesses that endpoint should only be allowed to perform operations allowed to them by the initial provisioning assignment.

9.1.2 Identification and Authentication

The identification and authentication functions of the TOE and the Operational Environment enforce FIA_ATD.1, FIA_SOS.1, FIA_UAU.2, and FIA_UID.2.

The TOE provides TOE users with a username and password to authenticate to the TOE and stores their e-mail address so they can perform tasks which require authentication by answering a pre-defined security verification question. The TOE contains a configurable password policy mechanism to ensure that TOE user passwords are sufficiently secure for a given deployment. It also stores a password recovery question and answer in case of forgotten password. Other security attributes which pertain to TOE users are the enabled/disabled state of their account, the admin roles to which they belong, and the scope of task access they're assigned. A certain set of self-management tasks are referred to as public tasks due to the fact that authentication to the TOE is performed by answering a pre-defined security verification question. All other tasks require username/password authentication.

The Task Execution Web Service (TEWS) relies on a simple directory-based authentication to allow access to the TOE's web service API. When a web service application is run against the TOE, the taskContext value of the SOAP request identifies

the TOE user running the application and contains their username credential, which is used to identify and authorize their actions. In the evaluated configuration, the TEWS interface requires a third party application to authenticate users prior to granting access to the interface. The user will then provide their identity to the TOE for identification and to determine access control restrictions. Thus, TEWS interface will only identify a user that has already been granted access to the interface by the third party application. The identity provided to the TOE does not have to match the one provided to the third party application, nor will the identity provided to the TOE be authenticated. Therefore, the TEWS interface is expected to only be used by a user that has been assigned all privileges of the TOE in the evaluated configuration.

NOTE: *Although it was not validated through the evaluation, the vendor has asserted that the TOE can have the TEWS interface protected by CA SiteMinder. This would allow for TOE users to have their TOE identity be authenticated by SiteMinder, and then have all actions on the TOE be associated with their validated TOE identity.*

Password policies can be defined by a TOE user with the appropriate task privileges. Options such as password length, composition (such as "at least one number"), and regular expression formatting can be applied. When a password policy is applied to an environment, the TOE forces TOE user passwords to comply with the policy before they can proceed.

When an endpoint has been configured by the TOE's provisioning capabilities, they enforce the provisioned identification and authentication policies as if they had been configured directly on that endpoint (without using the TOE as an intermediary). Access to endpoints, therefore, is governed by the native I&A of the endpoints themselves.

9.1.3 Security Audit

The security audit function of the TOE and the Operational Environment enforce the FAU_GEN.1 and FAU_GEN.2. FAU_GEN.1 requires a reliable time-stamp, which is assumed to be provided by the Operational Environment.

When the TOE is first configured, an audit settings file is created to define the types of events that will be audited by the TOE and the conditions under which they're audited. Audit records are stored in the Audit DB and contain the fields shown in Tables 6-3 and 6-4. This includes, among other data, the timestamp of the event, subject identity, and outcome of the event.

When a task is performed by the TOE, it is is composed as a series of events. For example, performing the Modify Admin Role Members/Administrators task can involve one or more of the following events: AddGrantorOnAccessRoleEvent, AssignAccessRoleEvent, RemoveGrantorOnAccessRoleEvent, and RevokeAccessRoleEvent. These events are the audit events which are entered into the Audit DB as audit records.

Audit review is performed in the Operational Environment because the TOE lacks the capability to review audit data natively. The Operational Environment must therefore be configured in a manner that allows for only authorized individuals to review or modify the audit trail.

The TOE relies on the underlying operating system to provide accurate time stamps to be used for audit records.

9.1.4 Security Management

The security management function of the TOE enforces the FMT_MSA.1, FMT_MSA.3, FMT_MTD.1, FMT_REV.1, FMT_SMF.1, and FMT_SMR.2 requirements.

