Security Target for
Oracle Application Server 10*g* (9.0.4)

May 2006

Security Evaluations
Oracle Corporation
500 Oracle Parkway
Redwood Shores, CA 94065

Security Target for Oracle Application Server 10*g* (9.0.4)

May 2006

Authors: Saad Syed and Peter Goatly.

Contributors: Shaun Lee and Julian Skinner.

Contents

CHAPTER

1 Introduction

This document is the security target for the Common Criteria evaluation of Oracle Application Server 10*g* (9.0.4). For this evaluation, the Oracle Application Server products which are in the Target of Evaluation are Oracle Application Server Containers for J2EE (OC4J) and Oracle Internet Directory (OID).

Identification and CC Conformance

Title: Security Target for Oracle Application Server 10*g* (9.0.4)

Target of Evaluation (TOE): Oracle Application Server Containers for J2EE and Oracle Internet Directory

Release:
Oracle Application Server Containers for J2EE:
10*g* (9.0.4.0.0) with Oracle Application Server 10*g* Patch Set 1 - July 2004,
which are together referred to as Oracle Application Server Containers for J2EE 10*g* (9.0.4.1.0) ;
Oracle Internet Directory:
10*g* (9.0.4.0.0).

Operating System Platform:

Sun SPARC Solaris 8 2/02
for which [CRP182] is the Common Criteria certification report.

Database Platform: Oracle9*i* Release 2 (9.2.0.1.0)
for which [CRP178] is the Common Criteria certification report.

CC Conformance: CC Part 2 Extended.

This Security Target conforms to [CC, Part 2] and [CC, Part 3], and all SFRs in the Security Target are derived from [CC]. The exceptions to this are that the TOE SFRs, FAU_GEN.1 and FPT_SEP.1 have been extended and there are two related extensions in the IT environment SFRs. ALC_FLR.3 is the only augmented assurance criterion specified.

Assurance: EAL4 augmented with ALC_FLR.3[1].

Keywords: Oracle Application Server, Oracle Application Server Containers for J2EE, OC4J, Oracle Internet Directory, OID, security target, EAL4

Version of the Common Criteria [CC] used to produce this document: 2.2 with amendments introduced by the CC Interpretations effective on 31st August 2004.

TOE Overview

The TOE consists of Oracle Application Server Containers for J2EE (OC4J) together with Oracle Internet Directory (OID), which is used to manage OC4J's user repository data.

Oracle Application Server Containers for J2EE is a Java 2 Enterprise Edition (J2EE) environment written entirely in Java that executes on the Java Virtual Machine. OC4J provides all the containers, APIs, and services that J2EE specifies.

Oracle Internet Directory is a general purpose directory service that enables fast retrieval and centralised management of information about users and network resources.

The security functionality provided by Oracle Application Server Containers for J2EE together with Oracle Internet Directory includes:

- user identification and authentication, with password management;
- application access control - which permits users to access applications hosted by OC4J if they have sufficient authorization;
- security attribute maintenance - which provides the means for creating and maintaining the security attributes for TOE users and user repository entries;
- user repository access controls - which use Access Control Items held in the directory to define users' authorizations for user repository data access; and
- auditing.

TOE Product Components

The Oracle Application Server products which constitute the TOE are Oracle Application Server Containers for J2EE 10*g* (9.0.4.1.0) and Oracle Internet Directory 10*g* (9.0.4.0.0).

Oracle Application Server Containers for J2EE 10*g* (9.0.4.1.0) relies on the Java Naming and Directory Interface 1.2.1 for communication with Oracle Internet Directory 10*g* (9.0.4.0.0). The Java Naming and Directory Interface (JNDI) is part of the Java platform provided by Sun Microsystems, Inc.

Oracle Process Manager and Notification Server (OPMN) is installed and configured with every Oracle Application Server installation type and is used to start, monitor and stop OC4J's processes in the TOE's evaluated configuration.

1. ALC_FLR.3 provides assurance at the highest defined component level that there are flaw remediation procedures for the TOE by which discovered security flaws can be reported to, tracked and corrected by the developer, and by which corrective actions can be issued to TOE users in a timely fashion.

Oracle Internet Directory relies on the Oracle9*i* Database Server Enterprise Edition 9.2.0 for the storage of directory data and uses Oracle Net Services 9.2.0 for communication interfaces.

[ECD] defines how the TOE products must be installed in the evaluated configuration and defines the requirements for setting up the TOE environment.

Document Overview

Chapter 2 of this security target provides a high-level overview of the security features of the TOE. Chapter 3 identifies the assumptions, threats, and security policies of the TOE environment. Chapter 4 describes the security objectives for the TOE and for the environment needed to address the assumptions, threats, and security policies identified in Chapter 3. Chapter 5 identifies the Security Functional Requirements (SFRs), the Security Assurance Requirements (SARs) and the security requirements for the IT environment. Chapter 6 summarises each Security Function (SF) provided by the TOE to meet the security requirements. Chapter 7 covers the topic of protection profile conformance by the TOE and Chapter 8 provides the rationale for the security claims made within this security target.

Annex A contains a list of references and Annex B provides a glossary of the terms.

Changes Bars indicate changes made since the previous issue.

This Page Intentionally Blank

CHAPTER

2 TOE Description

This chapter describes the product features that provide security mechanisms and contribute to the security of a system using the TOE. For this evaluation, the Oracle Application Server products which constitute the TOE are Oracle Application Server Containers for J2EE (OC4J) and Oracle Internet Directory (OID). For a detailed description of the security features of Oracle Application Server Containers for J2EE, the reader is referred to [OC4JSG, 6] and for Oracle Internet Directory to [OIDAG, 12] and [OIDAG, 10: Using the Audit Log]. In general, these descriptions correspond to the specifications of IT security functions provided in Chapter 6 of this Security Target.

The major elements of the OC4J and OID security architecture are described below, and the TOE is defined in terms of this architecture. The TOE's mechanisms for access control, identification and authentication, and accountability and auditing are summarised. Additional OC4J and OID security features that are not addressed by the security functional requirements of Chapter 5 are also briefly discussed. Finally, the main Oracle Application Server products not addressed by this evaluation are summarised.

OC4J and OID Architecture

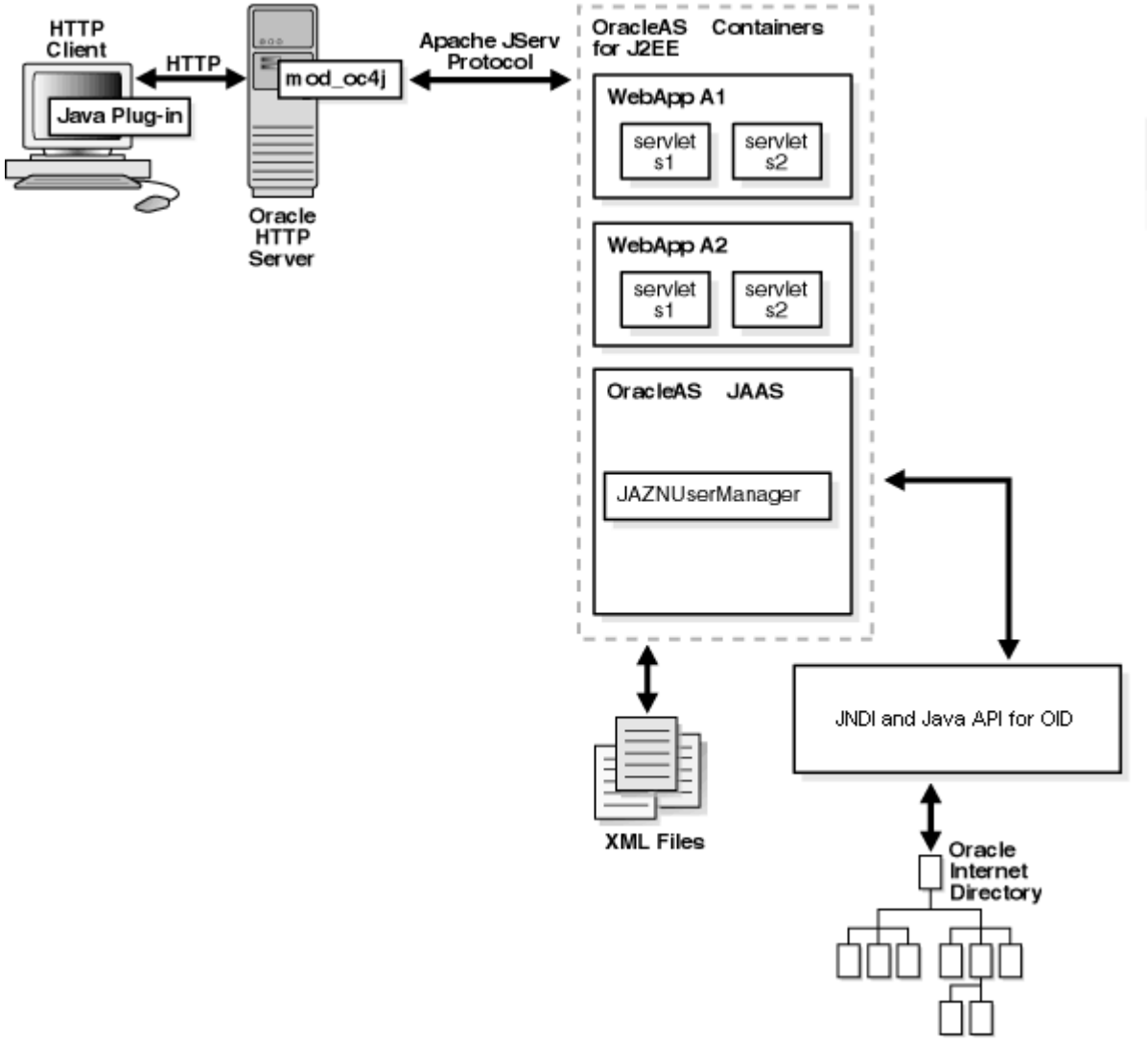
The overall architecture for Oracle Application Server is described in [OASC, 1: Overview of Oracle Application Server], which also lists its constituent components. Figure 1 displays the architectural components for OC4J at run-time.

The Oracle Application Server Containers for J2EE architecture is described in detail in [OASC, 2]. The Oracle Internet Directory architectural components are described in detail in [OIDAG, 2].

The figure below illustrates the TOE configuration involved in the authentication and authorization processes. In the diagram, the user has clicked on a link in a web page which requires the use of a web application that is hosted by OC4J. The user's browser communicates with Oracle HTTP Server, which passes the application access request to OC4J via the `mod_oc4j` module. Via the Apache JServ Protocol, OC4J requests the browser to obtain a username and password from the user and uses Oracle Internet

Directory to access the user repository to check the password. If user authentication is successful, OC4J checks the user repository and the XML configuration files to see if the user has permission to use the application.

Figure 1: Authentication and Authorization



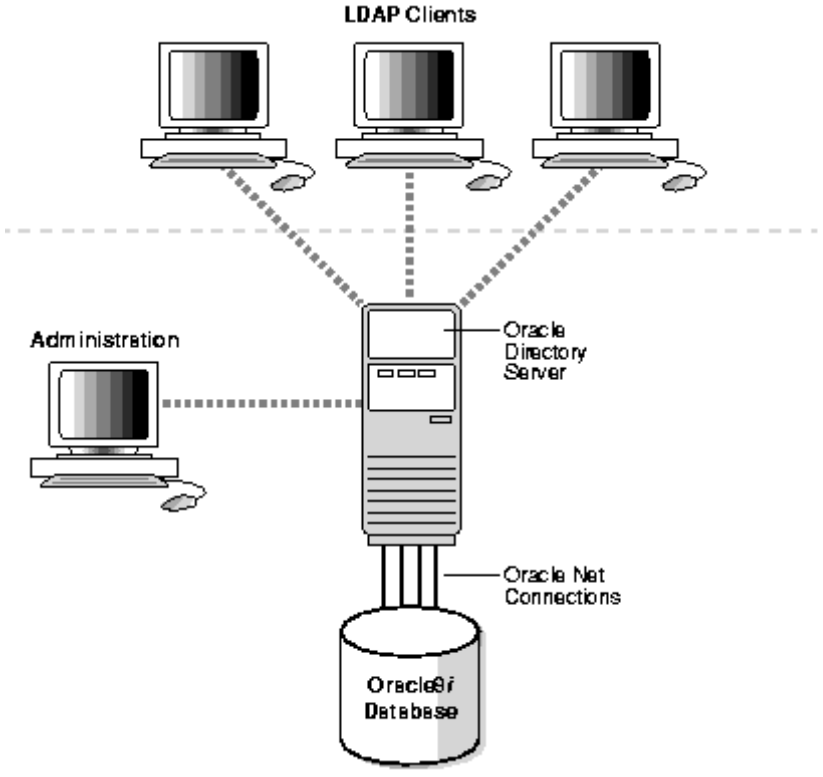
The TOE software components used for authentication and authorization are:

- `JAZNUserManager`;
- `RealmLoginModule`;
- Java Naming and Directory Interface (JNDI);
- Java API for OID; and
- Oracle Directory Server.

These components are described later in this chapter.

The next figure illustrates the TOE configuration for user repository administration. Clients using the LDAP protocol and administrators can connect to the Oracle Directory Server to access the user repository. OID holds the user repository data in an Oracle9*i* database, which it connects to via Oracle Net Services.

Figure 2: User Repository Administration



The TOE components used for user repository administration are:

- Oracle Directory Server;
- OID Monitor (OIDMON);
- OID Control Utility (OIDCTL); and
- OID command-line administration tools.

OC4J

OC4J is a J2EE 1.3 certified server implementation that is written entirely in Java and executes on the Java Virtual Machine (JVM). OC4J supports the Java Authentication and Authorization Service (JAAS) and is Oracle Application Server's JAAS Provider. The JAAS reduces development costs by allowing developers to use a declarative security model instead of having to integrate security programmatically.

OC4J can perform two sorts of authorization checks: J2EE authorization and JAAS authorization. J2EE authorization concerns a user's permission to access a J2EE application. JAAS authorization concerns a user's permission to perform an action on

a resource after the J2EE application has been entered. Only J2EE authorization is within the scope of this evaluation of Oracle Application Server.

Oracle Process Manager and Notification Server (OPMN) is installed and configured with every Oracle Application Server installation type and is used to start, monitor and stop OC4J's processes in the TOE's evaluated configuration.

Java 2 Platform Enterprise Edition (J2EE)

J2EE is a platform-independent, Java-centric environment for developing, building and deploying Web-based enterprise applications online. The J2EE platform consists of a set of services, APIs and protocols that provide the functionality for developing multitiered Web-based applications.

JAAS

JAAS is a Java package that enables applications to authenticate and enforce access controls on users. The JAAS framework and the Java 2 Security model form the foundation of JAAS. JAAS implements a Java version of the standard Pluggable Authentication Module (PAM) framework. This enables an application to remain independent from the authentication service.

JAAS Provider

A JAAS Provider is an implementation of the JAAS interface. OC4J is Oracle's JAAS Provider.

User Repository

OC4J's user credentials are stored in a *user repository*. The OC4J JAAS implementation supports two different provider types. Each provider type implements a user repository for secure, centralized storage, retrieval, and administration of provider data. This data consists of realm (user and roles or groups) and JAAS policy (permissions) information. The XML-based provider stores its provider information in an XML file. The LDAP-based provider is based on the Lightweight Directory Access Protocol (LDAP) for centralized storage of information in a directory. The LDAP-based provider is used in the TOE's evaluated configuration. It stores its provider data in a directory which is accessed via the Oracle Internet Directory product.

User Manager

OC4J employs a *user manager* for use in authenticating and authorizing users that attempt to access a J2EE application. OC4J provides two predefined user managers, `JAZNUserManager` and `XMLUserManager`. `JAZNUserManager` is the user manager which is used in the TOE's evaluated configuration.

Basic authentication is the authentication method for the TOE's evaluated configuration. When Basic authentication is configured, `JAZNUserManager` invokes `RealmLoginModule` to authenticate the user, using credentials supplied by the user via the browser over HTTP.

`JAZNUserManager` uses the Java Naming and Directory Interface (JNDI) for communication with Oracle Internet Directory to access the user repository. It also uses OID's Java Application Programming Interface for some OID-specific aspects of this access.

mod_oc4j

When a user requests access to an application hosted by OC4J, `mod_oc4j`, a web server module, routes the request from the Oracle HTTP Server to OC4J via the Apache JServ protocol.

Oracle Internet Directory

Oracle Internet Directory (OID) is a general purpose directory service that enables

fast retrieval and centralised management of information about dispersed users and network resources. It combines LDAP V3 with the high performance, scalability, robustness, and availability of the Oracle9*i* database server. The Oracle Internet Directory runs as an application on Oracle9*i* and uses its Oracle9*i* database to hold the directory data.

OID is the software used by the TOE to access user repository data (which is held in OID's directory).

LDAP

The *Lightweight Directory Access Protocol* (LDAP) is a standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate. LDAP Version 3 is defined in [LDAP3].

Directory

A *directory* stores and retrieves information about organisations, individuals and other resources. It acts as the user repository for OC4J in the configuration for this evaluation.

Directory Entries

In a directory, a collection of information about an object is called an *entry*. Each entry is uniquely identified by a *distinguished name* (DN), which defines exactly where in the directory's hierarchy the entry resides.

Each entry contains information stored in *attributes*. An *object class* is a group of attributes that define the structure of an entry.

Each directory has a *Directory-Specific Entry* (DSE), which holds information that relates to the whole directory, such as the audit log.

