Oracle® Communications Unified Inventory Management

Security Guide

Release 8.0
G36727-01
October 2025

ORACLE

Oracle Communications Unified Inventory Management Security Guide, Release 8.0

G36727-01

Contents

About This Content

1 UIM Security Overview

Basic Security Considerations 1
Understanding the UIM Environment 1
Overview of UIM Security 2
Recommended Deployment Scenarios 3
Operating System Security 3
    Firewall Port Configuration 4
Oracle Database Security 4
    Data Encryption 4
    Secure Database Connections 4
        SSL Authentication 5
WebLogic Server Security 5
    Authorization 5
        WebLogic Resources 5
        Security Policies 5
    Secure Sockets Layer (SSL) 6
    Logging Security 6
Oracle Security Documentation 6
File Permissions 7

2 Performing a Secure UIM Installation

Installing UIM Securely 1
    About Password Policies 2
Post-Installation Configuration 2
    Setting Up User Accounts to Lock and Expire 2

3 Implementing UIM Security

Configuring and Using Authentication 1
    Java Authentication and Authorization Service 1

About Callback Handlers 2
Configuring and Using Access Control 2
Enabling Access for Specifications 2
Configuring and Using Security Audit Logs 3
Secure Access to UIM Web Services 3
Managing UIM Security 3

4 Security Considerations for Developers

About UIM Security Policies 1
About Securing UIM APIs 1
About Securing Entity Data 1
About Securing Web Services 1

A UIM Secure Deployment Checklist

Secure Deployment Checklist A-1

About This Content

This guide provides guidelines and recommendations for setting up Oracle Communications Unified Inventory Management (UIM) in a secure configuration.

Audience

This guide is intended for system administrators, database administrators, developers, and integrators, who work with UIM.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Conventions

The following text conventions are used in this document.

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

UIM Security Overview

This chapter provides an overview of Oracle Communications Unified Inventory Management (UIM) security.

Basic Security Considerations

The following principles are fundamental to using any application securely:

- **Keep software up to date.** This includes the latest product release and any patches that apply to it.
- **Limit privileges as much as possible.** Users should be given only the access necessary to perform their work. User privileges should be reviewed periodically to determine relevance to current work requirements.
- **Monitor system activity.** Establish who should access which system components, how often they should be accessed, and who should monitor those components.
- **Install software securely.** For example, use firewalls, secure protocols (such as SSL), and secure passwords. See "Performing a Secure UIM Installation" for more information.
- **Learn about and use UIM security features.** See "Implementing UIM Security" for more information.
- **Use secure development practices.** For example, take advantage of existing database security functionality instead of creating your own application security. See "Security Considerations for Developers" for more information.
- **Keep up to date on security information.** Oracle regularly issues security-related patch updates and security alerts. You must install all security patches as soon as possible. See “Critical Patch Updates and Security Alerts" on the Oracle website:

  https://www.oracle.com/security-alerts/

Understanding the UIM Environment

When planning your UIM implementation, consider the following:

- **Which resources must be protected?**

  For example:

  - You must protect customer data.
  - You must protect internal data, such as proprietary source code.
  - You must protect system components from being disabled by external attacks or intentional system overloads.

- **Who are you protecting data from?**

  For example, if your business has service subscribers, you must protect their data from other subscribers, but someone in your organization might have to access that data to manage it. You can analyze your workflows to determine who needs access to the data; for

example, a system administrator could manage your system components without needing to access the system data.