The TOE provides management capabilities through the User Console that are used remotely by TOE users. The capability to manage various attributes is limited by the allowed tasks and authorized scope of TOE users. For example, one degree of scope can be authority to perform a task on behalf of all members of a particular group. Another can be for a TOE user to only modify their own attributes.

Management functions of the TOE are accomplished via performing tasks. Roles are given a set of tasks they are authorized to perform and the TOE associates TOE users with one or more of these roles. TOE users are taken from the user store, which is defined as an LDAP directory using the XML Directory Configuration File during initial setup of the TOE.

Like performing any other management function, groups are defined by a TOE user with the appropriate task privilege. Groups can be based on a static set of members, a dynamic LDAP query that changes the group membership as the user store changes, or by aggregating multiple existing groups.

During initial configuration of the TOE, the default values for new pieces of TOE data are restrictive by default. For example, a new TOE user won't be assigned scope over all other TOE users by default on the Create User Task page. However, any TOE user with the ability to create data (as defined by the available tasks in table 6-11) in the Identity Manager database can override these default values.

There is a superuser account on the TOE by default, but in the evaluated configuration it will only be used in the initial configuration and then disabled. The TOE contains a number of default roles, but custom roles can be defined as well by combining policies (to determine membership conditions and scope of operations) and tasks.

Provisioning is managed by TOE users with Create/Modify Endpoint tasks assigned to them. This allows TOE users to apply account templates to endpoints and perform provisioning. Once an endpoint is created, endpoint user accounts are configured on it by managing provisioning roles and account templates.

When a TOE user has rights revoked, the action will be processed as soon as the task is completed. This is enforced on the TOE user upon the next page loaded in the User Console.

9.1.5 Cryptographic Support

The Cryptographic Support function of the TOE enforces the FCS_CKM.1, FCS_CKM.4 and FCS_COP.1 requirements.

The TOE uses AES encryption with 256-bit keys that operates in accordance with FIPS PUB 197. Encryption is performed when directories and environments are exported, when new TOE user passwords are created, and when database fields configured as "encrypt on write" are written to. To secure communications between the TOE and the TOE user web browser, the application server on which the TOE is installed must be configured for HTTPS.

All cryptography for this product has only been asserted as tested by the vendor. The testing of the specific cryptographic algorithms will not be tested as part of this evaluation.

9.1.6 Protection of the TSF

The TOE relies on the host operating system to provide reliable timestamps for audit records.

9.1.7 Trusted Path/Channels

The Operational Environment shall provide a path for communication between the TSF and remote TOE users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification and disclosure. The Operational Environment shall allow initial communication to the trusted path by remote TOE users, and it shall require the use of the trusted path for initial TOE user authentication and all other TSF mediated actions by the TOE user. This is accomplished by configuring the environmental application server to use HTTPS for remote browser sessions.

9.1.8 Provisioning

The Provisioning function of the TOE is a subset of the activities performed by the User Data Protection security function. Account templates are defined, provisioning roles are assigned, and endpoint users are given provisioning roles through the same mechanism that admin roles are created and assigned.

Applications called connectors translate provisioning commands issued by the TOE into the format used by the endpoints. Depending on the application type, some connectors

are installed directly on the endpoints themselves, while others are installed on a central environment server.

Once endpoints are provisioned, their access control and authentication mechanisms are no longer the responsibility of the TOE. A provisioned endpoint does not require communication with the TOE (unless further provisioning is required) because it acts as if it was configured locally by an administrator. This allows the operational environment to function in its intended manner if the TOE itself is in a failed state. In the evaluated configuration, the user store used by the Provisioning Server is the same logical user store used by the Application Server. This allows endpoint accounts and roles to be provisioned and assigned to TOE users without the need to introduce an additional mapping between multiple user stores.