Oracle Directory Server Instance

Each *Oracle Directory Server instance* services directory requests through a single OID dispatcher process listening at a specific TCP/IP port number. There can be more than one directory server instance on a node, each listening on a different port.

One instance comprises one dispatcher process and one or more server processes. By default there is one server process for each instance.

Oracle9*i* Database

OID runs as an Oracle9*i* application. An Oracle9*i* database stores the directory data. The database can reside on the same node as the directory server processes or on a separate node.

Oracle Net Connections

OID communicates with the database using Oracle Net Services, Oracle's operating system-independent database connectivity solution. Oracle Net Services is used for all connections between the Oracle Database Server and the OID Control utility (`oidctl`), the directory server instance, and the OID Monitor (`oidmon`).

LDAP Clients

LDAP Clients send LDAP requests to an OID listener/dispatcher process listening for LDAP commands at its port.

The following command-line tools can be used to send LDAP commands to the Oracle Directory Server which is in the TOE's evaluated configuration:

`ldapadd, ldapaddmt, ldapbind, ldapcompare, ldapdelete, ldapmoddn, ldapmodify, ldapmodifymt, and ldapsearch.`

TOE Definition

For this evaluation of Oracle Application Server 10*g*, the products which constitute the TOE are OC4J and OID.

The evaluated configuration for OC4J includes JAZNUserManager as the user manager, and uses the Basic authentication method when users request access to applications hosted by OC4J. OID is used by TOE administrators to create and manage OC4J's user repository data, which is held in an OID directory. JAZNUserManager uses the Java Naming and Directory Interface (JNDI) and OID's Java Application Programming Interface for communication with Oracle Internet Directory. Oracle Process Manager and Notification Server (OPMN) is installed and configured with every Oracle Application Server installation type and is used to start, monitor and stop OC4J's processes in the TOE's evaluated configuration.

The evaluated configuration for OID comprises the Oracle Directory Server and the command-line directory administration tools that are essential for the directory to be maintained and administered securely. These command-line tools are the tools to run the Oracle Directory Server instances (`oidmon` and `oidctl`), the Catalog Management Tool (`catalog`), the Bulk Operations Tools (`bulkload` etc.), the OID Database Password Utility (`oidpasswd`), and the OID Database Statistics Collection Tool (`oidstats`).

Identification and Authentication

User Representation

In the directory holding the user repository data, the attributes a TOE user entry can have are listed in [OIDAG, Appendix B: Schema to Represent a User]. Some of the attributes relevant to identification and authentication include:

- `cn` - the user's common name; and
- `UserPassword` - the password to be used for authenticating the user to the TOE.

User Identification

Each TOE user entry is uniquely identified by its distinguished name (DN) attribute. The distinguished name indicates exactly where the entry resides in the directory hierarchy (represented by the *Directory Information Tree* (DIT)). A DN for a user could look like this:

```
cn=Alice Smith,ou=Server Technologies,c=UK,o=oracle
```

Within a distinguished name, the lowest component is called the *relative distinguished name* (RDN). In the example above the RDN is `cn=Alice Smith`.

To uniquely identify a particular entry within the overall DIT, the full DN must be used. This allows for user entries for two different Alice Smiths to exist within the same DIT.

super user

The *super user* is the administrator for the directory holding the TOE's user repository data, and has full access to all directory information (see [OIDAG, 5: Managing Super Users, Guest Users, and Proxy Users]). The actual name and the password for the super user are held in the DSE (by default the super user's name is `orcladmin`).

User Authentication

Authentication is the process by which the TOE validates the true identity of a user making a connection. Note that the authentication methods used for the TOE, which are described below, result in passwords being sent in clear text over the network. To ensure that no security issues arise from this, [ECD] imposes appropriate constraints on the TOE's networking environment.

Authentication for Application Access

Basic authentication is the authentication method employed when a user requests access to an application hosted by OC4J in the TOE's evaluated configuration. For this authentication method, `JAZNUserManager` invokes `RealmLoginModule` to authenticate the user, using the username and password supplied by the user via the browser over HTTP. OID is used to access the user repository data to check the username is valid and that the password supplied by the user matches the password in the user's entry.

Authentication for User Repository Access

OID implements four different levels of authentication when a connection is requested to the user repository's directory server:

- Anonymous;
- Password-based (Simple Authentication);
- Certificate-based through Secure Socket Layer (SSL); and
- Indirect Authentication.

For anonymous authentication, the directory user binds to the directory server without specifying a user name or password.

For Password-based (simple) authentication, a user must specify a user name (DN) and password in order to connect to the directory. The password is compared to the password for the user stored in the directory and, if they match, a directory session is created.

OID stores a user's directory password in the `UserPassword` attribute. Passwords are stored as one-way hashed values. During authentication the client provides a password to the directory server in clear text. The directory server hashes this value using the algorithm specified in the `orclCryptoScheme` attribute within the DSE (this is set up during installation). If the hashed password values match, the server authenticates the user.

For simple authentication passwords are not encrypted when sent over the network.

SSL based authentication involves the exchange of certificates issued by trusted certificate authorities. This method of authentication is not permitted in the evaluated configuration.

Indirect authentication can occur through any entity that has credentials in the directory, for example the Delegated Administration Service, or through a middle tier such as a firewall or an application. This type of authentication involves the entity connecting to the directory and then performing directory operations on an end user's behalf. This is only allowed if the access control policy governing the end user's directory entry allows the entity to act as a proxy for the end user.

Password Policies

Password policies enable an administrator to establish and enforce rules for how passwords are created and changed via Oracle Internet Directory and are used by the

TOE. When a user attempts to connect to the TOE, the TOE will not accept the password as valid unless it meets the requirements specified in the password policy.

OID provides a password policy which includes, but is not limited to, the following:

- the maximum length of time a given password is valid;
- whether a user's old password value can be used as the new one when the password is being changed;
- the minimum number of characters a password must contain;
- the number of numeric characters required in a password; and
- the number of consecutive failed userpassword checks after which a user's account is locked.

The OID password policy is only applicable to the `UserPassword` attribute, except that the super user is also subject to account lockout after the specified number of consecutive failed password checks.

The command-line tool `ldapmodify` may be used to modify the password policy.

More information on password policies may be found in [OIDAG, 15] and [OIDAG, Appendix C: Password Policy Fields in Oracle Directory Manager].

By default the directory enforces the password profile limits as specified in [OIDAG, 15], however, in the evaluated configuration it is necessary that more restrictive password controls are used so that the TOE achieves a *high* strength of function for the password mechanism (see the Minimum Strength of Function section in Chapter 5).

Guidance covering the different password controls, and instructions for modifying profiles to achieve SOF-*high*, are provided in the TOE's Evaluated Configuration Document [ECD].

Application Access Control

The TOE's application access controls use the identity of the user in order to determine the user's rights for access to particular applications hosted by OC4J.

The user repository and the TOE's XML configuration files can contain information about permissions for users to access applications. Such information includes user membership of groups which are mapped via configuration files to security roles that have access permissions for applications. [OC4JSG, 6: J2EE and JAAS Provider Role Mapping] describes how configuration files can be used to associate OID groups with security roles that have access permissions for applications. [OC4JUG, A] and [WebXML] give a full definition of the XML configuration files that are used by OC4J.

An authenticated TOE user is only granted a session with a Web application hosted by the TOE if the user is a member of a group defined in the user repository which is mapped via TOE configuration files to a security role that has permission to access the application.

Security Attribute Maintenance

These features provide the means for creating and maintaining the security attributes for TOE users. The mechanisms by which changes to security attributes become effective for a user repository session are also included.

User Repository Access Control

OID's access controls use the identity of the user in order to determine the user's rights for access to objects held in the directory.

Access Control Lists

The OID directory holds access control information to define the administrative policies relating to access control. This information is stored as user-modifiable operational attributes called *access control items* (ACIs). A list of such ACI attributes is called an *Access Control List* (ACL).

Access Control Lists (ACLs) are used to protect the directory information within OID and specify what actions can be carried out by which entities on a given resource. They ensure that a user only reads or updates the information for which they have the appropriate privileges.

When an attempt is made to perform an operation on an object during a session, the directory server is responsible for making sure that the user has the requisite permission to carry out this operation. If not, then the directory server disallows the operation.

ACI components

Access control information represents the permissions that various entries or subjects have to perform operations on a given object in the directory. An Access Control Item is comprised of 3 *ACI components*:

- The object to which access may be granted (an entry or attribute);
- The entities or subjects to which access may be granted; and
- The kind of access that may be granted (as listed in [OIDAG, 14: Operations: What Access Are You Granting?]).

Access Control Policy Points

An *Access Control Policy Point* (ACP) is an entry for which the `orclACI` attribute has been given a value. The `orclACI` attribute values are inherited by a subtree of entries starting with the ACP at the root of the subtree.

When a hierarchy of multiple ACPs exist in a directory subtree, a subordinate entry in that subtree inherits the access policies from all of the superior ACPs. The result is an aggregate of all the policies above the entry (where the root of a subtree is considered to be the highest entry in the subtree).

The `orclACI` attribute contains ACL directives that are prescriptive. That is, these directives apply to all entries in the subtree below the ACP where this attribute is defined.

Entry Level Access Control

The `orclEntryLevelACI` attribute is used for *entry level access control*. That is, when a policy pertains only to a specific entity. The `orclEntryLevelACI` attribute contains ACL directives that apply to only the entry with which it is associated.

Default Access Policies

The *default access control policy* grants the following to both entries and attributes:

- Everyone is given access to read, search, write, and compare all attributes in an entry, and selfwrite permissions are unspecified; and
- If permission to access an entry is not specified, access is determined at the next highest level at which access is specified.

How ACL evaluation works

When a user tries to perform an operation on a given object, the directory server determines whether the user has the appropriate access permission to perform that operation. If the object is an entry, it evaluates the access systematically for the entry and each of its attributes.

Evaluating access to an object can involve examining all the ACI directives for that object.

The directory server first examines the ACI directives in the entry-level ACI, `orclEntryLevelACI`. It proceeds to the nearest ACP, then considers each superior ACP in succession until the evaluation is complete.

The exact steps taken during the evaluation of ACLs, the precedence rules used, and the exceptions to these rules are described in [OIDAG, 14: How ACL Evaluation Works].

Access Control Groups

A group entry in OID contains a list of names. It is associated with either the `groupOfNames` or `groupOfUniqueNames` object class, which has the object class `orclPrivilegeGroup` as a subclass.

Membership in the group is determined by adding DNs to the multi-valued attribute `member` if the entry belongs to the `groupOfNames` object class, or `uniqueMember` if the entry belongs to the `groupOfUniqueNames` object class.

There are two types of access control groups: ACP groups and privilege groups.

If an individual is a member of an *ACP group*, then the directory server simply grants to that individual the privileges associated with that ACP group. ACP groups are associated with the `orclACPgroup` object class.

A *Privilege Group* is a higher-level access group. This is similar to an ACP group, but it also provides for additional checking beyond a single ACP. If an ACP denies access, an attribute in the user's entry tells the directory server whether the user being denied is in any privilege group. If so, then this user has additional rights at a higher administration level, and all higher administration levels in the DIT are checked. If the directory finds a higher ACP that grants to the privilege group access to the requested object, then it overrides the denials by the subordinate ACP, and grants access to the user.

Privilege groups should only be used when access control at higher levels need the right to override standard controls at lower levels.

Privilege Groups are associated with the `orclPrivilegeGroup` object class.

If a user is a member of both an ACP and a Privilege group, then OID evaluates each type of group. It resolves access rights for the privilege group by looking into ACPs higher in the DIT.

Managing Access Control

LDAP commands can be used to read and modify access control information. Oracle

Directory Manager or the command-line tools can be used to issue such commands. These tools are outside the scope of this evaluation.

Knowledge References

Knowledge references (or *referrals*) allow directory servers to return references to other servers as a result of a directory query (as described in [OIDAG, 2: Knowledge References and Referrals]).

There are two kinds of referrals:

Smart referral - these are returned to the client when the knowledge reference entry is within the scope of the search. A smart referral points the client to the server that stores the requested information.

Default referral - these are returned when the base object is not in the directory, and the operation is performed in a naming context not held locally by the server (because it is on another server). A default referral typically sends the clients to a server that has more knowledge about the directory partitioning arrangement.

Knowledge references are represented in the directory as a particular type of entry. They must be associated with the `referral` object class and the `extensibleObject` object class.

A knowledge reference provides users with a referral containing an LDAP URL. Such URLs are entered as values for the `ref` attribute. There can be multiple `ref` attributes for any knowledge reference entry. Similarly, there can be multiple knowledge reference entries in the DIT.

Java API for OID

The Oracle Internet Directory Java API is included in the scope of this evaluation, but only in respect of its use by OC4J when accessing the OID directory.

Auditing

Oracle Internet Directory ensures that relevant information about security-related operations performed by users on the directory can be recorded so that the consequences of these operations can later be linked to the user in question, and the user held accountable for his or her actions. OID does this by providing auditing options to ensure that exactly what needs to be audited, as dictated by the application or system security policy, is recorded, but nothing more. This helps to ensure that the size of audit trails remain manageable and that the important records are easily accessible.

As the audit log generation is contingent upon events occurring on the server, only the OID directory server itself can create the log entries. In particular, it is not possible to add audit log entries using Oracle Directory Manager or the command-line tools.

The *audit log* is made up of directory entries, where each entry records the audit data for one event.

No audit log entries are recorded during the process of authentication and authorization for a user to access an application hosted by OC4J. However, the risk of a successful attack against these mechanisms is minimal. This is because, in the TOE's evaluated configuration, the password policy results in user accounts being locked out if they come under a sustained attack, and action from the directory administrator, including investigation of the attacks, will be needed if the accounts are to be re-enabled. In addition, the authentication data and the permission information granting

authorization for a user to access an application hosted by OC4J are held in directory objects and TOE configuration files. These objects and files are fully protected by the User Repository Access Control mechanism, which is audited, and by the operating systems's Discretionary Access Control mechanism, which is also audited. To hold a user accountable for their actions in accessing an application hosted by OC4J, the hosted application can exploit the user identifier the TOE provides to it when the application session is started. [ECD] instructs administrators to ensure that such hosted applications log the identity of the user, so that this person can be held accountable for their use of the application.

Audit Level

To enable auditing, the attribute `orclauditlevel` in the DSE must be modified to the appropriate level. The value held in this attribute is called the directory's *audit level.* A value of 0 for the audit level indicates that no audit log entries are to be generated. By default `orclauditlevel` has a value of 0, but [ECD] instructs administrators that this value must not be used when the TOE is in its evaluated configuration.

Auditable Events

A directory super user can request auditing of one or more actions by *event type:*

- Super user login;
- Schema element add/replace;
- Schema element delete;
- Unsuccessful bind;
- Access violation;
- Directory-specific entry (DSE) modification;
- Replication login;
- ACL modification;
- User Password modification;
- Add;
- Delete;
- Modify;
- ModifyDN; and
- User login.

Both successful and unsuccessful events are recorded in the audit log if they are selected, except:

- Bind - only unsuccessful binds are audited; and
- Access Violation - which is only concerned with attempted access that has been denied by an access control policy.

The events from the above list which are to be audited are indicated via settings in the DSE attribute `orclauditlevel`.

Audit Records

Oracle Internet Directory auditing results in audit information being written to a directory audit trail. An audit trail entry always includes the following elements when they

are meaningful for the audited event:

- `orclsequence` (used to create the name of the audit entry);
- `orcleventtype` (the type of event that occurred);
- `orcleventtime` (the time at which the event occurred in coordinated universal time);
- `orcluserdn` (the identity of the user who caused the directory server to perform the operation);
- `orclopresult` (the outcome of the operation);
- `orclauditmessage` (a textual message); and
- `objectclass` (contains the preset values `top` and `orclauditoc`).

Audit Analysis

Audit log entries may be searched using `ldapsearch` commands or Oracle Directory Manager. However, audit log entries do not automatically become part of a search result even though the search filter may satisfy the query criteria. Only a search with `cn=auditlog` as the base of the search will retrieve audit log entries.

Purging the Audit Log

`bulkdelete` may be used to purge audit log objects under the container `cn=auditlog`. In order to run the `bulkdelete` tool, the correct password for OID's database user must be entered. This ensures that only users who are suitably authorised administrators can use this tool.

Other OC4J and OID Security Features

The features described below are **not** part of the evaluated configuration defined in [ECD]. This information is provided to help clarify which aspects of OC4J and OID are covered in the evaluation and which are out of scope.

Note that some of the features mentioned below are related to security and provide significant security capabilities to support robust and reliable application systems. However, they do not directly address any of the functional requirements identified in this Security Target.

Other OC4J Features

Table 2-3 in [OASC, 2: Oracle Application Server Containers for J2EE] lists the J2EE APIs supported by OC4J. Only the Java Authentication and Authorization Service (JAAS) API is covered by this evaluation. Thus, for example, the following OC4J features are outside of this evaluation's scope:

- Java Message Service (This feature provides a set of interfaces and associated semantics for Java programs to access messaging products);
- Remote Method Invocation (The remote procedure call paradigm);
- Data Sources (An instantiation of an object that implements the javax.sql.DataSource interface which allows a connection to a database server to be retrieved);
- Java Transaction API (Used by Enterprise JavaBeans for managing transactions involving single-phase and two-phase commits);
- J2EE Connector Architecture (A standard architecture for connecting the J2EE platform to Enterprise Information Systems); and

- Java Object Cache (Set of Java classes that manage objects for improved performance).