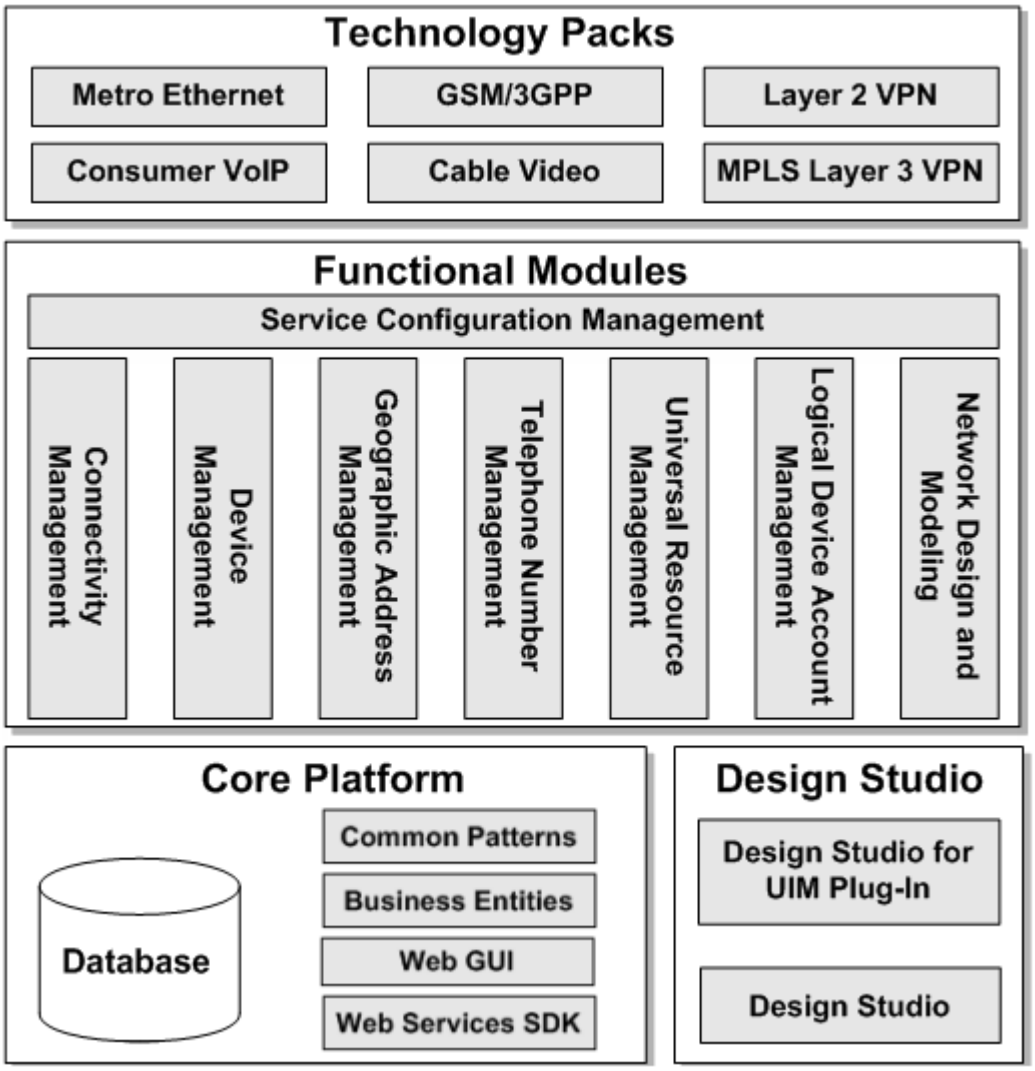
- **What happens if protections on strategic resources fail?**

  In some cases, a fault in your security scheme is nothing more than an inconvenience. In other cases, a fault might cause great damage to you or your customers. Understanding the security ramifications of each resource helps you protect it properly.

Overview of UIM Security

Figure 1-1 shows all the various components that can comprise UIM, including the components to which it connects. Each installed or integrated component requires special steps and configurations to ensure system security.

Figure 1-1 UIM Components



Recommended Deployment Scenarios

> **Note**
>
> For UIM cloud native deployments, see "Overview of the UIM Cloud Native Deployment" in *UIM Cloud Native Deployment Guide*.

Figure 1-2 shows a single server deployment scenario: the simplest UIM deployment architecture.

Figure 1-2 Single Sever Deployment

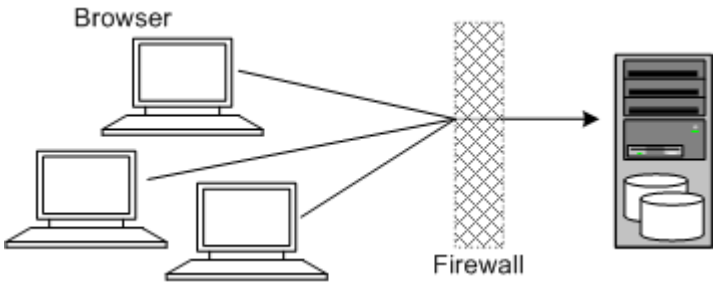


Figure 1-3 shows a clustered server deployment: a scalable UIM deployment offering greater security and high availability.

Figure 1-3 Clustered Server Deployment

Operating System Security

This section lists UIM-specific operating system security configurations. This section applies to all supported operating systems.

Firewall Port Configuration

> **Note**
>
> For UIM cloud native deployments, see *UIM Cloud Native Deployment Guide*.