9.1.9 Workflow

The Workflow function of the TOE enforces the FDP_ACC.1(2), FDP_ACF.1(2), FDP_IFC.1, and FDP_IFF.1 requirements. It is also a subset of the activities performed by the User Data Protection security function. A TOE user with Create or Modify Admin Task privileges is able to designate workflow approval steps for that task based on the events that occur as part of the task. These events are the same events which are audited. These TOE users also determine who can approve those steps.

When a task requires action from an approver to continue, that action is considered a work item. The TOE allows TOE users with delegation roles to give their work items to other TOE users. A TOE user with the ability to modify tasks can reassign work items for that task to different TOE users.

When multiple TOE users are capable of approving a single work item, it's possible for one TOE user to reserve the work item so that they can prevent the others from approving or rejecting it.

Workflow is defined as an information flow for the TOE in the sense that the work item flows through multiple subjects until final approval, at which point the task is performed. The information flow will not be performed unless the TOE is enabled for workflow, a task is associated with a workflow process, and the workflow process designates at least one approver for the given task. Explicit authorization of this information flow is defined by the workflow process applied to the task, which indicates the events which require approval and the set of TOE users which must do so in order for the events to proceed.

10 Security Problem Definition Rationale

10.1 Security Objectives Rationale

The following table provides a mapping with rationale to identify the security objectives that address the stated assumptions and threats.

Assumption	Objective	Rationale
A.ADMIN One or more TOE users will be assigned to install, configure and manage the TOE.	OE.ADMIN One or more TOE users will be assigned to install, configure and manage the TOE.	OE.ADMIN maps to A.ADMIN in order to ensure that authorized administrators install, manage and operate the TOE in a manner that maintains its security objectives.
A.PATCHES Users responsible for management of the operational environment exercise due diligence to update the TOE with the latest patches and patch the Operational Environment (e.g., OS and database) so they are not susceptible to network attacks.	OE.ADMIN One or more TOE users will be assigned to install, configure and manage the TOE.	OE.ADMIN maps to A.PATCHES in order to ensure that the authorized administrators properly patch the Operational environment in a manner that maintains its security objectives.
A.NOEVIL TOE users are not careless, wilfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.	OE.NOEVIL No TOE users are careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the organization's guidance documentation.	OE.NOEVIL maps to A.NOEVIL in order to ensure that there are no careless, willfully negligent, or hostile TOE users.
A.LOCATE The network the TOE will monitor and manage is isolated from any other network. No connections exist to other networks.	OE.LOCATE The TOE will be located on an isolated network with no connections to other networks.	OE.LOCATE maps to A.LOCATE in order to ensure the physical security in which the TOE operates.

Table 10-1: Assumption to Objective Mapping

Threat	Objective	Rationale
T.ACCESS TOE users could gain electronic access to protected resources by attempting to establish a connection that they are not permitted to perform.	O.ACCESS The TOE will provide measures to authorize TOE users to access specified resources once the user has been authenticated. TOE user authorization is based on access rights configured by other TOE users with the ability to configure them.	O.ACCESS (FDP_ACC.1(1), FDP_ACC.1(2), FDP_ACC.1(3), FDP_ACF.1(1), FDP_ACF.1(2), FDP_ACF.1(3), FDP_IFC.1, FDP_IFF.1, ADV_ARC.1) addresses T.ACCESS by specifying access restrictions on the protected TOE resources to authenticated users based on various policies.
	OE. FILESYS The security features offered by the underlying Operating System and Database protect the files used by the TOE and access to the audit records.	OE.FILESYS addresses T.ACCESS by ensuring that the underlying Operating System provides the capability to store and protect the files used by the TOE.
T.ADMIN_ERROR A TOE user may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.	O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	O.ROBUST_ADMIN_GUIDANCE (ALC_DEL.1, AGD_PRE.1, AGD_OPE.1) helps to mitigate T.ADMIN_ERROR by ensuring the TOE