In addition, the following features of OC4J are outside of the scope of this evaluation:

- The XML-based JAAS Provider (The TOE's user repository data is held in an LDAP directory for this evaluation);
- `XMLUserManager` (`JAZNUserManager` is the user manager which is used in the TOE's evaluated configuration);
- Authentication other than via Basic authentication (The OracleAS Single Sign-On and Secure Sockets Layer authentication environments are not in the evaluated configuration); and
- JAZN Admintool (This tool is needed when maintaining the data used to control the JAAS authorization process, but JAAS authorization is outside the scope of this evaluation).

Other OID Components

The following OID components are outside of the scope of this evaluation:

- The Directory Replication Service (Replicates LDAP data between Oracle directory servers);
- Directory Integration Platform (This feature allows connectivity and synchronisation with other applications and directories, both Oracle-built and otherwise); and
- Server Side plug-in framework (This feature enables OID applications to make use of advanced capabilities such as referential integrity / cascading deletions of LDAP objects, external authentication of directory clients, brokered access, and synchronisation with external relational tables).

Directory Administration Tools

The following Directory Administration Tools are outside of the scope of this evaluation:

- Oracle Directory Manager (OID's standalone, 100% Java on-line administration tool);
- The Directory Replication Service Tools (Two tools exist to help administer the directory replication service: the OID reconciliation tool and the human intervention queue manipulation tool);
- The Delegated Administration Service (This allows delegated administrators, such as non-technical managers, to create and manage both users and groups. It also allows end users to modify and manage their own passwords without needing to know how to run a command-line tool); and
- Enterprise Manager Integration (used to start, stop, and monitor OID instances).

In addition, the command-line tools which can be used to send LDAP messages to a host Directory Server are not in the TOE's evaluated configuration. It is the Directory Server, which receives and acts upon those messages, that is the subject of this evaluation.

SSL

OID can make sure that no data has been modified, deleted, replayed, or disclosed to unauthorised parties during transmission through the use of Secure Sockets Layer (SSL). The use of SSL by OID will not be part of the configuration for this evaluation

since it is assumed that the Directory Server and the Clients used to access it are all within a secure network.

Enterprise Users

Enterprise users can be managed via entries in a directory and can be given access to multiple schemas and databases without having to create an account or schema in each database. The standard OID security functions for access control, identification and authentication, and auditing are relied on by Oracle products that implement enterprise user facilities. However, these enterprise user facilities are not part of the OID TOE and are therefore outside the scope of this evaluation.

guest user and proxy user

The *guest user* and the *proxy user* are special users (like the super user). In the evaluated configuration defined in [ECD], only the directory administrator is permitted to know the passwords for these users (which are held in the Directory-Specific Entry). This restriction is placed to prevent further unidentified users gaining access to the directory, in addition to those using anonymous authentication.

Please note that the proxy user is a particular special user. This concept is not connected with the concept of a directory session in which a user can act as a proxy for another user if the access control policy governing the second user's directory entry permits it.

APIs

The Oracle Internet Directory C API and PL/SQL API are outside the scope of this evaluation, which focuses on the LDAP service provided by the Directory Server. The Oracle Internet Directory Java API is included in the scope of this evaluation, because OC4J uses this interface when accessing the OID directory.

Configuration Tools

Oracle Application Server provides tools for configuring the TOE for use, but which are not part of the TOE for this evaluation. These include the tools listed in the "Directory Administration Tools" section above.

The tools listed below can be used for setting up the TOE's XML configuration files before the TOE is used. [OC4JSG, 3: Configuring and Deploying the JAAS Provider] describes how to perform the configuration tasks to use the JAAS Provider in a J2EE environment under OC4J. [OC4JUG, 2] and [OC4JUG, 3] describe how to configure OC4J and deploy applications to be hosted by it. [OC4JUG, A] and [WebXML] give a full definition of the XML configuration files that are used by OC4J. Such XML configuration files can be maintained by the use of file editors, provided that their contents conform to the definitions given in [OC4JUG, A] and [WebXML].

The configuration tools are:

- Enterprise Manager (This tool can be used as described in [OC4JSG, 8], [OC4JUG, 2] and [OC4JUG, 3], although the Delegated Administration Service, which is one of the tools listed in the "Directory Administration Tools" section above, should be used rather than Enterprise Manager for managing user and group entries in the directory.); and
- Distributed Configuration Management (This tool can be used as described in [OC4JSG, 8], [OC4JUG, 2] and [OC4JUG, 3].).

Other Oracle Application Server Products

[OASC, 1: Oracle Application Server Components] lists the products which are components of Oracle Application Server. The TOE only includes OC4J and OID for this

evaluation of Oracle Application Server. The main security-related products which lie outside the boundary of the TOE for this evaluation are described below.

Oracle HTTP Server

Oracle HTTP Server (OHS) makes data available to users through a standard Web interface. OHS mediates user access to both static and dynamic content by restricting access to URLs and directories on the server.

Oracle Application Server Single Sign-On

Oracle Application Server Single Sign-On enables users to access multiple Oracle Application Server applications with a single password. Using Single Sign-On, users can login to Oracle Application Server and gain access to all applications for which they are authorized, without requiring them to re-enter a user name and password for each application. Oracle Application Server Single Sign-On retrieves user information from an OID directory.

Oracle Application Server Portal

The OracleAS Portal allows customers to organize Web content and applications in a logical and consistent Web portal format. OracleAS Portal provides a flexible, sophisticated model for managing user access to OracleAS Portal resources based on user identity and privilege.

Oracle Application Server Certificate Authority

Oracle Application Server Certificate Authority generates and publishes X.509 v3 Public Key Infrastructure certificates to support strong authentication methods and digital certificates.

CHAPTER

3 Security Environment

This section identifies the IT assets protected by the TOE. It also identifies the threats to those IT assets, the organisational security policies supported by the TOE, and the assumptions for secure usage of the TOE.

IT Assets

The IT assets requiring protection consist of:

- the applications hosted by OC4J, which are to be protected against unauthorised access;
- the user repository data stored within the directory, the confidentiality, integrity or availability of which could be compromised; and
- other directory data required for the secure operation of the TOE.

The types of directory data to be protected are:

- *User entries* and *Group entries* which hold the TOE users' security credentials, and which hold some of the information about permissions for users to access applications hosted by OC4J;
- *Directory control data* used by the directory server to organise and protect the directory objects; and
- *Audit data* generated in the directory by the directory server during use of the TOE.

Threats

The assumed threats to the TOE security, along with the threat agents which might instigate these threats, are specified below. Each threat statement identifies a means by which the TOE and its underlying system might be compromised.

These threats will be countered by:

a) technical security measures provided by the TOE, in conjunction with

b) technical security measures provided by an underlying system, and

c) non-technical operational security measures (personnel, procedural and physical measures) in the environment.

Threat agents

The threat agents are:

- Persons who are not authorised users of the underlying system (operating system and/or database system and/or network services and/or custom software);
- Persons who are authorised users of the TOE;
- Persons who are authorised users of the underlying system. System Users may be:
  a) those persons who are not TOE users, or
  b) those persons who are TOE users;
- Interruptions to operations arising from failures of hardware, power supplies, storage media, etc.

Threat agents can initiate the types of threats against the Application Server that are listed below.

Threats countered by the TOE

The threats in this section are countered by technical security measures provided by the TOE, supported by technical security measures provided by the underlying system and non-technical operational security measures in the environment.

T.ACCESS *Unauthorised Access to the Directory.* An outsider or system user who is not (currently) an authorised TOE user accesses the directory which is used by the TOE to hold its user repository data. This threat includes: *Impersonation* - a person, who may or may not be an authorised TOE user, accesses the directory by impersonating an authorised TOE user (including an authorised user impersonating a different user who has different - possibly more privileged - access).

T.DATA *Unauthorised Access to Information.* An authorised TOE user accesses information contained within the directory, which is used to hold the TOE's user repository data, without the permission of the TOE user who has responsibility for ensuring that this data is protected.

Note that this threat includes unauthorised access to directory information, residual information held in memory, or storage resources managed by the TOE, or directory control data.

T.APPLICATION *Unauthorised Access to an Application Hosted by OC4J.* An outsider or a TOE user starts a session with an application hosted by OC4J without the permission or authorization of the TOE administrator who has responsibility for ensuring that this application is protected.

T.ATTACK *Undetected Attack.* An undetected compromise of the user repository occurs as a result of an attacker (whether an authorised TOE user or not) attempting to perform actions that the individual is not authorised to perform.

Note that this threat and T.APPATTACK are included because, whatever countermeasures are provided to address the other threats, there is still a residual threat of a violation of the security policy by attackers attempting to defeat these countermeasures. Both threats are analogous to that of a burglar rattling doors. There is an effective countermeasure to the threat of a burglar gaining unauthorised access (i.e. locks on the doors), but sooner or later a prolonged attack by a burglar will find a door that has not been locked, thus enabling the threat to be realised. The only way of addressing this residual threat is to detect the fact that such an attack is ongoing, which provides a chance to do something about it before it succeeds.

T.APPATTACK *Undetected Attack on an Application.* An undetected compromise of an application hosted by OC4J occurs as a result of an attacker (whether an authorised TOE user or not) attempting to perform actions that the individual is not authorised to perform.

T.ABUSE.USER *Abuse of Privileges Allowing User Repository Attack.* An undetected compromise of the user repository occurs as a result of a TOE user (intentionally or otherwise) performing actions the individual is authorised to perform.

Note that this threat and T.ABUSE.APP are included because, whatever countermeasures are provided to address the other threats, there is still a residual threat of a violation of the security policy occurring, or the TOE's assets being placed at risk, as a result of actions taken by authorised TOE users. For example, a TOE user may grant access to a directory object they are responsible for to another TOE user who is able to exploit this to perform a fraudulent action.
Note also that threats T.ABUSE.USER and T.ABUSE.APP do not extend to highly trusted TOE users (see assumption A.MANAGE below).

T.ABUSE.APP *Abuse of Privileges Allowing Application Attack.* An undetected compromise of an application hosted by OC4J occurs as a result of a TOE user (intentionally or otherwise) performing actions the individual is authorised to perform.

Threats countered only by the Operating Environment

T.OPERATE *Insecure Operation.* Compromise of the directory or an application hosted by OC4J may occur because of improper configuration, administration, and/or operation of the composite system.

T.CRASH *Abrupt Interruptions.* Abrupt interruptions to the operation of the TOE may cause security related data, such as audit data, to be lost or corrupted. Such interruptions may arise from human error or from failures of software, hardware, power supplies, or storage media.

T.PHYSICAL *Physical Attack.* Security-critical parts of the TOE or the underlying operating system and/or network services may be subjected to physical attack by unauthorised users which could compromise security.

Note that the security critical parts do not include the processing resources which are covered by A.PHYSICAL below.

Organisational Security Policies

P.ACCESS Access to directory objects is determined by:

a) the user identity and access control group memberships associated with the subject attempting the access; *and*

b) directory access control information directives that apply to the object.

Assumptions

The TOE is dependent upon both technical IT and operational aspects of its environment.

TOE Assumptions

A.TOE.CONFIG The TOE is installed, configured, and managed in accordance with [ECD] its evaluated configuration.

Underlying System Assumptions

A.PHYSICAL The processing resources of the TOE and the underlying system are located within controlled access facilities which prevents unauthorised physical access by outsiders, system users and TOE users.

A.SYS.CONFIG The underlying system (operating system and/or secure network services and database server) is installed, configured, and managed in accordance with its secure configuration documentation.

A.ACCESS The underlying system is configured such that only the approved group of individuals may obtain access to the system.

A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the underlying system and the security of the information it contains who can be trusted not to abuse their privileges.

A.PEER Any other IT components with which the TOE communicates are assumed to be under the same management control and operate under the same security policy.

A.NETWORK When required by the TOE, in a distributed environment the underlying network services are assumed to be based on secure communications protocols which ensure the authenticity of users.

Hosted Application Assumption

A.APPLICATION It is assumed that each application hosted by OC4J will implement appropriate measures to ensure that its users can be held accountable for their use of the application.

Note that [ECD] instructs administrators to ensure that each application hosted by OC4J logs the identity of the user, so that this person can be held accountable for their use of the application.

CHAPTER

4 Security Objectives

This section first describes the security objectives of the TOE and the threats and policies they address. Then the requirements on the operational environment needed to support the TOE objectives are presented.

TOE Security Objectives

This section defines the security objectives that are to be satisfied by the TOE in combination with the IT security environment. Table 4 in chapter 8 correlates the TOE security objectives to each of the threats and security policies, showing that each threat is countered by at least one security objective, and that each security policy is satisfied by at least one security objective.

In the definitions below, the directory referred to is the directory used to hold the TOE's user repository data.

O.APPLICATION The TOE must ensure that only properly authenticated and authorized users can open a session with an application hosted by OC4J.

O.ACCESS.OBJECTS The TOE must prevent the unauthorised or undesired disclosure, entry, modification, or destruction of directory data, objects, and control and audit data.

O.ACCESS.CONTROL The TOE must allow directory users who are responsible for administering directory data to control the access to that data by other authorised directory users.

O.ACCESS.RESIDUAL The TOE must prevent unauthorised access to residual data remaining in directory objects and resources e.g. memory or reused directory objects following the use of those objects and resources.

Note that the above three O.ACCESS objectives are concerned with the TOE providing end-users and administrators with the capability of controlling and limiting access by identified individuals, or grouping of individuals, to the data or resources they

are responsible for, in accordance with the P.ACCESS security policy.

O.I&A.TOE The TOE must provide the means of identifying and authenticating users of the TOE.

O.AUDIT The TOE must provide the means of recording security relevant events in sufficient detail to help an administrator of the TOE to:

a) detect attempted security violations, or potential misconfiguration of the TOE security features that would leave the TOE open to compromise; *and*

b) hold individual TOE users accountable for any actions they perform that are relevant to the security of the TOE.

O.ADMIN.TOE The TOE, where necessary in conjunction with the underlying system, must provide functions to enable an authorised administrator to effectively manage the TOE and its security functions, ensuring that only authorised administrators can access such functionality.

Environmental Security Objectives

The following IT security objectives are to be satisfied by the environment in which the TOE is used.

O.ADMIN.ENV The underlying system must provide functions to enable an authorised administrator to effectively manage the TOE and its security functions, ensuring that only authorised administrators can access such functionality. In particular, to enable the effective management of the TOE's audit functions:
a) the underlying database server's functions must include auditing of the startup and shutdown of the TOE's database sessions, and
b) the underlying operating system's functions must include the provision of reliable timestamps for use in audit records.

O.FILES The underlying system must provide access control mechanisms by which all of the user repository related files (including executables, run-time libraries, database files, export files, redo log files, control files, configuration files, trace files and dump files) and directory related database tables may be protected from unauthorised access.

O.SEP The underlying operating system must provide the means to isolate the TOE Security Functions (TSF) and assure that the TSF components cannot be tampered with.

The following non-IT security objectives are to be satisfied by procedural and other measures taken within the TOE environment.

O.INSTALL Those responsible for the TOE must ensure that:

a) The TOE is delivered, installed, managed and operated in accordance with the operational documentation of the TOE, and

b) The underlying system is installed and operated in accordance with its operational documentation. If the system components are certified they should be installed and operated in accordance with the appropriate certification documentation.

O.LOCKOUT The TOE is configured so that each user account is locked when the number of consecutive authentication failures has reached a level indicating that an attempt is probably being made to discover the user's password. Before unlocking the user's account, the administrator will investigate whether there is evidence of such an attack, and, if so, will take action to prevent or discourage further attacks.

Note that, to ensure O.LOCKOUT is met, [ECD] includes appropriate instructions for the administrator when setting the password policy for the TOE's evaluated configuration.

O.PHYSICAL Those responsible for the TOE must ensure that those parts of the TOE that are critical to the security policy are protected from physical attack.

O.AUDITLOG TOE administrators must ensure that audit facilities are used and managed effectively. These procedures shall apply to the directory audit trail, the audit log for applications hosted by OC4J, the audit trail for the underlying operating system and the database server and/or secure network services. In particular:

a) Appropriate action must be taken to ensure continued audit logging, e.g. by regular archiving of logs before audit trail exhaustion to ensure sufficient free space;

b) Audit logs must be inspected on a regular basis and appropriate action should be taken on the detection of breaches of security, or events that are likely to lead to a breach in the future;

c) The system clocks must be protected from unauthorised modification (so that the integrity of audit timestamps is not compromised); and

d) Each application hosted by OC4J must log the identity of the user (using information provided to it by OC4J).

Note that [ECD] instructs administrators to ensure that each application hosted by OC4J logs the identity of the user, so that this person can be held accountable for their use of the application (see A.APPLICATION in chapter 3).

O.RECOVERY Those responsible for the TOE must ensure that procedures are in place to ensure that, after system failure or other discontinuity, recovery without security compromise is obtained.

O.TRUST Those responsible for the TOE must ensure that only highly

trusted users have the privilege which allows them to:

a) set or alter the OID audit trail configuration;

b) modify the contents of the OID audit trail;

c) create any user account or modify any user security attributes;

c) create or modify any group entries and TOE configuration files that give permission for users to access applications hosted by the TOE; or

d) authorise use of administrative privileges.

O.AUTHDATA Those responsible for the TOE must ensure that the authentication data for each user account for the TOE as well as the underlying system is held securely and not disclosed to persons not authorised to use that account. In particular:

a) The media on which the authentication data for the underlying operating system is stored shall not be physically removable from the underlying platform by unauthorised users;

b) Users shall not disclose their passwords to other individuals;

c) Passwords generated by the system administrator shall be distributed in a secure manner.

O.MEDIA Those responsible for the TOE must ensure that the confidentiality, integrity and availability of directory data held on storage media is adequately protected. In particular:

a) The on-line and off-line storage media on which user repository and security related data (such as operating system backups, database backups and transaction logs, and audit trails) must not be physically removable from the underlying platform by unauthorised users;

b) The on-line and off-line storage media must be properly stored and maintained, and routinely checked to ensure the integrity and availability of the security related-data;

c) The media on which directory-related files (including database files, export files, redo log files, control files, trace files and dump files) have been stored shall be purged prior to being re-used for any non-directory purpose.

Table 5 in chapter 8 illustrates how each of the above objectives counters a threat, supports a TOE security objective, supports a policy, or maps to a secure usage assumption.

CHAPTER

5 IT Security Requirements

TOE Security Functional Requirements

Table 1 below lists each Security Functional Requirement (SFR) included in this Security Target. This table identifies which Common Criteria operations (assignment (A), selection (S), refinement (R), and/or iteration (I)) have been applied to the requirement relative to Part 2 of [CC]. The text for such completed operations is highlighted with *ITALICISED CAPITAL LETTERS* within each requirement. SFRs in Table 1 that are extended relative to Part 2 of [CC] are indicated in this table by the presence of a "*" after the element identifier.

The remainder of this section details the functional requirements for this Security Target. Annex B provides definitions for various terms used in the functional requirements. In the requirements below, the directory referred to is the directory used to hold the TOE's user repository data. Note that the phrase "suitably authorised users", which is used in the SFRs listed below, refers to users who are permitted by the User Repository Access Control SFP to perform the operation in question.

Table 1: List of Security Functional Requirements

Element	Name	A	S	R	I
FAU_GEN.1T.1 *	Audit Data Generation	X	X		
FAU_GEN.1T.2 *	Audit Data Generation	X			
FAU_GEN.2.1	User Identity Association				
FAU_SAR.1.1	Audit Review	X			
FAU_SAR.1.2	Audit Review				
FAU_SAR.3.1	Selectable Audit Review	X	X		
FAU_SEL.1.1	Selective Audit	X	X		

Element	Name	A	S	R	I
FAU_STG.1.1	Protected Audit Trail Storage				
FAU_STG.1.2	Protected Audit Trail Storage		X		
FAU_STG.4.1	Prevention of Audit Data Loss	X	X	X	
FDP_ACC.1.1	Subset Access Control	X			
FDP_ACF.1.1	Security Attribute Based Access Control	X			
FDP_ACF.1.2	Security Attribute Based Access Control	X			
FDP_ACF.1.3	Security Attribute Based Access Control	X			
FDP_ACF.1.4	Security Attribute Based Access Control	X			
FDP_RIP.2.1	Full Residual Information Protection		X		
FIA_AFL.1.1	Authentication Failure Handling	X			
FIA_AFL.1.2	Authentication Failure Handling	X			
FIA_ATD.1.1	User Attribute Definition	X			
FIA_SOS.1.1	Verification of Secrets	X			
FIA_UAU.1.1	Timing of Authentication	X			
FIA_UAU.1.2	Timing of Authentication				
FIA_UID.1.1	Timing of Identification	X			
FIA_UID.1.2	Timing of Identification				
FMT_MSA.1.1	Management of Security Attributes	X	X		
FMT_MSA.3.1	Static Attribute Initialisation	X	X		
FMT_MSA.3.2	Static Attribute Initialisation	X			
FMT_MTD.1.1	Management of TSF Data	X	X		X
FMT_REV.1.1	Revocation	X	X		
FMT_REV.1.2	Revocation	X			
FMT_SMF.1.1	Specification of Management Functions	X			
FMT_SMR.1.1	Security Roles	X			
FMT_SMR.1.2	Security Roles				
FPT_RVM.1.1	Non-Bypassability of the TSP				
FPT_SEP.1T.1 *	TSF Domain Separation				
FPT_SEP.1T.2 *	TSF Domain Separation				
FTA_TSE.1.1	TOE Session Establishment	X		X	X

Security Audit

FAU_GEN.1T.1 The TSF shall be able to generate an audit record of the following auditable events:

a) All auditable events *AS IDENTIFIED IN TABLE 2 BELOW*.

*Note that the selection operation for the FAU_GEN.1.1 element defined in Section 3.2 of [CC] Part 2 has effectively been completed with "*for the *NOT SPECIFIED* level of audit*". However, a refinement has been applied to omit these words for the sake of clarity.*

Table 2: Required Auditable Events

Component	Event	Additional Data
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	*IDENTITY OF DIRECTORY ENTRY MODIFIED*
FDP_ACF.1	Unsuccessful requests to perform an operation on an object covered by the SFP and successful requests to perform an operation which changes an object covered by the SFP	*OBJECT IDENTIFIER, REQUESTED ACCESS*
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts and the actions taken and the subsequent, if appropriate, restoration to the normal state, except that there are no audit events associated with authentication to OC4J	None
FIA_SOS.1	Rejection or acceptance by the TSF of any tested secret, except that there are no audit events associated with authentication to OC4J	None
FIA_UAU.1	Unsuccessful and successful use of the authentication mechanism, except for anonymous authentication and authentication to OC4J	None
FIA_UID.1	Unsuccessful and successful use of the user identification mechanism (including the user identity provided), except if anonymous authentication is used or if OC4J's user identification mechanism is used	None
FMT_MSA.1	All modifications of the values of security attributes	None
FMT_MSA.3	All modifications of the initial values of security attributes	None
FMT_REV.1	Unsuccessful revocation of security attributes	*SECURITY ATTRIBUTE*
FMT_SMF.1	Specification of Security Management Functions.	None
FMT_SMR.1	Modifications to the group of users that are part of a role	None

Note that the FAU_GEN.1.1 element defined in Section 3.2 of [CC] Part 2 requires that the TSF shall be able to generate an audit record for the start-up and shutdown of the audit functions. However, auditing takes place throughout the TOE's directory session according to the value of the audit level at the start of the session.

To cater for this, FAU_GEN.1 has been extended as a requirement for this TOE. This extended component has been designated as FAU_GEN.IT. The Security Requirements for the IT Environment defined later in this Chapter include the requirement FAU_GEN.1E.2 for the IT Environment to be able to generate an audit record for the start-up and shutdown of the TOE's database session (during which any audit records generated by the TOE are stored in a database table). This requirement therefore satisfies the need for the TOE administrator to know the periods of time during which audit records could have been written to the TOE's audit trail.

Note that, as stated in assumption A.APPLICATION and the environmental objective O.AUDIT_LOG part d), responsibility for auditing of application access rests with the application, which must log the identity of the user accessing it.

FAU_GEN.1T.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 *SECURITY TARGET AND OTHER AUDIT RELEVANT INFORMATION AS IDENTIFIED IN TABLE 2 ABOVE*.

Note that FAU_GEN.1T.2 is identical to FAU_GEN.1.2

FAU_GEN.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SAR.1.1 The TSF shall provide *SUITABLY AUTHORISED USERS* with the capability to read *ALL AUDIT INFORMATION* from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

FAU_SAR.3.1 The TSF shall provide the ability to perform *SEARCHES* of audit data based on *THE VALUES OF AUDIT DATA ATTRIBUTES.*

FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) *EVENT TYPE*;
b) *AND NO OTHER ATTRIBUTES.*

FAU_STG.1.1 The TSF shall protect the stored audit records from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to *PREVENT* unauthorised modifications to the audit records in the audit trail.

FAU_STG.4.1 The TSF shall *IGNORE AUDITABLE EVENTS* if the audit trail is full.

Note that the assignment operation for the FAU_STG.4.1 element defined in Section 3.6 of [CC] Part 2 has effectively been completed with "and NO OTHER ACTIONS". However, a refinement has been applied to omit these words for the sake of clarity.

User Data Protection

FDP_ACC.1.1 The TSF shall enforce the *USER REPOSITORY ACCESS CONTROL SFP* on:

a) *REPOSITORY USERS*;

b) *REPOSITORY OBJECTS, EACH OF WHICH IS A REPOSITORY ENTRY OR AN ATTRIBUTE OF A REPOSITORY ENTRY*; *AND*
c) *OPERATIONS PROVIDING THE TYPES OF ACCESS IN THE FOLLOWING LIST:*
COMPARE
SEARCH
BROWSE
PROXY
READ
SELFWRITE
WRITE
ADD
DELETE.

FDP_ACF.1.1 The TSF shall enforce the *USER REPOSITORY ACCESS CONTROL SFP* to objects based on:

a) *THE USER IDENTITY AND ACCESS CONTROL GROUP MEMBERSHIPS ASSOCIATED WITH THE SUBJECT; AND*
b) *THE ACCESS CONTROL INFORMATION (ACI) DIRECTIVES THAT APPLY TO THE OBJECT.*

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

a) *WHEN A USER TRIES TO PERFORM AN OPERATION ON AN OBJECT, THE TSF SHALL EVALUATE WHETHER ACCESS IS ALLOWED BY EXAMINING ALL OF THE ACI DIRECTIVES THAT APPLY TO THAT OBJECT.*
b) *IF THE OBJECT IS AN ENTRY, ACCESS SHALL BE EVALUATED FOR THE ENTRY AND EACH OF ITS ATTRIBUTES.*
c) *THE TSF SHALL EVALUATE ACCESS BY FIRST EXAMINING THE ACI DIRECTIVES IN THE ENTRY-LEVEL ACI FOR THE OBJECT. IT SHALL PROCEED TO THE NEAREST ACCESS CONTROL POINT (ACP), AND SHALL THEN CONSIDER EACH SUPERIOR ACP IN SUCCESSION UNTIL THE EVALUATION IS COMPLETE.*
d) *DURING ACCESS EVALUATION, AN ATTRIBUTE'S STATUS IS "RESOLVED WITH PERMISSION" IF THE REQUIRED ACCESS FOR THE ATTRIBUTE HAS BEEN GRANTED IN THE ACI; ITS STATUS IS "RESOLVED WITH DENIAL" IF THE REQUIRED ACCESS FOR THE ATTRIBUTE HAS BEEN EXPLICITLY DENIED IN THE ACI; AND ITS STATUS IS "UNRESOLVED" IF NO APPLICABLE ACI HAS YET BEEN ENCOUNTERED FOR THE ATTRIBUTE.*
e) *FOR A SEARCH OPERATION, THE ACCESS EVALUATION SHALL CONTINUE UNTIL ALL THE ATTRIBUTES REACH A RESOLVED STATE. ATTRIBUTES THAT ARE RESOLVED WITH DENIAL SHALL NOT BE RETURNED.*

f) *FOR OPERATIONS OTHER THAN SEARCH, THE ACCESS EVALUATION SHALL STOP IF ACCESS TO THE ENTRY ITSELF IS DENIED OR IF ANY OF THE ATTRIBUTES REACH A RESOLVED WITH DENIAL STATE. IN THIS CASE, ACCESS IS DENIED, OTHERWISE ACCESS IS GRANTED.*

Note that exceptions to the normal case presented in this SFR are provided by FDP_ACF.1.3 and FDP_ACF.1.4.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules:

a) *DURING ACCESS EVALUATION, AN AUTHORISED USER REPOSITORY ADMINISTRATOR SHALL BE GRANTED ACCESS TO ANY OBJECT.*
b) *DURING ACCESS EVALUATION, A USER SHALL BE GRANTED ACCESS TO WRITE TO THE ATTRIBUTE OF THE USER'S ENTRY THAT HOLDS THE USER'S PASSWORD.*
c) *IF ACCESS TO AN ENTRY IS DENIED OR IF AN ATTRIBUTE REACHES A RESOLVED WITH DENIAL STATE, THEN IF THE USER IS A MEMBER OF A PRIVILEGE GROUP OBJECT, THE ACCESS EVALUATION SHALL CONTINUE AS IF IT IS STILL UNRESOLVED. IF A STATE OF "RESOLVED WITH PERMISSION" IS REACHED THROUGH A GROUP SUBJECT SELECTOR DURING THIS CONTINUING ACCESS EVALUATION, THEN THIS RESOLUTION EFFECTIVELY OVERRIDES THE EARLIER DENIAL STATE.*

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *NONE*

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *ALLOCATION OF A RESOURCE TO* all objects.

Identification and Authentication

FIA_AFL.1.1 The TSF shall detect when *AN ADMINISTRATOR CONFIGURABLE POSITIVE INTEGER WITHIN THE RANGE 1 TO 10^{63} - 1* unsuccessful authentication attempts occur related to *A USER BINDING TO THE DIRECTORY OR COMPARING A USER'S PASSWORD WITH A PARTICULAR VALUE.*

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall *LOCK THE USER'S ACCOUNT.*

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:

a) *USER IDENTITY;*
b) *GROUP MEMBERSHIPS;*
c) *AUTHENTICATION DATA;*
d) *PERMISSIONS TO ACCESS APPLICATIONS HOSTED BY THE TOE.*

Note that a permission for a user to access an application hosted by the TOE is to be represented as membership by the user of a group which is mapped via TOE configuration files to a security role that has permission to access the application. The TOE reads such configuration files to obtain information, but is not required

to provide facilities for their maintenance.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet *REUSE, LIFETIME, AND CONTENT METRICS AS DEFINED BY AN AUTHORISED ADMINISTRATOR*.

FIA_UAU.1.1 The TSF shall allow *VIEWING OF PUBLICLY AVAILABLE USER REPOSITORY INFORMATION* on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UID.1.1 The TSF shall allow *VIEWING OF PUBLICLY AVAILABLE USER REPOSITORY INFORMATION* on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Security Management

FMT_MSA.1.1 The TSF shall enforce the *USER REPOSITORY ACCESS CONTROL SFP* to restrict the ability to *MODIFY, DELETE, CREATE* the security attributes *USER IDENTITY, GROUP MEMBERSHIPS AND AUTHENTICATION DATA FOR USERS, AND ACCESS CONTROL INFORMATION FOR OBJECTS* to *SUITABLY AUTHORISED USERS.*

FMT_MSA.3.1 The TSF shall enforce the *USER REPOSITORY ACCESS CONTROL SFP* to provide *RESTRICTIVE* default values for security attributes that are used to enforce the SFP.

Note that Section H.2 of Part 2 of [CC] states that FMT_MSA.3.1 applies only to security attributes for objects.

FMT_MSA.3.2 The TSF shall allow *SUITABLY AUTHORISED USERS* to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1.1.1 The TSF shall restrict the ability to *QUERY, CLEAR* the *AUDIT TRAIL* to *SUITABLY AUTHORISED USERS.*

FMT_MTD.1.1.2 The TSF shall restrict the ability to *MODIFY* the *SET OF AUDITED EVENTS* to *AUTHORISED USER REPOSITORY ADMINISTRATORS.*

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the *USERS AND OBJECTS* within the TSC to *SUITABLY AUTHORISED USERS*.

FMT_REV.1.2 The TSF shall enforce the rules:

a) *THE REVOCATION OF A USER'S SECURITY ATTRIBUTES SHALL BE IN EFFECT WHEN THE USER NEXT BINDS FOR A USER REPOSITORY SESSION;*

b) *THE REVOCATION OF AN OBJECT'S SECURITY ATTRIBUTES SHALL BE IN EFFECT WHEN A USER NEXT ATTEMPTS TO ACCESS THE OBJECT.*

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions:

a) *QUERY, CLEAR* the *AUDIT TRAIL.*

b) *QUERY, MODIFY* the *SET OF AUDITED EVENTS.*

c) *MODIFY, DELETE, CREATE the SECURITY ATTRIBUTES.*

Note, please refer to FIA_ATD.1.1 and FMT_MSA.1.1 for the definition of SECURITY ATTRIBUTES.

FMT_SMR.1.1 The TSF shall maintain the roles:

a) *AUTHORISED USER REPOSITORY ADMINISTRATOR*;

b) *USER*.

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Protection of the TOE Security Functions

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

FPT_SEP.1T.1 The TSF shall maintain a security domain for its own execution *SO THAT ITS UNDERLYING OPERATING SYSTEM CAN* protect it from interference and tampering by untrusted subjects.

Note that the FPT_SEP.1.1 element defined in Section 10.11 of CC Part 2 requires that the TSF shall be self-protecting so that an untrusted subject cannot modify or damage the TSF. However, it is the underlying operating system which protects the TSF against untrusted subjects modifying or damaging it during its execution. To cater for this, FPT_SEP.1 has been extended as a requirement for the TOE. This extended component has been designated as FPT_SEP.1T. FPT_SEP.1T.1 is underpinned by the Security Requirement for the IT Environment FPT_SEP.1E.1 which requires the underlying operating system to protect the TSF against untrusted subjects modifying or damaging it during its execution.

FPT_SEP.1T.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

Note that FPT_SEP.1T.2 is identical to the FPT_SEP.1.2 element defined in section 10.11 of CC Part 2.

TOE Access

FTA_TSE.1.1.1 The TSF shall be able to deny session establishment based on *EXPIRATION OF A USER'S AUTHENTICATION DATA.*

FTA_TSE.1.1.2 The TSF shall be able to deny session establishment *WITH AN APPLICATION HOSTED BY THE TOE* based on *THE ABSENCE OF A PERMISSION FOR THE USER TO ACCESS THE APPLICATION.*

Note that a permission for a user to access an application hosted by the TOE is to be represented as membership by the user of a group which is mapped via TOE configuration files to a security role that has permission to access the application.

TOE Security Assurance Requirements

The target assurance level is EAL4 as defined in Part 3 of the CC, augmented with ALC_FLR.3.

Security Requirements for the IT Environment

This section details the security requirements for the IT Environment.