Threat	Objective	Rationale
		administrators have guidance that instructs them how to administer the TOE in a secure manner and to provide the administrator with instructions to ensure the TOE was not corrupted during the delivery process. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is unsecure.
	O.MANAGE The TOE will provide authorized users with the resources to manage and monitor user accounts, TOE resources and security information relative to the TOE.	O.MANAGE (FIA_SOS.1, FMT_MSA.1, FMT_MSA.3, FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_SMF.1, FMT_SMR.2, FMT_REV.1) addresses T.ADMIN_ERROR by ensuring only authorized administrators can use the provided resources for managing and monitoring TOE user accounts, TOE resources and security information relative to the TOE.
T.AUDIT_COMPROMISE A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a TOE user's action.	OE.FILESYS The security features offered by the underlying operating system protect the files used by the TOE.	OE.FILESYS addresses T.AUDIT_COMPROMISE by ensuring that the underlying operating system provides the capability to store and protect the audit files used by the TOE.
	O.FILESYS The security features offered by the TOE protects the confidentiality of TOE data that is stored in the database.	O.FILESYS (FCS_CKM.1, FCS_CKM.4, FCS_COP.1) addresses T.AUDIT_COMPROMISE by ensuring that the TOE encrypts audit data prior to writing it to the database.
	OE.AUDIT The operational environment will provide a secure mechanism by which audit data can be reviewed.	OE.AUDIT addresses T.AUDIT_COMPROMISE by ensuring that the underlying operating system provides the capability to allow only authorized users the ability to review the audit data.
	O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its	O.AUDIT (FAU_GEN.1, FAU_GEN.2) addresses T.AUDIT_COMPROMISE by actually creating the audit records to be protected by the

Threat	Objective	Rationale
	security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.	operational environment.
T.EAVESDROPPING A malicious user could eavesdrop on network traffic to gain unauthorized access to TOE data.	O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.	O. EAVESDROPPING (FCS_CKM.1, FCS_CKM.4, FCS_COP.1) mitigates T.EAVESDROPPING by ensuring that all communication to and from the TOE or between components of the TOE are not sent unless they are encrypted.
T.MASK Users whether they be malicious or non-malicious, could gain unauthorised access to the TOE by bypassing identification and authentication countermeasures.	O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security objectives of the TOE.	O.AUDIT (FAU_GEN.1, FAU_GEN.2) addresses T.MASK by monitoring user activity on the TOE to ensure that misuse of the TOE is quickly detected and mitigated.
	OE.AUDIT The operational environment will provide a secure mechanism by which audit data can be reviewed.	OE.AUDIT addresses T.MASK by ensuring that the underlying Operating System provides the capability to allow only authorized users the ability to review the audit data generated by O.AUDIT.
	O.AUTH The TOE will provide measures to uniquely identify all TOE users and will authenticate the claimed identity prior to granting a user access to the resources protected by the TOE.	O.AUTH (FIA_UAU.2, FIA_UID.2) addresses T.MASK by providing measures to uniquely identify and authenticate users through Username/Password.
T.MASQUERADE A user may masquerade as an TOE user or an authorized IT entity to gain access to data or TOE resources.	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate.	O.ROBUST_TOE_ACCESS (FDP_ACC.1(1), FDP_ACC.1(2), FDP_ACC.1(3), FDP_ACF.1(1), FDP_ACF.1(2), FDP_ACF.1(3), FDP_IFC.1, FDP_IFF.1, FIA_ATD.1, FIA_UAU.2, FIA_UID.2, addresses T.MASQUERADE by controlling the logical access to the TOE and its resources. By constraining how and when authorized users can access the TOE, and by

Threat	Objective	Rationale
		mandating the type and strength of the authentication scheme, this objective helps mitigate the possibility of a user attempting to login and masquerade as an authorized user.
T.UNAUTH Users could gain unauthorised access to the TOE or its data stores by bypassing identification and authentication requirements.	O.AUTH The TOE will provide measures to uniquely identify all TOE users and will authenticate the claimed identity prior to granting a user access to the resources protected by the TOE.	O.AUTH (FIA_UAU.2, FIA_UID.2) addresses T.UNAUTH by providing measures to uniquely identify and authenticate users through Username/Password.