Support for SFRs

The functional requirements for the IT Environment to support the SFRs defined in this chapter are defined below via elements which have been extended relative to Part 2 of [CC] (using refinements which have been highlighted with *ITALICISED CAPITAL LETTERS*).

FAU_GEN.1E.1 The *DATABASE SYSTEM UNDERLYING THE TSF* shall be able to generate an audit record of the following auditable event:

a) start-up and shutdown of the *TSF'S DATABASE SESSION*.

FAU_GEN.1E.2 The *DATABASE SYSTEM UNDERLYING 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 as defined for SFR FAU_GEN.1.2 in [DPP, 5].

Note that FAU_GEN.1T.1, FAU_GEN.1T.2, FAU_GEN.1E.1 and FAU_GEN.1E.2 together meet the requirements of the FAU_GEN.1 component defined in Section 3.2 of [CC] Part 2.

FPT_SEP.1E.1 The *OPERATING SYSTEM UNDERLYING THE TSF DURING ITS EXECUTION SHALL PROTECT* it from interference and tampering by untrusted subjects.

Note that FPT_SEP.1T.1 and FPT_SEP.1E.1 together meet the requirements of the FPT_SEP.1.1 element defined in Section 10.11 of [CC] Part 2. FPT_SEP.IT.2 is not relevant to the IT environment and hence there is no equivalent requirement called FPT_SEP.1E.2.

FPT_STM.1.1 The *OPERATING SYSTEM UNDERLYING THE TSF DURING ITS EXECUTION* shall be able to provide reliable time stamps for *USE BY THE TSF*.

Note that FPT_STM.1.1 satisfies the dependency of the SFR FAU_GEN.1.2 for the provision of reliable time stamps.

Support for security objectives

The underlying operating system, database server, network services and/or customer software (collectively, the *system*) shall support the security objectives of the TOE as follows:

O.I&A.TOE The operating system and database server shall identify and authenticate users prior to providing access to the underlying system.

O.ACCESS The system shall provide the access control mechanisms required to support O.FILES and A.NETWORK. In addition these mechanisms are required to support O.AUTHDATA and O.ADMIN.TOE.

O.RECOVERY & O.AUDITLOG The system shall provide backup, restore and other secure recovery mechanisms. Such mechanisms are to be capable of archiving and restoring the directory audit trail.

Note that O.AUDITLOG d) is not refined into a requirement on the IT environment as the auditing requirements are dependent on the functionality of the individual application and the assets that it may process (which are outside the scope of the TOE).

In addition to the above, the system shall provide mechanisms to ensure that the system security functions are always invoked prior to passing control to the TOE and that non-TOE activity within the system does not interfere with the operation of the TOE. Thus the system shall at least support FPT_RVM.1 and FPT_SEP.1. Also the underlying operating system platform should perform testing to demonstrate the security assumptions made about the underlying abstract machine upon which the TSF relies. Therefore the system shall also support FPT_AMT.1.

Note that an operating system meeting the functional and assurance requirements defined in [CAPP], or equivalent, and a database system meeting the functional and assurance requirements defined in [DPP], or equivalent, will meet the above requirements (although conformance to [CAPP] and [DPP] is not a mandatory requirement).

Minimum Strength of Function

The minimum strength of function for the TOE is *SOF-High*.

CHAPTER

6 TOE Summary Specification

TOE Security Functionality

This section contains a high-level specification of each Security Function (SF) of the TOE that contributes to satisfaction of the Security Functional Requirements (SFRs) of chapter 5. The specifications cover five major areas: identification and authentication, application access control, security attribute maintenance, user repository access control, and audit and accountability.

Oracle Application Server Containers for J2EE (OC4J) has a user repository to hold information about users and authorization for such users to access applications that are hosted by OC4J. For this evaluation, the TOE includes an LDAP-based JAAS Provider, with the user repository data being managed by the Oracle Internet Directory (OID).

The Security Functions of the TOE which are implemented in OC4J and documented in this chapter are:

- IA.AUTH, which defines the identification and authentication checks made by OC4J;
- APP.PERMS, APP.APPREF and APP.ACCESS, which define how access to applications hosted by OC4J is controlled;
- RAC.OC4JSEP, which defines the domain separation function as it applies to the OC4J product.

The TOE allows direct access to the directory holding the user repository data via the Oracle Internet Directory product. OID is used to create and maintain user and group entries and to create and maintain user security attributes. Such entries are used for the identification and authentication of OC4J users and for the authorization of such users to access applications hosted by OC4J in the TOE's evaluated configuration.

Thus all of the Oracle Internet Directory SFs in chapter 6 of [OID_ST] are included here, but with some re-wording to show how the SFs apply to user repositories. The mapping of these SFs to the original OID SFs is described in the notes below each such SF.

Note that the phrase "userpassword check", which is used in the SFs listed below, refers to operations of the TOE which check whether the `userpassword` attribute of a user entry in the user repository has a particular value. These are LDAP `bind` operations for which a password has been supplied and LDAP `compare` operations which are acting on the `userpassword` attribute of a user entry.

Note that, in the descriptions of the SFs below, "authorized by RAC.POL" means "permitted by the user repository access control policy defined in RAC.POL".

Table 9 in chapter 8 shows that all the SFRs are satisfied by at least one SF and that every SF is used to satisfy at least one SFR (but note that SFR FDP_ACF.1.4 is not explicitly satisfied by any particular SF because this SFR specifies null functionality).

Identification and Authentication

IA.UID Each TOE user is uniquely identified via the distinguished name (DN) of the user's repository entry. The exception to this is that there are three special users: the super user, the guest user and the proxy user, whose names are held in the root directory-specific entry of the directory holding the user repository data. The super user is the administrative user for the user repository.

This SF is equivalent to SF IA.UID in Chapter 6 of [OID_ST].

IA.AUTH If a user requests access to a Web application hosted by the TOE and Basic authentication is configured, then if

a) the user provides a valid user identifier; and

b) the user's account is not locked; and

c) the user provides a password corresponding to the stored password for that user;

then the user will be deemed to be authenticated.

Note that SF IA.PWDC covers the conditions under which a user's account can be locked. In particular, if check c) fails, then IA.PWDC a) specifies the condition under which the user's account may consequently be locked.

IA.ASESS If a connection is requested to a user repository's directory server and no username and no password are supplied, the TOE will create an anonymous user repository session in which only publicly available material authorised for access by anonymous users is accessible.

This SF is equivalent to SF IA.ASESS in Chapter 6 of [OID_ST].

IA.USESS If a user requests to connect to a user repository's directory server and that user is configured for simple authentication, then if

a) the user provides a valid user identifier; and

b) the user's account is not locked; and

c) the user provides a password corresponding to the stored password for that user;

then the TOE will create a user repository session for the user.

Note that SF IA.PWDC covers the conditions under which a user's account can be locked. In particular, if check c) fails, then IA.PWDC a) specifies the condition under which the user's account may consequently be locked.
This SF is equivalent to SF IA.USESS in Chapter 6 of [OID_ST].

IA.IDE A subject can only submit requests to a user repository's directory server and receive responses (information) from a user repository's directory server while the subject is establishing or has established a user repository session as per IA.ASESS or IA.USESS. During a user repository session, the TOE will either have access to information recording the fact that this session is anonymous or will have access to the user's identifier.

This SF is equivalent to SF IA.IDE in Chapter 6 of [OID_ST].

IA.PSESS If a user repository session has been established as per IA.USESS and a connection is requested to the user repository's directory server from within this user repository session, then if

a) the connection request supplies a valid identifier for a user, but supplies no password; and

b) the user for the user repository session has proxy access to the new user's entry as per the policy defined in RAC.POL;

then the user repository session will continue under the new user's identifier.

Note that this SF describes a process by which the user repository session's user can act as a proxy for the user specified in the new connection request.
This SF is equivalent to SF IA.PSESS in Chapter 6 of [OID_ST].

IA.CRUG The TOE will allow users to create user entries and group entries in the user repository only if this is authorized by RAC.POL. The default values of attributes of such new user repository entries are as described in Chapters 6, 7 and 9 of [OIDAG].

This SF is equivalent to SF IA.CRUG in Chapter 6 of [OID_ST].

IA.PWDC The following configurable controls can be applied to user passwords held in the user repository:

a) the number of consecutive failed userpassword checks before the user account is locked;

b) the maximum length of time the same password can be used before it expires and the user's account becomes locked;

c) the number of seconds before password expiration that the user repository's directory server sends a warning to the user;

d) whether the current password can be reused when the user password attribute of a user is modified;

e) the number of seconds a user account will remain locked after the specified number of consecutive failed userpassword checks;

f) the number of seconds after which the count of the number of consecutive failed userpassword checks is purged from the user entry;

g) after a password expires, the maximum number of grace logins that are allowed before the user's account is locked; and

h) a password complexity check performed on values supplied for the user password attribute of a user entry.

None of the above list of controls applies to the three special users: the super user, the guest user and the proxy user. The exception is that the value configured for a) gives the number of consecutive failed super user login attempts before the super user account is locked.

Note that the TOE provides more configurable controls on user passwords than are listed in the above SF. [OIDAG, Annex B: Password Policy Scheme Elements] defines the full set of such controls.
Note also that the TOE does not accept values to be configured for a) if they are negative or greater than 10^{63}-1.
This SF is equivalent to SF IA.PWDC in Chapter 6 of [OID_ST].

IA.PWDCM Configurable controls on user passwords held in the user repository are held in a password policy entry. A user can read or modify attributes in a password policy entry only if this is authorized by RAC.POL. Changes to a password policy entry only take effect after the directory server instance has next been re-started.

This SF is equivalent to SF IA.PWDCM in Chapter 6 of [OID_ST].

Application Access Control

APP.PERMS The user repository and configuration files can contain information about permissions for users to access applications hosted by the TOE. Such information includes user membership of groups which are mapped via configuration files to security roles that have access to applications.

[OC4JSG, 6: J2EE and JAAS Provider Role Mapping] describes how configuration files can be used to associate OID groups with security roles that have access permissions for applications. [OC4JUG, A] and [WebXML] give a full definition of the XML configuration files that are used by OC4J. The TOE reads such configuration files to obtain information, but is not involved in their maintenance. Tools such as editors may be used to maintain the information in TOE configuration files. [ECD] requires administrators to use operating system DAC permissions to ensure that only administrators can access TOE configuration files.

APP.APPREF The TOE correctly resolves every reference to a Web application hosted by the TOE to which a user is requesting access.

APP.ACCESS Before authorizing an authenticated TOE user to be granted a session with a Web application hosted by the TOE, the TOE must check that the user is a member of a group defined in the user repository. In addition, that group must be mapped via configuration files to a security role that has permission to access the application.

Note that the checks covered by APP.ACCESS relate to information about permissions for users to access applications hosted by the TOE, which is covered by APP.PERMS.

Security Attribute Maintenance

SAM.UATT The user repository contains a set of security attributes for each TOE user, including relative distinguished name, group memberships and password.

This SF is equivalent to SF SAM.UATT in Chapter 6 of [OID_ST].

SAM.EATT The user repository contains a set of security attributes for each user repository entry. These include the Access Control Information related to the attributes for the entry and the entry itself. During the installation of the TOE, a directory is created to hold the user repository data. The default Access Control Information for this directory is as described in the Default Access Policies section of Chapter 17 of [OIDAG].

This SF is equivalent to SF SAM.EATT in Chapter 6 of [OID_ST].

SAM.CHPWD The following constraints apply when a user attempts to change the user password attribute of a user entry and this is authorized by RAC.POL:

a) if the password profile applying to the user entry includes a complexity check function, then the new password is accepted only if it meets the criteria of the complexity check; and

b) if the password profile applying to the user entry specifies password reuse constraints and the user attempts to reuse a password, the TOE rejects the change if the reuse constraints are not met.

This SF is equivalent to SF SAM.CHPWD in Chapter 6 of [OID_ST].

SAM.MODATT A user can create, read, modify or delete security attributes for user repository users and user repository entries only if this is authorized by RAC.POL.

This SF is equivalent to SF SAM.MODATT in Chapter 6 of [OID_ST].

SAM.UEFF A user security attribute defined by SAM.UATT will be effective in a user repository session only if the user had that attribute at the start of the session.

This SF is equivalent to SF SAM.UEFF in Chapter 6 of [OID_ST].

SAM.OEFF A user repository object security attribute defined by SAM.EATT will be effective for user repository access control only if the user had that attribute when the access was attempted.

This SF is equivalent to SF SAM.OEFF in Chapter 6 of [OID_ST].

User Repository Access Control

RAC.OBID The TOE ensures that every object created in a user repository is uniquely identified in that repository via the distinguished name (DN) of the user repository entry.

This SF is equivalent to SF DAC.OBID in Chapter 6 of [OID_ST].

RAC.OBREF The TOE correctly resolves every reference to a user repository object, including knowledge references via referrals.

This SF is equivalent to SF DAC.OBREF in Chapter 6 of [OID_ST].

RAC.SUA The TOE enforces the user repository access control policy on user repository users based on the following subject attributes:

a) the identity of the user; and

b) any access control group memberships associated with the user.

This SF is equivalent to SF DAC.SUA in Chapter 6 of [OID_ST].

RAC.OBA The TOE enforces the user repository access control policy defined in RAC.POL on each user repository object based on the access control information (ACI) directives that apply to the object. The entries in a user repository and the attributes of user repository entries constitute the set of user repository objects.

This SF is equivalent to SF DAC.OBA in Chapter 6 of [OID_ST].

RAC.POL When a TOE user attempts to perform an LDAP operation on a user repository object, the TOE enforces the rules specified in FDP_ACF.1.2 that relate to the operations listed in FDP_ACC.1.1 by imposing the directory access control policy that applies to the directory holding the user repository. The rules of this policy are given in Chapter 14 of [OIDAG]. The super user for the directory is the administrative user for the user repository and therefore has access to all user repository objects. A user is always allowed to modify the `userpassword` attribute of that user's repository entry.

This SF is equivalent to SF DAC.POL in Chapter 6 of [OID_ST].

RAC.OIDSEP The OID product does not allow interference between concurrent uses of the TOE and the OID product does not perform any actions which could allow untrusted subjects to observe or modify the TOE's internal data or code.

This SF is equivalent to SF DAC.SEP in Chapter 6 of [OID_ST].

RAC.OC4JSEP The OC4J product does not allow interference between concurrent uses of the TOE and the OC4J product does not perform any actions which could allow untrusted subjects to observe or modify the TOE's internal data or code.

RAC.OR When a new user repository object is created, none of the information previously contained in the resource allocated to

the object will be capable of being accessed by any user repository user.

This SF is equivalent to SF DAC.OR in Chapter 6 of [OID_ST].

Audit and Accountability

AUD.INF When the audit level for the directory holding the user repository data is non-zero, for every occurrence of an auditable event described in Chapter 10 of [OIDAG], the TOE will write an audit record which holds the following information:

date and time of event; type of event; subject identity (which may be that of the anonymous user); and the outcome (success or failure) of the event.

In addition:

a) when the audit level is changed, the identity of the directory entry modified is recorded;

b) when a user attempts to access a directory object, the object identifier and the requested access operation is recorded (provided that the operation was unsuccessful or caused a change to the object); and

c) when a security attribute is modified, the attribute name is recorded.

Note that the TOE events that are auditable are those events that arise from the use of Oracle Internet Directory to access the directory holding the user repository.
This SF is equivalent to SF AUD.INF in Chapter 6 of [OID_ST].

AUD.SET The TOE will allow only the super user to set the audit level to specify which types of event are auditable.

This SF is equivalent to SF AUD.SET in Chapter 6 of [OID_ST].

AUD.ACC The TOE will allow only users authorized by RAC.POL to view all records in the audit log in a format suitable for the users to interpret the information. The TOE provides facilities to search for audit records according to their attribute values.

This SF is equivalent to SF AUD.ACC in Chapter 6 of [OID_ST].

AUD.DEL The TOE will allow only authorized users to delete audit records from the audit log (such users are authorized by the system administrator, who will inform them of the OID password). No other modification to the audit records is permitted.

This SF is equivalent to SF AUD.DEL in Chapter 6 of [OID_ST].

AUD.FULL If the audit log becomes full, auditable actions are not audited until space has been made available to write further audit records.

Note that the TOE attempts to write audit entries to the user repository for auditable actions even when the audit log is full. Under such circumstances, the writing of the audit entry will fail and messages are output to report the failure.
This SF is equivalent to SF AUD.FULL in Chapter 6 of [OID_ST].

Security Mechanisms and Techniques

A password is used for authentication of TOE users. The TOE employs a one-way hashing algorithm to encrypt passwords prior to storing them in the user repository. The TOE password management functions (together called the PWD mechanism) provide a Strength of Function level of *SOF-High*.

Specific SFs supporting the claimed SOF are:

- IA.AUTH and IA.USESS (SOF-High); *and*
- IA.PWDC, SAM.UATT and SAM.CHPWD, which support IA.AUTH and IA.USESS by providing password management facilities.

Assurance Measures

The target assurance level is EAL4 augmented with ALC_FLR.3. The following table indicates the documentation that will be supplied to support each security assurance requirement for EAL4 and also the assurance requirement for ALC_FLR.3. No other specific assurance measures are claimed.

Table 3: Oracle Application Server Containers for J2EE Assurance Measures

Component	Name	Documents
ACM_AUT.1	Partial CM Automation	Document(s) describing the TOE's configuration management will be provided.
ACM_CAP.4	Generation Support and Acceptance Procs	Document(s) describing the TOE's configuration management will be provided.
ACM_SCP.2	Problem Tracking CM Coverage	Document(s) describing the TOE's configuration management will be provided.
ADO_DEL.2	Detection of Modification	Document(s) describing the TOE's delivery procedures will be provided.
ADO_IGS.1	Installation, Generation, and Startup	Document(s) describing the TOE's installation and configuration will be provided.
ADV_FSP.2	Fully Defined External Interfaces	Document(s) covering the TOE's external interfaces will be provided.
ADV_HLD.2	Security Enforcing High-level Design	Document(s) describing the TOE's high level design will be provided.
ADV_IMP.1	Subset of the TSF Implementation	All of the TOE's source code will be provided.
ADV_LLD.1	Descriptive Low-level Design	Document(s) describing the TOE's low level design will be provided.
ADV_RCR.1	Informal Correspondence Demonstration	A demonstration of correspondence will be provided within the design documentation.
ADV_SPM.1	Informal TOE Security Policy Model	A document describing the TOE's Security Policy Model will be provided.
AGD_ADM.1	Administrator Guidance	Administrator guidance document(s) will be provided.

Table 3: Oracle Application Server Containers for J2EE Assurance Measures

Component	Name	Documents
AGD_USR.1	User Guidance	User guidance document(s) will be provided.
ALC_DVS.1	Identification of Security Measures	Document(s) covering the security of the TOE's development environment will be provided.
ALC_FLR.3	Systematic Flaw Remediation	Document(s) covering the flaw remediation procedures will be provided.
ALC_LCD.1	Developer Defined Life Cycle Model	Document(s) covering the TOE's life cycle model will be provided.
ALC_TAT.1	Well Defined Development Tools	Document(s) covering the TOE's development tools will be provided.
ATE_COV.2	Analysis of Coverage	Document(s) describing the TOE's developer testing will be provided.
ATE_DPT.1	Testing - High-level Design	Document(s) describing the TOE's developer testing will be provided.
ATE_FUN.1	Functional Testing	Document(s) describing the TOE's developer testing will be provided.
ATE_IND.2	Independent Testing	Document(s) describing the TOE's developer testing will be provided.
AVA_MSU.2	Validation of Analysis	Document(s) providing guidance analysis for the TOE will be provided.
AVA_SOF.1	Strength of TOE Security Functions	Document(s) analysing the strength of the TOE security functions will be provided.
AVA_VLA.2	Independent Vulnerability Analysis	Document(s) providing vulnerability analysis for the TOE will be provided.

This Page Intentionally Blank

CHAPTER

7

Protection Profile Claims

PP Reference

This security target does not make any claims about Protection Profile conformance.

This Page Intentionally Blank

CHAPTER

8 Rationale

Note that, in the sections below, the directory referred to is the directory used to hold the TOE's user repository data.

Security Objectives Rationale

This section demonstrates how the identified security objectives are suitable to counter the identified threats and meet the stated security policies.

The threats for the TOE and the security policies are stated in Chapter 3. The TOE security objectives and the environmental security objectives are stated in Chapter 4.

The table below covers those threats countered by the TOE and the security policies, showing that a threat is countered by at least one TOE security objective, and that each security policy is satisfied by at least one TOE security objective. This table does not cover threats addressed purely by the environment. A *YES* in the table indicates that the identified TOE security objective is relevant to the identified threat or security policy.

Table 4: Correlation of Threats and Policies to TOE Security Objectives

Threat/ Policy	O.I&A. TOE	O.ACCESS	O.AUDIT	O.ADMIN. TOE	O.APPLICATION
T.ACCESS	YES	YES		YES	
T.DATA	YES	YES		YES	
T.APPLICATION	YES	YES		YES	YES
T.ATTACK	YES	YES	YES	YES	
T.APPATTACK	YES	YES	YES	YES	
T.ABUSE.USER	YES	YES	YES	YES	

Table 4: Correlation of Threats and Policies to TOE Security Objectives

Threat/ Policy	O.I&A. TOE	O.ACCESS	O.AUDIT	O.ADMIN. TOE	O.APPLICATION
T.ABUSE. APP	YES	YES	YES	YES	
P.ACCESS		YES		YES	

The following table illustrates how each of the environmental security objectives counters a threat, supports a policy or maps to a secure usage assumption.

Table 5: Mapping of Environmental Security Objectives to Threats, Policy, and Secure Usage Assumptions

Environmental Objective	Counters Threat	Supports Policy	Maps to Secure Usage Assumptions
O.INSTALL	T.OPERATE		A.TOE.CONFIG, A.SYS.CONFIG, A.MANAGE, A.ACCESS, A.PEER
O.LOCKOUT	T.APPATTACK		A.TOE.CONFIG, A.MANAGE
O.PHYSICAL	T.PHYSICAL		A.PEER, A.PHYSICAL
O.AUDITLOG	T.ATTACK, T.APPATTACK, T.ABUSE.USER, T.ABUSE.APP		A.MANAGE, A.APPLICATION
O.RECOVERY	T.CRASH		A.MANAGE
O.TRUST		P.ACCESS	A.MANAGE, A.ACCESS
O.AUTHDATA	T.ACCESS, T.APPLICATION	P.ACCESS	A.MANAGE, A.NETWORK, A.ACCESS
O.MEDIA	T.CRASH		A.MANAGE
O.ADMIN.ENV		P.ACCESS	A.MANAGE, A.ACCESS
O.FILES	T.ACCESS, T.DATA, T.APPLICATION, T.ATTACK, T.APPATTACK	P.ACCESS	A.MANAGE
O.SEP	T.ACCESS, T.APPLICATION	P.ACCESS	A.MANAGE

T.ACCESS Rationale

T.ACCESS (*Unauthorised Access to the Directory)* is directly countered by O.I&A.TOE which ensures the TOE can protect the resources of the directory from access by persons not authorised to use the TOE. O.I&A.TOE ensures the TOE has the means of authenticating the claimed identity of any user. O.ACCESS.CONTROL and O.ADMIN.TOE provide support by controlling access to the directory control data and administrative functionality that might otherwise enable circumvention of the directory access controls. O.SEP and O.FILES together prevent bypass of the TOE. O.AUTHDATA ensures that authentication data is held securely to stop it being used by unauthorised users to authenticate to the TOE.

T.DATA Rationale

T.DATA (*Unauthorised Access to Information)* is directly countered by O.ACCESS.OBJECTS. O.ACCESS.OBJECTS ensures access is controlled to information contained within specific directory objects. O.ACCESS.RESIDUAL ensures access is prevented to residual information held in memory or reused directory objects. O.I&A.TOE provides support by providing the means of identifying the user attempting to access a directory object. O.ACCESS.CONTROL and O.ADMIN.TOE provide support by controlling access to directory control data and administrative functionality that might otherwise enable circumvention of directory object access controls. O.FILES prevents bypass of the TOE by users gaining direct access to files holding the directory data.

T.APPLICATION Rationale

T.APPLICATION (*Unauthorized Access to an Application hosted by OC4J)* is countered directly by O.APPLICATION which ensures the TOE can protect the application from unauthorized persons opening a session to use the application. O.I&A.TOE has the means of authenticating the claimed identity of any user. O.ACCESS.CONTROL and O.FILES provide support by controlling access to group entries and TOE configuration files, which hold information about permissions for users to access applications. O.ADMIN.TOE controls access to administrative functionality that might otherwise enable circumvention of the access controls. O.SEP and O.FILES together prevent bypass of the TOE. O.AUTHDATA ensures that authentication data is held securely to stop it being used by unauthorised users to authenticate to the TOE in order to gain access to an application.

T.ATTACK Rationale

T.ATTACK (*Undetected Attack)* is countered directly by O.AUDIT which ensures the TOE has the means of recording security relevant events which could be indicative of an attack aimed at defeating the directory's security features. O.I&A.TOE provides support by reliably identifying the user responsible for particular events, where the attacker is an authorised user of the TOE. O.ACCESS.CONTROL, and O.ADMIN.TOE provide support by controlling access to audit configuration data which only highly trusted individuals must be allowed to view and modify. O.FILES provides support by preventing users gaining direct access to files holding the directory audit data to modify evidence of an attack. O.AUDITLOG ensures that audit data is correctly managed by the administrator so that it can be used to detect attacks.

T.APPATTACK Rationale

T.APPATTACK (*Undetected Attack on an Application)* is directly countered by environmental security objective O.AUDITLOG part d), which ensures that each application hosted by OC4J logs the identity of the user. This enables the application administrator to check if any users are gaining unauthorised access to the application. O.LOCKOUT provides support by ensuring that attacks against the authentication mechanism come to the attention of the directory administrator (because the user accounts being attacked will become locked out and action from the directory administrator, including investigation of the attacks, will be needed if the accounts are to be re-enabled). O.AUDIT provides support by ensuring the TOE has the means of recording security relevant events which could be indicative of an attack involving attempts to perform unauthorized alteration of the TOE's user authentication and/or application authorization data. O.I&A.TOE provides support by reliably identifying the user responsible for particular events, where the attacker is an authorised user of the TOE. O.ACCESS.CONTROL and O.ADMIN.TOE provide support by controlling access to audit configuration data which only highly trusted individuals must be allowed to view and modify. O.FILES provides support

by preventing users gaining direct access to files holding the directory audit data (to modify evidence of an attack). O.AUDITLOG ensures that audit data is correctly managed by the administrator so that it can be used to detect attacks.

Note that [ECD] achieves the requirements of O.LOCKOUT by including instructions for the administrator when setting the password policy for the TOE's evaluated configuration. These instructions will ensure that each user account is locked when the number of consecutive authentication failures has reached a level indicating that an attempt is probably being made to discover the user's password. In addition, the [ECD] instructions will ensure that the authentication failures count in the password policy is set high enough to minimise account lockout due to user memory lapses and mistyping.

T.ABUSE.USER Rationale

T.ABUSE.USER (*Abuse of Privileges Allowing User Repository Attack)* is countered directly by O.AUDIT, which ensures the TOE has the means of recording security relevant events which could be indicative of abuse of privilege by an authorised user of the TOE who was accessing the directory. O.I&A.TOE provides support by reliably identifying the user responsible for particular events, thus ensuring that the user can be held accountable for actions for which he or she is responsible. O.ACCESS.CONTROL and O.ADMIN.TOE provide support by controlling access to audit configuration data which only highly trusted individuals must be allowed to view and modify. O.AUDITLOG ensures that audit data is correctly managed by the administrator so that it can be used to detect users abusing their privileges.

T.ABUSE.APP Rationale

T.ABUSE.APP (*Abuse of Privileges Allowing Application Attack)* is countered directly by O.AUDIT, which ensures the TOE has the means of recording security relevant events which could be indicative of abuse of privilege by an authorised user of the TOE who was accessing the directory (which holds the TOE's user authentication and application authorization data). O.I&A.TOE provides support by reliably identifying the user responsible for particular events, thus ensuring that the user can be held accountable for actions for which he or she is responsible. O.ACCESS.CONTROL and O.ADMIN.TOE provide support by controlling access to audit configuration data which only highly trusted individuals must be allowed to view and modify. O.AUDITLOG ensures that audit data is correctly managed by the administrator so that it can be used to detect users abusing their privileges.

T.OPERATE Rationale

T.OPERATE (*Insecure Operation)* is countered directly by O.INSTALL, which ensures that the TOE and its underlying platform are correctly installed, managed and operated.

T.PHYSICAL Rationale

T.PHYSICAL (*Physical Attack)* is countered directly by O.PHYSICAL, which protects critical parts of the TOE from physical attack.

T.CRASH Rationale

T.CRASH (*Abrupt Interruptions)* is countered by O.MEDIA and O.RECOVERY. These ensure that suitable recovery mechanisms are in place to recover from a crash and that the media used during the crash recovery is able to maintain the confidentiality, integrity and availability of the TOE.

P.ACCESS Rationale

P.ACCESS is satisfied by O.ACCESS.OBJECTS, O.ACCESS.CONTROL, O.ADMIN.TOE & O.ADMIN. ENV, O.TRUST, O.AUTHDATA, O.FILES and

O.SEP. O.ACCESS.OBJECTS ensures that the subjects using the TOE are able to control access to the objects for which they are responsible. O.ADMIN.TOE and O.ADMIN.ENV ensure that only authorised administrators can effectively manage the TOE and its Security Functions. O.FILES, O.TRUST, O.AUTHDATA and O.SEP ensures that components of the TSF cannot be tampered with by unauthorised access.

Assumptions Rationale

This section demonstrates how the security objectives map to the TOE secure usage assumptions.

A.TOE.CONFIG is directly provided by O.INSTALL part a) and O.LOCKOUT because [ECD] is part of the operational documentation of the TOE.

A.SYS.CONFIG is directly provided by O.INSTALL part b).

A.PHYSICAL is directly provided by O.PHYSICAL.

A.APPLICATION is directly provided by O.AUDITLOG part d).

A.ACCESS is provided by O.INSTALL, O.LOCKOUT, O.TRUST, O.AUTHDATA and O.ADMIN.ENV.

A.MANAGE is provided by O.TRUST, supported by O.INSTALL, O.LOCKOUT, O.AUDITLOG, O.AUTHDATA, O.MEDIA, O.ADMIN.ENV, O.FILES, O.RECOVERY and O.SEP.

A.PEER is provided by O.PHYSICAL and O.INSTALL. Since connected systems will require a physical connection to the TOE to be established they fall into the scope of O.PHYSICAL.

A.NETWORK is directly provided by O.AUTHDATA, because the network is used to transport authentication data.

Security Requirements Rationale

Suitability of Security Requirements

The table below correlates the IT security objectives to the SFRs which satisfy them (as indicated by a *YES*), showing that each IT security objective is satisfied by at least one SFR, and that each SFR satisfies at least one IT security objective.

Table 6: Correlation of IT Security Objectives to Security Functional Requirements

Requirement	O.I&A.TOE	O.ACCESS	O.AUDIT	O.ADMIN. TOE	O.APPLICATION
FAU_GEN.1T			YES		
FAU_GEN.2			YES		
FAU_SAR.1			YES		
FAU_SAR.3			YES		
FAU_SEL.1			YES		
FAU_STG.1			YES	YES	

Table 6: Correlation of IT Security Objectives to Security Functional Requirements

Requirement	O.I&A.TOE	O.ACCESS	O.AUDIT	O.ADMIN. TOE	O.APPLICATION
FAU_STG.4			YES	YES	
FDP_ACC.1		YES			YES
FDP_ACF.1		YES			YES
FDP_RIP.2		YES			
FIA_AFL.1	YES				
FIA_ATD.1	YES	YES	YES	YES	
FIA_SOS.1	YES				
FIA_UAU.1	YES				
FIA_UID.1	YES	YES			
FMT_MSA.1	YES	YES		YES	YES
FMT_MSA.3		YES			YES
FMT_MTD.1			YES	YES	YES
FMT_REV.1		YES			YES
FMT_SMF.1	YES	YES	YES	YES	YES
FMT_SMR.1				YES	
FPT_RVM.1		YES			
FPT_SEP.1T		YES			
FTA_TSE.1	YES				YES

O.I&A.TOE Suitability

O.I&A.TOE is directly provided by FIA_UID.1 and FIA_UAU.1, which provide the means of identifying and authenticating users of the TOE. FIA_AFL.1 performs certain actions if a specified number of consecutive unsuccessful authentication attempts is made. FIA_ATD.1 provides a unique set of user attributes for each user while FMT_MSA.1 and FMT_SMF.1 specify controls over the modification of these attributes. FIA_SOS.1 provides for quality metrics to be applied when new passwords are chosen. FTA_TSE.1 controls the ability to create a user repository session by a user.

O.ACCESS Suitability

O.ACCESS is directly provided by FDP_ACC.1 which defines the access control policy and FDP_ACF.1 which specifies the access control rules. FMT_REV.1 enforces revocation of security attributes. FDP_RIP.2 ensures prevention of access to information residing in reused storage objects when they are reallocated to another subject. FIA_ATD.1 ensures the security attributes of a user are bound to subjects

created to act on his or her behalf. FIA_UID.1 ensures users are identified prior to any TSF-mediated access actions. FPT_RVM.1 ensures that the traditional reference monitor is always invoked prior to access. FMT_MSA.1, FMT_MSA.3 and FMT_SMF.1 provide support for the management of security attributes to control access to directory objects. FPT_SEP.1T assures that objects one subject are accessing cannot be intentionally or inadvertently accessed by another subject without a TSF access decision being made for the second subject.

O.APPLICATION Suitability

O.APPLICATION is directly provided by FTA_TSE.1 which specifies when a session with an application hosted by OC4J can be denied. FDP_ACC.1 and FDP_ACF.1 enforce access control on the group entries which contain some of the information about permissions for users to access an OC4J hosted application. FMT_MSA.1, FMT_MSA.3, FMT_REV.1 and FMT_SMF.1 provide support for the management of security attributes to control access to directory objects.

O.AUDIT Suitability

O.AUDIT is directly provided by FAU_GEN.1T which generates audit records for all security relevant directory events. FAU_GEN.2 supports the enforcement of individual accountability by ensuring the user responsible for each event can be identified. FIA_ATD.1 provides for the storage of user security attributes. FAU_STG.1 provides permanent storage for the audit trail, FAU_STG.4 provides for mechanisms to deal with full audit trails, while FMT_MTD.1.1.1, FMT_MTD.1.1.2 and FMT_SMF.1 provide for the protection of that audit trail. FAU_SAR.1 and FAU_SAR.3 provide functions to review the contents of the audit trail, while FAU_SEL.1 provides the ability to select which events are to be audited. FMT_MTD.1 provides the ability to manage the set of audited events.