Table 10-2: Threat to Objective Mapping

10.2 EAL 3 Justification

The threats that have been countered are consistent with an attacker of low attack potential, therefore EAL3 was chosen for this ST. The augmentation of ASE_TSS.2 was chosen to provide disclosure of architectural security typically reserved for ADV_ARC.1. The augmentation of ALC_FLR.1 was chosen to provide assurance that the TOE can receive security and functional patches based on defects which are discovered once deployed.

10.3 Strength of Function Rationale

A strength of function level of SOF-basic counters an attack level of low. The environment is one where the potential attacker is unsophisticated, with access to only standard equipment and public information about the product, consistent with a Common Criteria Level of Evaluation of EAL3. Specifically, AVA_VAN.2 requires that the TOE be provided to the evaluator for vulnerability testing, and to confirm that the information provided meets all requirements for content and presentation of evidence. Through this testing, the evaluator shall determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential. Consequently, the metrics (password) chosen for inclusion in this ST for this TOE were determined to be acceptable for SOF-basic and would adequately protect information in a Basic Robustness Environment. There is one security function (FIA_UAU.2) based on probabilistic methods. A policy for selecting a strong password for user authentication to meet this claim is described in the organization's security policy. The TOE can enforce this policy through the FIA_SOS.1 requirement.

10.4 Requirement Dependency Rationale

All Security Functional Requirement component dependencies have been met by the TOE as defined by the CEM with one exception. The sole deviation is the dependency of FAU_GEN.1 on FPT_STM.1 which is not satisfied because the TOE itself does not

provide reliable system time. It is assumed that the TOE relies on the Operational Environment to provide reliable time via the OE.SYSTEM objective.

10.5 Security Functional Requirements Rationale

The following table provides a mapping with rationale to identify the security functional requirement components that address the stated TOE and environment objectives.

Objective	Security Functional Components	Rationale
O.ACCESS The TOE will provide measures to authorize TOE users to access specified resources once the user has been authenticated. TOE user authorization is based on access rights configured by other TOE users with the ability to configure them.	FDP_ACC.1(1) Subset access control	FDP_ACC.1(1) states the TSF shall enforce the User Policy on all TOE users.
	FDP_ACC.1(2) Subset access control	FDP_ACC.1(2) states the TSF shall enforce the Workflow Policy on all TOE users.
	FDP_ACC.1(3) Subset access control	FDP_ACC.1(3) states the TSF shall enforce the Web Service Policy on all TOE users.
	FDP_ACF.1(1) Security attribute based access control	FDP_ACF.1(1) states the TSF shall enforce the User Policy based on task, role, and scope.
	FDP_ACF.1(2) Security attribute based access control	FDP_ACF.1(2) states the TSF shall enforce the Workflow Policy based on explicit authorization to perform workflow tasks.
	FDP_ACF.1(3) Security attribute based access control	FDP_ACF.1(3) states the TSF shall enforce the Web Service Policy based on explicit authorization to execute TEWS applications.
	FDP_IFC.1 Information Flow Control	FDP_IFC.1 states the TSF shall define an information flow policy for workflow.
	FDP_IFF.1 Simple Security Attributes	FDP_IFF.1 states the TSF shall enforce the workflow information flow based on delegation assignments.
	ADV_ARC.1 Security Architecture	ADV_ARC.1 states the TOE shall be designed and implemented in a manner such that access control mechanisms cannot be bypassed.
O.AUDIT The TOE will provide measures for recording security relevant events that will assist the authorized users in detecting misuse of the TOE and/or its security features that would compromise the integrity of the TOE and violate the security	FAU_GEN.1 Audit data generation	FAU_GEN.1 states that the TSF shall be able to generate an audit record for the start-up and shutdown of the audit functions and all auditable events for the level of audit. For each record, the TSF shall record the date/time/type/outcome of the event, subject identity, and all other values specified in Tables 6-3 and 6-4.

Objective	Security Functional Components	Rationale
objectives of the TOE.	FAU_GEN.2 User identity association	FAU_GEN.2 states the TSF shall be able to associate each auditable event with the identity of the user that caused the event.
O.AUTH The TOE will provide measures to uniquely identify all TOE users and will authenticate the claimed identity prior to granting a user access to the resources protected by the TOE.	FIA_UAU.2 Timing of Authentication	FIA_UAU.2 allows a user on the TOE to perform public tasks with a security question-based authentication mechanism, and that all other TSF-mediated actions are governed by a password-based authentication mechanism.
	FIA_UID.2 User Identification Before Any Action	FIA_UID.2 requires each TOE user to be successfully identified before performing any TSF-mediated actions on behalf of the user.
O.FILESYS The security features offered by the TOE protects the confidentiality of TOE data that is stored in the database.	FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 states the TSF shall generate 256-bit AES keys in accordance with FIPS PUB 197.
	FCS_CKM.4 Cryptographic key destruction	FCS_CKM.4 states the TSF shall destroy cryptographic keys with an overwrite method.
	FCS_COP.1 Cryptographic operation	FCS_COP.1 states the TSF shall perform encryption and decryption using 256-bit AES keys.
O.MANAGE The TOE will provide TOE users with the resources to manage and monitor user accounts, resources, and security information relative to the TOE.	FMT_MSA.1 Management of Security Attributes	FMT_MSA.1states the TSF shall enforce the User Policy to only allow management of tasks within the TOE to users who belong to the appropriate role and have the proper scope.
	FMT_MSA.3 Static Attribute Initialization	FMT_MSA.3 states the TSF shall enforce the User Policy to provide restrictive default values for security attributes that are used to enforce the SFP. It allows the authorized TOE users to override the default values set for security attributes when creating data for use by the TOE such as user accounts.

Objective	Security Functional Components	Rationale
	FMT_MTD.1(1) Management of TSF Data	FMT_MTD.1(1) states that the ability to query TSF data is restricted to a particular subset of TOE users.
	FMT_MTD.1(2) Management of TSF Data	FMT_MTD.1(2) states that the ability to modify TSF data is restricted to a particular subset of TOE users.
	FMT_MTD.1(3) Management of TSF Data	FMT_MTD.1(3) states that the ability to delete TSF data is restricted to a particular subset of TOE users.
	FMT_REV.1 Revocation	FMT_REV.1 states that the TOE shall enforce the revocation of access rights from a TOE user upon the establishment of their first session following the completion of the revocation.
	FMT_SMF.1 Specification of management functions	FMT_SMF.1 states that the TOE shall provide the management functions shown in Table 6-8.
	FMT_SMR.2 Security Roles	FMT_SMR.2 requires the TOE to provide the ability to set roles for security relevant authority as well as to restrict the ability to define and assign roles to authorized administrators.
O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	ALC_DEL.1 Delivery Procedures	ALC_DEL.1 describes product delivery and a description of all procedures used to ensure objectives are not compromised in the delivery process.
	AGD_PRE.1 Preparative Procedures	AGD_PRE.1 documents the procedures necessary and describes the steps required for the secure installation, generation, and start-up of the TOE.
	AGD_OPE.1 Operational user guidance	AGD_OPE.1 describes the proper use of the TOE from a TOE user standpoint.
O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.	FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 states the TSF shall generate 256-bit AES keys in accordance with FIPS PUB 197.
	FCS_CKM.4 Cryptographic key destruction	FCS_CKM.4 states the TSF shall destroy cryptographic keys with an overwrite method.
	FCS_COP.1 Cryptographic operation	FCS_COP.1 states the TSF shall perform encryption and decryption using 256-bit AES keys.
O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a TOE	FDP_ACC.1(1) Subset access control	FDP_ACC.1(1) states the TSF shall enforce the User Policy on all TOE users.