O.ADMIN.TOE Suitability

O.ADMIN.TOE is directly provided by FMT_SMR.1, FMT_SMF.1 and FMT_MTD.1, which provide essential administrative functionality which is restricted to authorised administrators. FIA_ATD.1 provides support by ensuring that the security attributes of users are associated with the subjects acting on the user's behalf. FMT_MSA.1 and FMT.SMF.1 provide administrative functionality which enforces the User Repository Access Control Security Function Policy to restrict the ability to modify, delete and create security attributes. FAU_STG.1 provides support by preventing unauthorised modification or deletion of audit records. FAU_STG.4 provides support by handling auditable events when the audit log is full.

The rationale above demonstrates the suitability of the TOE security requirements.

Suitability of Security Requirements for the IT Environment

The Security Requirements for the IT Environment section of Chapter 5 defines a set of SFRs for the IT environment to support the TOE SFRs, and also provides an informal description of requirements for the IT environment to support the security objectives. Most of these requirements are described informally in order not to unduly limit the environments that can satisfy them. The Support for Security Objectives section in Chapter 5 gives a rationale as to why these requirements are needed. The requirements for the IT environment that are described in the Support for Security Objectives section are together sufficient to meet the objectives for the IT environment defined in Chapter 4 (which are O.ADMIN.ENV, O.FILES and O.SEP).

The functional security requirements for the IT environment (defined in the Support for SFRs section in Chapter 5) are traced to security objectives for the environment as follows:

- FAU_GEN.1E.1, FAU_GEN.1E.2 and FPT_STM.1.1 map to O.ADMIN.ENV which includes requirements on the underlying database system and operating system for the support of auditing functions. O.ADMIN.ENV's requirements for auditing correspond to the requirements defined in FAU_GEN.1E.1, FAU_GEN.1E.2 and FPT_STM.1.1.
- FPT_SEP.1E.1 maps to O.SEP, which covers requirements for the operating system to provide separation features to protect the TOE.

Dependency Analysis

The table on the next page demonstrates that all dependencies of functional components are satisfied.

Table 7: Functional Component Dependency Analysis

Component Reference	Component	Dependencies	Dependency Reference
1	**FAU_GEN.1T**	FPT_STM.1	See notes 1, 2 and 3 below
2	**FAU_GEN.2**	FAU_GEN.1T FIA_UID.1	1 15
3	**FAU_SAR.1**	FAU_GEN.1T	1
4	**FAU_SAR.3**	FAU_SAR.1	3
5	**FAU_SEL.1**	FAU_GEN.1T FMT_MTD.1.1.2	1 18
6	**FAU_STG.1**	FAU_GEN.1T	1
7	**FAU_STG.4**	FAU_STG.1	6
8	**FDP_ACC.1**	FDP_ACF.1	9
9	**FDP_ACF.1**	FDP_ACC.1 FMT_MSA.3	8 17
10	**FDP_RIP.2**	-	-
11	**FIA_AFL.1**	FIA_UAU.1	14
12	**FIA_ATD.1**	-	-
13	**FIA_SOS.1**	-	-
14	**FIA_UAU.1**	FIA_UID.1	15
15	**FIA_UID.1**	-	-

Table 7: Functional Component Dependency Analysis

Component Reference	Component	Dependencies	Dependency Reference
16	**FMT_MSA.1**	FDP_ACC.1 FMT_SMF.1 FMT_SMR.1	8 20 21
17	**FMT_MSA.3**	FMT_MSA.1 FMT_SMR.1	16 21
18	**FMT_MTD.***	FMT_SMF.1 FMT_SMR.1	20 21 See note 4 below
19	**FMT_REV.1**	FMT_SMR.1	21
20	**FMT_SMF.1**	-	-
21	**FMT_SMR.1**	FIA_UID.1	15
22	**FPT_RVM.1**	-	-
23	**FPT_SEP.1T**	-	- See note 2 below
24	**FTA_TSE.1**	-	-

Note 1: The security requirement for the IT environment FPT_STM.1.1 satisfies the dependency of the SFR FAU_GEN.1T.2 for the provision of reliable timestamps (see the section on security requirements for the IT environment in chapter 5).

Note 2: The nature of the extensions does not impact on the dependencies as defined for the CC Part 2 components from which they are derived.

Note 3: The modification of FAU_GEN.1 does not impact its ability to satisfy the dependencies of FAU_GEN.2, FAU_SAR.1, FAU_SEL.1 and FAU_STG.1 - especially given that collectively the TOE and IT environment meet FAU_GEN.1.

Note 4: FMT_MTD.1 has 2 iterations. Its entry in the table above indicates that all FMT_MTD.1 dependencies are satisfied by FMT_SMF.1 and FMT_SMR.1.

Dependency analysis of the security assurance requirements

EAL4 is a self-contained assurance package and ALC_FLR.3 has no dependencies on any other component.

Demonstration of Mutual Support

The dependency analysis provided in the table above demonstrates mutual support between functional components, showing that all dependencies required by Part 2 of the CC are satisfied.

The following supportive dependencies exist for the TOE to prevent bypassing of and tampering with the SFRs:

FIA_UID.1 and FIA_UAU.1 together with FIA_ATD.1 and FMT_MSA.1 provide support to all SFRs which rely on the identification of individual users and their security attributes, namely: FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_SMR.1, FAU_GEN.1, FAU_GEN.2, FMT_MTD.1, FMT_SMF.1, FAU_SAR.1 and FAU_SEL.1.

FDP_RIP.2 supports FDP_ACC.1 and FDP_ACF.1 by preventing the bypassing of those SFRs through access to reused to storage objects.

FMT_MSA.3 provides support to FDP_ACC.1 and FDP_ACF.1 by ensuring objects are protected by default when newly created.

FMT_MSA.1 provides support to FDP_ACC.1, FDP_ACF.1 and FMT_SMF.1 by controlling the modification of object security attributes.

FMT_REV.1 provides support to FMT_MSA.1, FDP_ACC.1 and FDP_ACF.1 by enforcing revocation of object security attributes.

FAU_STG.1 and FAU_STG.4 support FAU_GEN.1T by providing permanent storage for the audit trail, and dealing with the audit trail full condition.

FMT_MTD.1 supports FAU_STG.1, FAU_STG.4 and FMT_SMF.1 by protecting the integrity of the audit trail.

FAU_SEL.1 supports FAU_STG.1 by providing the means of limiting the events to be audited, thereby ensuring that the available space for the audit trail is not exhausted more frequently than necessary.

FPT_RVM.1 and FPT_SEP.1T support FDP_ACC.1 and FDP_ACF.1 by restricting access to residual data and providing separate domains.

FDP.ACC.1 and FDP.ACF.1 support FAU_STG.1 and FMT_SMF.1 by preventing unauthorised modifications to the audit trail. They also support FMT_MSA.1 by preventing unauthorised modifications of directory objects' security attributes and they protect the TSF data from unauthorised modification to support FMT_MTD.1.

Strength of Function Validity

The user password mechanism is the only TOE mechanism that is probabilistic or permutational, and has a strength of SOF-*high*. This strength of function is intended to provide enough protection against straight forward or intentional attack from threat agents having a high attack potential.

Assurance Requirements Appropriate

The target assurance level is EAL4, augmented with ALC_FLR.3. EAL4 is appropriate because the TOE is designed for use with an underlying operating system and database server that have been assured to EAL4.

ALC_FLR.3 has been included in addition to EAL4 to cause the evaluation of the TOE's flaw remediation procedures which TOE users need to be in place following the release of the TOE. These procedures are required to offer continuing assurance to users that the TOE provides secure access to the applications and data which are crucial to their enterprise's success.

To meet this requirement, the flaw remediation procedures must offer:

- the ability for TOE users to report potential security flaws to Oracle,
- the resolution and correction of any flaws with assurance that the corrections introduce no new security flaws, and

- the timely distribution of corrective actions to users.

ALC_FLR.3 is the ALC_FLR component which is at an appropriate level of rigour to cover these requirements.

TOE Summary Specification Rationale

This section demonstrates that the TOE Security Functions and Assurance Measures are suitable to meet the TOE security requirements.

TOE Security Functions Satisfy Requirements

The table below demonstrates that for each SFR the TOE security functions are suitable to meet the SFR, and the combination of TOE security functions work together so as to satisfy the SFR:

Table 8: TOE Security Function Suitability and Binding

SFR	TOE Security Functions	Rationale
FIA_AFL.1.1	IA.PWDC IA.USESS IA.AUTH	IA.PWDC provides the configurable control governing the number of failed userpassword check attempts before a user account is locked. IA.USESS and IA.AUTH detect if this number is reached or exceeded.
FIA_AFL.1.2	IA.PWDC IA.USESS IA.AUTH	IA.PWDC provides the configurable control governing the number of failed userpassword check attempts before a user account is locked. IA.USESS and IA.AUTH lock the user's account if this number has been reached or exceeded.
FIA_ATD.1.1	SAM.UATT APP.PERMS	SAM.UATT ensures that the user repository contains the name, group membership and password security attributes for each user repository user. APP.PERMS ensures the user repository and the TOE configuration files can contain information about permissions for users to access applications. Such information includes user membership of groups which are mapped via configuration files to security roles that have access to applications.
FIA_SOS.1.1	IA.PWDC SAM.CHPWD	IA.PWDC specifies the configurable metrics user passwords have to meet. SAM.CHPWD allows users to change their own passwords within the configured metrics.
FIA_UAU.1.1	IA.ASESS IA.IDE	If a user does not authenticate, IA.ASESS will allow an anonymous user repository session in which only publicly available material is accessible. IA.IDE states that a user repository session must be established in order to submit a request to and receive information from a user repository's directory server.

Table 8: TOE Security Function Suitability and Binding

SFR	TOE Security Functions	Rationale
FIA_UAU.1.2	IA.USESS IA.AUTH IA.ASESS IA.IDE	IA.IDE states that a user repository session must be established in order to submit a request to and receive information from a user repository's directory server. IA.USESS, IA.AUTH and IA.ASESS state the conditions for being able to establish a session. Authentication is required in order to perform TSF-mediated actions other than accessing publicly available material.
FIA_UID.1.1	IA.ASESS IA.IDE	If the user supplies no username, IA.ASESS will allow an anonymous user repository session in which only publicly available material is accessible. IA.IDE states that a user repository session must be established in order to submit a request to and receive information from a user repository's directory server.
FIA_UID.1.2	IA.USESS IA.AUTH IA.ASESS IA.UID IA.PSESS IA.IDE	IA.IDE states that a user repository session must be established in order to submit a request to and receive information from a user repository's directory server. If the user supplies no username, IA.ASESS will allow an anonymous user repository session in which only publicly available material is accessible. IA.UID, IA.USESS, IA.AUTH and IA.PSESS state the conditions for being able to establish a session with an identified user. Identification is required in order to perform TSF-mediated actions other than accessing publicly available material.
FDP_ACC.1.1	IA.UID RAC.OBID RAC.OBREF RAC.SUA RAC.OBA	IA.UID enforces that each user is uniquely identified. RAC.OBID and RAC.OBREF ensures that all objects (which are subject to user repository access control) can be uniquely identified. RAC.SUA and RAC.OBA state that the user repository access control policy for access operations extends to all subjects and objects.
FDP_ACF.1.1	IA.UID RAC.OBID RAC.OBREF RAC.SUA RAC.OBA RAC.POL SAM.UATT SAM.EATT SAM.UEFF SAM.OEFF	IA.UID ensures each user has a unique user identity. RAC.OBID and RAC.OBREF ensure that all objects (which are subject to user repository access control) can be uniquely identified. RAC.SUA and RAC.OBA state that the RAC policy for access operations extends to all subjects and objects. RAC.POL is a statement of the user repository access control policy. SAM.UATT and SAM.EATT cover the security attributes for users and user repository objects (including the identity and group memberships for users and access control information for objects). SAM.OEFF and SAM.UEFF state the conditions under which the user and object security attributes are effective for a user repository session.
FDP_ACF.1.2	IA.UID IA.CRUG RAC.OBID RAC.OBREF RAC.POL SAM.UEFF SAM.OEFF	RAC.POL fully covers the access control rules defined in FDP_ACF.1.2. RAC.OBID and RAC.OBREF ensure that all objects (which are subject to user repository access control) can be uniquely identified. IA.CRUG is relevant as I&A data is subject to the user repository access control policy. IA.UID ensures users are uniquely identified via a DN name. SAM.OEFF and SAM.UEFF state the conditions under which the user and object security attributes are effective for a user repository session.

Table 8: TOE Security Function Suitability and Binding

SFR	TOE Security Functions	Rationale
FDP_ACF.1.3	RAC.POL SAM.UEFF SAM.OEFF	RAC.POL states that the super user for the directory is the administrative user for the user repository and therefore has access to all user repository objects. SAM.OEFF and SAM.UEFF state the conditions under which the user and object security attributes are effective for a user repository session.
FDP_ACF.1.4	N/A	This SFR does not mandate any functionality. It is included for compliance with the CC.
FDP_RIP.2.1	RAC.OR	RAC.OR satisfies FDP_RIP.2.1 directly.
FMT_MSA.1.1	IA.CRUG SAM.MODATT SAM.CHPWD	IA.CRUG only allows authorised users to create user entries and group memberships in the user repository. SAM.MODATT ensures an authorised user can create, read, modify or delete security attributes for user repository users and user repository entries. SAM.CHPWD allows authorised users to change user password attributes.
FMT_MSA.3.1	RAC.POL SAM.EATT IA.CRUG	SAM.EATT states that default security attributes for user repository entries are as per the Default Access Policies defined for the TOE. RAC.POL defines the access control policy. The user security attributes used to enforce repository access control are user identity and group memberships. IA.CRUG states that the default values of attributes of newly created user entries and group entries are as described in Chapters 6, 7 and 9 of [OIDAG].
FMT_MSA.3.2	RAC.OBA RAC.POL SAM.EATT	Unless access to an object has been explicitly granted, as described in RAC.OBA and RAC.POL, no access will be allowed. SAM.EATT states that default security attributes for user repository entries are as per the Default Access Policies defined for the TOE, which cover how authorised users can reset the default security configuration.
FMT_MTD.1.1.1	AUD.ACC AUD.DEL	AUD.DEL states that only an authorised administrator can clear the audit trail. AUD.ACC will allow only authorised users to view records in the audit log.
FMT_MTD.1.1.2	AUD.SET	AUD.SET allows only the super user to set the audit level that specifies which events are auditable.
FMT_REV.1.1	SAM.MODATT	SAM.MODATT ensures that only a suitably authorised user can create, read, modify or delete security attributes for user repository users and user repository entries and hence to effectively revoke such attributes.
FMT_REV.1.2	SAM.OEFF SAM.UEFF	SAM.OEFF and SAM.UEFF state the conditions under which the user and object security attributes are effective for a user repository session and hence when a change to revoke such attributes becomes effective.

Table 8: TOE Security Function Suitability and Binding

SFR	TOE Security Functions	Rationale
FMT_SMF.1.1	AUD.ACC AUD.DEL AUD.SET RAC.POL IA.CRUG IA.PWDC IA.PWDCM SAM.CHPWD SAM.MODATT	IA.CRUG allows only authorised users to create user entries and group memberships in the user repository. SAM.MODATT and RAC.POL ensures an authorised user can create, read, modify or delete security attributes for user repository users and user repository entries. SAM.CHPWD allows authorised users to change user password attributes. AUD.DEL states that only an authorised administrator can clear the audit trail. AUD.ACC will allow only authorised users to view records in the audit log. AUD.SET allows only the super user to access the audit level that specifies which events are auditable. IA.PWDC and IA.PWDCM ensure that the configurable controls on user passwords (i.e. reuse, lifetime and content metrics for passwords) can only be accessed and updated by suitably authorised users.
FMT_SMR.1.1	IA.UID	IA.UID ensures that the TSF maintains the roles of normal user and super user and states that the super user is the administrator for the user repository.
FMT_SMR.1.2	IA.UID	IA.UID states how user repository entries for normal users and the super user are identified.
FPT_RVM.1.1	IA.IDE RAC.POL	IA.IDE ensures that the TOE always knows who the current user is. RAC.POL ensures that the user repository access control policy enforcement functions are always invoked for this user before an access operation can proceed.
FPT_SEP.1T.1	RAC.OIDSEP RAC.OC4JSEP	RAC.OIDSEP ensures that the interactions between different users and the OID product cannot interfere with each other. RAC.OC4JSEP ensures that the interactions between different users and the OC4J product cannot interfere with each other. Additionally there is no way to access the TOE except through the evaluated interfaces described by the TOE security functions.
FPT_SEP.1T.2	IA.IDE RAC.OIDSEP RAC.OC4JSEP	IA.IDE ensures that the identity of the user associated with each interaction with the TOE is clear. RAC.OIDSEP ensures that the interactions between different users and the OID product cannot interfere with each other. RAC.OC4JSEP ensures that the interactions between different users and the OC4J product cannot interfere with each other.
FTA_TSE.1.1.1	IA.PWDC IA.USESS IA.AUTH	IA.PWDC provides the configurable control governing the expiration of a password. IA.USESS and IA.AUTH prevent the establishment of a session if the user's password has expired.