Objective	Security Functional Components	Rationale
user's logical access to the TOE and to explicitly deny access to specific TOE users when appropriate	FDP_ACC.1(2) Subset access control	FDP_ACC.1(2) states the TSF shall enforce the Workflow Policy on all TOE users.
	FDP_ACC.1(3) Subset access control	FDP_ACC.1(3) states the TSF shall enforce the Web Service Policy on all TOE users.
	FDP_ACF.1(1) Security attribute based access control	FDP_ACF.1(1) states the TSF shall enforce the User Policy based on task, role, and scope.
	FDP_ACF.1(2) Security attribute based access control	FDP_ACF.1(2) states the TSF shall enforce the Workflow Policy based on explicit authorization to perform workflow tasks.
	FDP_ACF.1(3) Security attribute based access control	FDP_ACF.1(3) states the TSF shall enforce the Web Service Policy based on explicit authorization to execute TEWS applications.
	FDP_IFC.1 Information Flow Control	FDP_IFC.1 states the TSF shall define an information flow policy for workflow.
	FDP_IFF.1 Simple Security Attributes	FDP_IFF.1 states the TSF shall enforce the workflow information flow based on delegation assignments.
	FIA_ATD.1 User Attribute Definition	FIA_ATD.1 defines the TSF-relevant security attributes for a TOE user as their username, password, assigned roles, and scope of membership for each role.
	FIA_SOS.1 Verification of Secrets	FIA_SOS.1 states that the TSF shall provide a mechanism to enforce Identity Manager password policies.
	FIA_UAU.2 Timing of Authentication	FIA_UAU.2 allows a user on the TOE to perform public tasks with a security question-based authentication mechanism, and that all other TSF-mediated actions are governed by a password-based authentication mechanism.
	FIA_UID.2 User Identification Before Any Action	FIA_UID.2 requires each TOE user to be successfully identified before performing any TSF-mediated actions on behalf of the user.

Table 10-3: Security Functional Requirements Rationale

11 Assurance Measures

This section identifies the assurance measures provided by the developer in order to meet the security assurance requirement components for EAL3 augmented with ALC.FLR.1. A description of each of the TOE assurance measures follows in Table 11-1.

Component	Document(s)	Rationale
ADV_ARC.1 Security Architecture Design	TOE Design Specification for Identity Manager r12.5 v1.2	This document describes the security architecture of the TOE.
ADV_FSP.3 Functional Specification with complete summary	Functional Specification Document for CA Identity Manager r12.5 v1.2	This document describes the functional specification of the TOE with complete summary.
ADV_TDS.2 Architectural Design	TOE Design Specification for Identity Manager r12.5 v1.2	This document describes the architectural design of the TOE.
AGD_OPE.1 Operational User Guidance	CA Identity Manager Administration Guide r12.5 CA Identity Manager Configuration Guide r12.5 CA Identity Manager Release Notes r12.5	These documents describe the operational user guidance for the TOE.
AGD_PRE.1 Preparative Procedures	CA Identity Manager Administration Guide r12.5 CA Identity Manager Standard Connector Guide 12.5 CA Identity Manager Provisioning Reference Guide CA Identity Manager Configuration Guide r12.5	These documents describe the preparative procedures that need to be done prior to installing the TOE.
ALC_CMC.3 Authorizations Controls	CM Plan for IM r12.5 CA Identity Manager Configuration Management for Common Criteria r12.5 Build and Release for IM r12 IM Source Code Management	These documents describe the authorization controls for the TOE.
ALC_CMS.3 CM Scope	imr12GA clearcase filelist imr12GA svn filelist imr 12-5 filelist	These documents show the CM scope of the TOE and the delta between the previous and current versions of it.

Component	Document(s)	Rationale
ALC_DEL.1 Delivery Procedures	Global Supply Chain and Logistics Overview Product Manufacturing QA Process Prototype Approval Form – North America Production Service Level Agreement (SLA) for New Products Identity Manager NIAP Installation Instructions	These documents describe product delivery for CA Identity Manager r12.5 and a description of all procedures used to ensure objectives are not compromised in the delivery process.
ALC_DVS.1 Identification of Security Measures	Backup Procedures Global Security Pre-Employment Screening Global Risk and Compliance Global Safety and Asset Protection Security Operations Access Procedure Records Security and Confidentiality Policy Records Disposal Procedure Privileged Access Control of Source Code and Design Documents Policy Enterprise Procedure – Privacy and Data Protection Inactive User Account Procedure Server Security Procedure Computer Usage Policy Employee Handbook 101600.html (evidence of defect tracking procedures) CALevel3Melbourne (floor plan of development site) "How security savvy are you?" (periodic security awareness email distributed to staff) IM Star defect example (evidence of defect tracking procedures) Melbourne CA Office (evidence of physical security controls at development site) SRF-CQ110163 (evidence of QA procedure implementation) Submission Request Form – Template (evidence of QA procedure definition) Onsite Assessment Report	These documents provide a description of the physical, procedural, and personnel security mechanisms implemented to ensure that the TOE's development environment is trusted.

Component	Document(s)	Rationale
ALC_FLR.1 Basic Flaw Remediation	Identity Manager L2 Sustaining Cycle Process	This document provides the policies for issuing new releases of the TOE as corrective actions.
ALC_LCD.1 Life-Cycle Definition	PRIME Reference Guide	This document provides the life-cycle definition of the TOE.
ASE_CCL.1 Conformance Claims	CA Identity Manager r12.5 Security Target v2.0	This document describes the CC conformance claims made by the TOE.
ASE_ECD.1 Extended Components Definition	CA Identity Manager r12.5 Security Target v2.0	This document provides a definition for all extended components in the TOE.
ASE_INT.1 Security Target Introduction	CA Identity Manager r12.5 Security Target v2.0	This document describes the Introduction of the Security Target.
ASE_OBJ.2 Security Objectives	CA Identity Manager r12.5 Security Target v2.0	This document describes all of the security objectives for the TOE.
ASE_REQ.2 Security Requirements	CA Identity Manager r12.5 Security Target v2.0	This document describes all of the security requirements for the TOE.
ASE_SPD.1 Security Problem Definition	CA Identity Manager r12.5 Security Target v2.0	This document describes the security problem definition of the Security Target.
ASE_TSS.2 TOE Summary Specification	CA Identity Manager r12.5 Security Target v2.0	This document describes the TSS section of the Security Target.
ATE_COV.2 Analysis of Coverage	Functional Specification Document for CA Identity Manager r12.5 v1.2 CA Identity Manager r12.5 Security Target v2.0 Identity Manager r12.5 Common Criteria Mapping List Core Test Cases Proofs	These documents provide rationale that tests were executed which cover the security claims made by the TOE and the relevant external interfaces which are used to accomplish the operations mapped to these claims.
ATE_DPT.1 Basic Design	TOE Design Specification for Identity Manager r12.5 v1.2 Identity Manager r12.5 Common Criteria Mapping List Core Test Cases Proofs	These documents provide evidence that tests were executed which exercise all subsystems of the TSF.
ATE_FUN.1 Functional Tests	Identity Manager r12.5 Common Criteria Mapping List Core Test Cases Proofs	This document describes the tests executed to demonstrate proper functionality of the TSF.

Component	Document(s)	Rationale
ATE_IND.2 Independent Testing	CA Identity Manager 12.5 Evaluation Team Test Report v1.0	This document describes the independent testing for the TOE.
AVA_VAN.2 Vulnerability Analysis	CA Identity Manager 12.5 Vulnerability Analysis Vulnerability Analysis v1.0	This document describes the vulnerability analysis of the TOE.

Table 11-1: Assurance Requirements Evidence