Table 8: TOE Security Function Suitability and Binding

SFR	TOE Security Functions	Rationale
FTA_TSE.1.1 .2	APP.PERMS APP.APPREF APP.ACCESS	APP.PERMS ensures the user repository and the TOE configuration files can contain information about permissions for users to access applications. Such information includes user membership of groups which are mapped via configuration files to security roles that have access to applications. APP.APPREF ensures the TOE identifies the correct Web application referenced from the permissions information. APP.ACCESS prevents the user establishing a session with the application unless the user has the necessary permission.
FAU_GEN.1T .1	AUD.INF AUD.SET	AUD.SET allows the super user to set which events are to be auditable. Audit records are generated to contain information as defined by AUD.INF.
FAU_GEN.1T .2	AUD.INF	Audit records are generated to contain the required information as defined by AUD.INF.
FAU_GEN.2.1	IA.UID AUD.INF	IA.UID ensures that each user is uniquely identified. Audit records are generated to contain the required user identity information as defined by AUD.INF.
FAU_SAR.1.1	AUD.ACC	AUD.ACC directly satisfies FAU_SAR.1.1
FAU_SAR.1.2	AUD.ACC	AUD.ACC directly satisfies FAU_SAR.1.2
FAU_SAR.3.1	AUD.ACC	AUD.ACC provides facilities for searching for audit records according to their attribute values.
FAU_SEL.1.1	AUD.SET	AUD.SET directly satisfies FAU_SEL.1.1.
FAU_STG.1.1	AUD.DEL	AUD.DEL directly satisfies FAU_STG.1.1.
FAU_STG.1.2	AUD.DEL	F.AUD.DEL protects audit records from modification.
FAU_STG.4.1	AUD.FULL	AUD.FULL directly satisfies FAU_STG.4.1

The table below shows that all the SFRs are satisfied by at least one SF and that every SF is used to satisfy at least one SFR (but note that SFR FDP_ACF.1.4 is not explicitly satisfied by any particular SF because this SFR specifies null functionality).

Table 9: Mapping of SFs to SFRs

	FIA								FDP						FMT										FPT			FTA	FTA	FAU									
	AFL.1.1	AFL.1.2	ATD.1.1	SOS.1.1	UAU.1.1	UAU.1.2	UID.1.1	UID.1.2	ACC.1.1	ACF.1.1	ACF.1.2	ACF.1.3	ACF.1.4	RIP.2.1	MSA.1.1	MSA.3.1	MSA.3.2	MTD.1.1.1	MTD.1.1.2	REV.1.1	REV.1.2	SMF.1.1	SMR.1.1	SMR.1.2	RVM.1.1	SEP.1T.1	SEP.1T.2	TSE.1.1.1	TSE.1.1.2	GEN.1T.1	GEN.1T.2	GEN.2.1	SAR.1.1	SAR.1.2	SAR.3.1	SEL.1.1	STG.1.1	STG.1.2	STG.4.1
IA.UID								Y	Y	Y	Y												Y	Y								Y							
IA.AUTH	Y	Y				Y		Y																				Y										
IA.ASESS					Y	Y	Y	Y																															
IA.USESS	Y	Y				Y		Y																				Y										
IA.IDE					Y	Y	Y	Y																	Y		Y												
IA.PSESS								Y																														
IA.CRUG											Y				Y	Y						Y																
IA.PWDC	Y	Y		Y																		Y						Y										
IA.PWDCM																						Y																
APP.PERMS			Y																										Y									
APP.APPREF																													Y									
APP.ACCESS																													Y									
SAM.UATT			Y							Y																												
SAM.EATT										Y						Y	Y																					
SAM.CHPWD				Y											Y							Y																
SAM.MODATT															Y					Y		Y																
SAM.UEFF										Y	Y	Y									Y																	
SAM.OEFF										Y	Y	Y									Y																	
RAC.OBID									Y	Y	Y																											
RAC.OBREF									Y	Y	Y																											
RAC.SUA									Y	Y																												
RAC.OBA									Y	Y							Y																					
RAC.POL										Y	Y	Y				Y	Y					Y			Y													
RAC.OIDSEP																										Y	Y											
RAC.OC4JSEP																										Y	Y											
RAC.OR														Y																								
AUD.INF																														Y	Y	Y						
AUD.SET																			Y			Y								Y					Y			
AUD.ACC																		Y				Y											Y	Y	Y			
AUD.DEL																		Y				Y														Y	Y	
AUD.FULL																																						Y

Assurance Measures Rationale

Table 3 in chapter 6 shows that, for each Security Assurance Requirement, there is an appropriate assurance measure.

PP Claims Rationale

This security target makes no claims about Protection Profile conformance.

ANNEX

A References

[CAPP] *Controlled Access Protection Profile,*
Version 1.d, NSA, October 1999.

[CC] *Common Criteria for Information Technology Security Evaluation,*
Version 2.2, ISO/IEC 15408, CCIMB-2004-01-001, January 2004.

[CRP178] *Common Criteria Certification Report No. P178*
Oracle9i Database Enterprise Edition Release 2 (9.2.0.1.0),
Issue 1.0, UK IT Evaluation and Certification Scheme, September 2003.

[CRP182] *Common Criteria Certification Report No. P182*
Sun Solaris Version 8 2/02,
Issue 1.0, UK IT Evaluation and Certification Scheme, April 2003.

[DPP] *Database Management System Protection Profile (DBMS PP),*
Issue 2.1, Oracle Corporation, May 2000.

[ECD] *Evaluated Configuration for Oracle Application Server 10g (9.0.4),*
Oracle Corporation.

[LDAP3] *Lightweight Directory Access Protocol Version 3,*
Request For Comments (RFC) 2251 of the Internet Engineering Task Force, December 1997,
available on the World Wide Web at http://www.ietf.org/rfc.htm

[OASC] *Oracle Application Server 10g Concepts 10g (9.0.4),*
Part Number B10375-01, Oracle Corporation, September 2003.

[OC4JSG] *Oracle Application Server Containers for J2EE Security Guide 10g (9.0.4),*
Part Number B10325-01, Oracle Corporation, September 2003.

[OC4JUG] *Oracle Application Server Containers for J2EE User's Guide 10g (9.0.4),*
Part Number B10322-01, Oracle Corporation, September 2003.

[OIDAG] *Oracle Internet Directory Administrator's Guide 10g (9.0.4),* Part Number B12118-01, Oracle Corporation, September 2003.

[OID_ST] *Security Target for Oracle Internet Directory 10g (9.0.4),* Issue 0.9, Oracle Corporation, August 2004.

[TCSEC] *Trusted Computer Security Evaluation Criteria,* Department of Defense, United States of America, DoD 5200.28-STD, December 1985.

ANNEX

B Glossary

Acronyms

ACI	Access Control Item
ACL	Access Control List
ACP	Access Control Policy Point
ASN.1	Abstract Syntax Notation One
AVL	Adelson, Velskii and Landis (a type of binary tree)
BER	Basic Encoding Rules (for ASN.1)
DAC	Directory Access Control
DIB	Directory Information Base
DIT	Directory Information Tree
DN	Distinguished Name
DSE	Directory-Specific Entry
EJB	Enterprise Java Beans
J2EE	Java 2 Platform, Enterprise Edition
JAAS	Java Authentication and Authorization Service

JNDI	Java Naming and Directory Interface
JSP	JavaServer Pages
JVM	Java Virtual Machine
LDAP	Lightweight Directory Access Protocol
LDIF	LDAP Data Interchange Format
OCI	Oracle Call Interface
OC4J	Oracle Application Server Containers for J2EE
OID	Oracle Internet Directory
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement
SOF	Strength of Function
TOE	Target Of Evaluation
TSC	TOE Scope of Control
TSF	TOE Security Functions
TSFI	TSF Interface
TSP	TOE Security Policy
XML	eXtensible Markup Language (XML) is a set of rules for defining data markup in a plain text format.

Terms

If a term described below has [CC] or [TCSEC] written after it, then this term is defined in the IT security evaluation scheme. All other terms relate to Oracle Application Server Containers for J2EE (OC4J) or Oracle Internet Directory (OID). [OASC, Glossary] and [OIDAG, Glossary] cover the full set of terms for OC4J and OID. The terms used in this document are described below.

Access Control Group

A group entry in OID contains a list of names. A user is a member of the group if the user's DN is held in the group entry's multi-valued attribute `member` or `uniqueMember`. There are two types of access control groups: ACP groups and privilege groups.

Access Control Item (ACI)	The OID directory holds access control information to define the administrative policies relating to access control. This information is stored as user-modifiable operational attributes called access control items (ACIs).
Access Control List (ACL)	A list of Access Control Items is called an Access Control List (ACL).
Access Control Policy Point (ACP)	An Access Control Policy Point (ACP) is an entry for which the `orclACI` attribute has been given a value. The `orclACI` attribute contains ACL directives that are prescriptive. That is, these directives apply to all entries in the subtree below the ACP where this attribute is defined.
ACP Group	If an individual is a member of an ACP group, then the directory server grants to that individual the privileges associated with that ACP group.
Application Program Interface (API)	An application program interface (API) is a set of exposed data structures and functions that an application can use to invoke services on a component.
Attribute	Each entry in a directory contains information stored in attributes.
Audit Log	The OID audit log is made up of directory entries, where each entry records the audit data for one event.
Audit Level	To enable auditing, the attribute `orclauditlevel` in the DSE must be modified to the appropriate level. The value held in this attribute is called the directory's audit level.
Authentication	Authentication is the process by which the true identity of a user connecting to the TOE is validated. In the TOE's evaluated configuration, OID implements three different levels of directory user authentication: Anonymous, Password-based (Simple Authentication), and Indirect Authentication. OC4J refers to password-based authentication as Basic Authentication.
Authorization	Authorization is the evaluation of security constraints to send a message or make a request. Authorization uses specific criteria to determine whether the request should be permitted. The criteria are authentication and restriction. OC4J can perform two sorts of authorization checks: J2EE authorization and JAAS authorization. J2EE authorization concerns a user's permission to access a J2EE application. JAAS authorization concerns a user's permission to perform an action on a resource after the J2EE application has been entered. Only J2EE authorization is within the scope of this evaluation of Oracle Application Server.
AVL Tree	A binary tree representation that can be used for the entries in a directory.
Basic Authentication	Basic authentication is an authentication scheme based on the use of a username and password, in which passwords are not encrypted for sending over the network.
Binding	The process of authenticating a user to a directory.
Container	A container is a component that contains other components such as a servlet. A container executes and manages a servlet. A container is either part of or associated with and used by a Web server. When a client HTTP request calls a servlet, the Web server passes the HTTP request to the container. The container translates the

	HTTP request into a Java method invocation and then passes the request to the servlet.
Directory	A directory stores and retrieves information about organisations, individuals and other resources.
Directory Information Base (DIB)	The complete set of all information held in a directory. The DIB consists of entries that are related to each other hierarchically in a directory information tree.
Directory Information Tree (DIT)	A hierarchical tree-like structure consisting of the DNs of the entries.
Directory Server Instance	Each Oracle Directory Server instance services directory requests through a single OID dispatcher process listening at a specific TCP/IP port number. There can be more than one directory server instance on a node, each listening on a different port.
Directory Access Control (DAC)	Access control on directory objects based on access control information established by directory users.
Distinguished Name (DN)	Each entry in a directory is uniquely identified by a distinguished name, which defines exactly where in the directory's hierarchy the entry resides. It comprises all of the individual names of the parent entries back to the root.
Enterprise Java Beans (EJB)	Enterprise Java Beans (EJB) are the component-based application model for Java defined by JavaSoft. This model provides most of the system-level services, such as multi-threading, to ease application programming
Entry	In a directory, a collection of information about an object is called an entry.
Entry Level Access Control	The `orclEntryLevelACI` attribute is used for entry level access control, for which the policy pertains only to a specific entity.
JAAS	JAAS is a Java package that enables applications to authenticate and enforce access controls on users. The JAAS framework and the Java 2 Security Model form the foundation of JAAS. The use of JAAS enables an application to remain independent from the authentication service, and allows developers to avoid devoting resources to developing authentication, authorization, and delegation services.
JAAS Provider	A JAAS Provider is an implementation of the JAAS interface.
Java 2 Platform Enterprise Edition (J2EE)	Java 2 Platform, Enterprise Edition (J2EE) is a platform that enables application developers to develop, deploy and manage multitiered server-centric Web-based enterprise level applications.
Java Naming and Directory Interface (JNDI)	JNDI provides naming and directory functionality for Java applications. JNDI enables Java applications to access different, possibly multiple, naming and directory services using a single API.
JavaServer Pages (JSP)	JavaServer Pages are an extension to the servlet functionality that enables a simple programmatic interface to Web pages. JSPs are HTML pages with special tags and embedded Java code that is executed on the Web or application server, providing

	dynamic functionality to HTML pages.
Java Virtual Machine (JVM)	The Java Virtual Machine is part of the Java runtime environment responsible for interpreting Java bytecode. Java bytecode is executable by any JVM running on any machine.
JAZN Admintool	JAZN Admintool is a Java application which provides facilities for managing information about realms, roles and permissions that is held in entries in the user repository.
Knowledge Reference	A knowledge reference (or referral) allows a directory server to return a reference to another server as a result of a directory query.
LDAP Client	LDAP Clients send LDAP requests to an OID listener/dispatcher process listening for LDAP commands at its port.
LDAP Data Interchange Format (LDIF)	The set of standards for formatting an input file for any of the LDAP command-line utilities.
Lightweight Directory Access Protocol (LDAP)	The Lightweight Directory Access Protocol is a standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate. LDAP Version 3 is defined in [LDAP3].
mod_oc4j	When a user requests access to an application hosted by OC4J, `mod_oc4j`, a web server module, routes the request from the Oracle HTTP Server to OC4J.
Object	An entity within the TSC that contains or receives information and upon which subjects perform operations. Objects are visible through the TSFI and are composed of one or more TOE resources encapsulated with security attributes. [CC]
Object Class	An object class is a group of attributes that define the structure of a directory entry.
Oracle Call Interface (OCI)	Oracle Call Interface is an API or low-level tool for accessing Oracle databases and executing SQL and PL/SQL statements.
Oracle Internet Directory (OID)	Oracle Internet Directory is a general purpose directory service that enables fast retrieval and centralised management of information about dispersed users and network resources. LDAP V3 is used to communicate with it and OID is an Oracle9*i* database application.
Password Policy	A password policy is a set of rules about how passwords can be created, changed and used within Oracle Internet Directory.
Permission	The authorization process grants a user permission to do or to have something.
Platform	The combination of software and hardware underlying the TOE.
Principal	A *principal* is a specific identity, such as a user named `frank` or a role named `hr`. A principal is associated with a subject upon successful authentication to a computing service. Principals are instances of classes that implement the `java.security.Principal` interface. A principal class must define a namespace that contains a unique name for each instance of the class.

Privilege Group	A Privilege Group is a higher-level access control group. This is similar to an ACP group, but it also provides for additional checking beyond a single ACP. Thus, if the directory finds an ACP at a higher level in the DIT that grants the privilege group access to the requested object, then it overrides any denials by a subordinate ACP and grants the user access to the object.
Realm	For the TOE, a realm is associated with a set of entries in a user repository. These entries can contain information about users and roles or groups, including permissions for the users to access applications hosted by OC4J.
Referral	A referral (or knowledge reference) allows a directory server to return a reference to another server as a result of a directory query.
Role	A predefined set of rules establishing the allowed interactions between a user and the TOE. [CC]
Role	A named group of permissions that can be granted to users and other roles.
Security Attribute	Information associated with subjects, users, and/or objects which is used for the enforcement of the TSP. [CC]
Security Domain	The set of objects that a subject has the ability to access. [TCSEC]
Security Function (SF)	A part or parts of the TOE which have to be relied upon for enforcing a closely related subset of the rules from the TSP. [CC]
Security Function Policy (SFP)	The security policy enforced by a SF. [CC]
Security Functional Requirement (SFR)	A security functional requirement defined in a protection profile or security target. [CC]
SOF-high	A level of the TOE strength of function where analysis shows that the function provides adequate protection against a deliberately planned or organised breach of TOE security by attackers possessing a high attack potential. [CC]
Strength of Function (SOF)	A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behaviour by directly attacking its underlying security mechanisms. [CC]
Subject	An entity within the TSC that causes operations to be performed. [CC]
Subjects	A *subject* represents a grouping of related information for a single user of a computing service, such as a person, computer, or process. This related information includes the subject's identities and security-related attributes (such as passwords and cryptographic keys).[OC4JSG]
Suitably Authorised User	When a user is attempting to perform an operation on an object, a suitably authorised user is one who is permitted by the Directory Access Control SFP to perform the operation on the object.
super user	The super user is the administrator for the directory and has full access to all direc-

	tory information. The actual name and the password for the super user are held in the DSE (by default the super user's name is `orcladmin`)
System	A specific IT installation, with a particular purpose and operational environment [CC]
Target Of Evaluation (TOE)	The product or system being evaluated. [CC]
TOE resource	Anything usable or consumable in the TOE. [CC]
TOE Scope of Control (TSC)	The set of interactions which can occur with or within a TOE and are subject to the rules of the TSP. [CC]
TOE Security Functions (TSF)	A set consisting of all the software of the TOE that must be relied on for the correct enforcement of the TSP. [CC]
TOE Security Policy (TSP)	A set of rules that regulate how assets are managed, protected and distributed within a TOE. [CC]
TSF Interface (TSFI)	A set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which TOE resources are accessed, mediated by the TSF, or information is obtained from the TSF. [CC]
User	Any entity (human or machine) outside the TOE that interacts with the TOE. [CC]
Userpassword Check	This phrase refers to operations of the TOE which check whether the `userpassword` attribute of a user entry has a particular value. These are `bind` operations for which a password has been supplied and `compare` operations which are acting on the `userpassword` attribute of a user entry.
User Manager	OC4J employs a *user manager* to authenticate and authorize users that attempt to access a J2EE application.
User Repository	OC4J user credentials are stored in a *user repository*. In the evaluated configuration for the TOE, OC4J uses Oracle Internet Directory to manage its user repository data.

This Page Intentionally Blank