UIM communicates through the firewall with various components on specific ports. Ensure that the operating system IPtables for the firewalls are configured to manage traffic on the following ports:

- Port 22 (optional, both directions): Used by the File Transfer and Parsing cartridge for SSH communication. Close this port if you are not using the File Transfer and Parsing cartridge.
- WebLogic Server SSL listen ports (both directions): Used by Administration and Managed servers for listening for traffic.
- Oracle Database listener ports: Used to listen for Oracle Database traffic.

Close all unused ports, especially non-SSL ports. Opt for SSL-enabled ports, instead of non-SSL ports, for all communications (for example: HTTPS, IIOPS, t3s).

For more information about securing your operating system, see your operating system documentation.

Oracle Database Security

This section lists the UIM-specific security configurations for the Oracle Database:

- Data Encryption
- Secure Database Connections

For more information about securing Oracle Database, see *Oracle Database Security Guide* and *Oracle Database Advanced Security Administrator's Guide*.

Data Encryption

If your database connection is not configured to use data encryption, data is sent across the network in a format that is designed for fast transmission and can be decoded by interceptors given some time and effort.

It is also possible (but not recommended) to encrypt the UIM tablespace and schema, at the expense of system performance. Encrypting the schema and tablespace is not necessary, because the database is sufficiently secure without the encryption.

See *Oracle Database Advanced Security Administrator's Guide* for more information.

Secure Database Connections

Encrypting network data is a critical security measure that ensures that data traveling over the network is difficult to intercept and access. Secure network connections to the Oracle Database using the Oracle Advanced Security feature. You can configure the Oracle Database with either Network Data Encryption or SSL authentication, as both ensure that the data is secure while traveling over the network.

The Oracle Advanced Security feature also provides security against the following types of attacks:

- Data modification attack, where an unauthorized party intercepts data in transit over the network, alters it, and transmits the altered data to the database.
- Replay attack, where an unauthorized party repeatedly transmits entire sets of valid data.

SSL Authentication

Use the Oracle Advanced Security feature to enable SSL authentication, using a digital certificate, on data that travels over the network to the database. See *Oracle Database Advanced Security Administrator's Guide* for more information.

Using SSL authentication allows UIM to communicate with servers over an encrypted connection and to communicate with the database over an encrypted connection.

SSL authentication supports the following authentication modes:

- Only the server authenticates itself to the client.
- Both client and server authenticate themselves to each other.
- Neither the client nor the server authenticate with each other (SSL encryption feature by itself).

WebLogic Server Security

For information about securing WebLogic Server, see *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server*.

Authorization

Authorization is the process where the interactions between users and WebLogic Server resources are controlled, based on user identity or other information. In WebLogic Server, an Authorization provider is used to limit the interactions between users and WebLogic resources to ensure integrity, confidentiality, and availability.

For more information about changing WebLogic Server passwords, see the WebLogic Server Administration Console Help.

WebLogic Resources

A WebLogic Server resource is a structured object used to represent an underlying WebLogic Server entity, which can be protected from unauthorized access using security roles and security policies.

WebLogic resources are hierarchical. Therefore, the level at which you define these security roles and security policies is up to you. For example, you can define security roles and security policies on entire enterprise applications; an Enterprise Java Bean JAR containing multiple EJBs; a particular Enterprise Java Bean (EJB) within that JAR; or a single method within that EJB.

Security Policies

Security policies replace access control lists and answer the question “Who has access to a WebLogic server resource?" A security policy is created when you define an association between a WebLogic resource and one or more users, groups, or security roles. You can

optionally define date and time constraints for a security policy. A WebLogic resource has no protection until you assign it a security policy.

Security policies are stored in an authorization provider's database. By default, the XACML Authorization provider is configured in a domain, and security policies are stored in the embedded LDAP server.

To use a user or group to create a security policy, the user or group must be defined in the security provider database for the authentication provider that is configured in the default security realm. To use a security role to create a security policy, the security role must be defined in the security provider database for the Role Mapping provider that is configured in the default security realm. By default, the authentication and XACML Role Mapping providers are configured in the database in the embedded LDAP server. Also by default, security policies are defined in WebLogic Server resources. These security policies are based on security roles and default global groups. You also have the option of basing a security policy on a user.

Secure Sockets Layer (SSL)

SSL enables secure communication between applications connected through the Web. WebLogic Server fully supports SSL communication. By default, WebLogic Server is configured for one-way SSL authentication. Using the WebLogic Server Administration Console, you can configure WebLogic Server for two-way SSL authentication.

- To use one-way SSL from a client to a server, enable the SSL port on the server, configure identity for the server and trust for the client.
- To use two-way SSL between a client and a server, enable two-way SSL on the server, configure trust for the server, and identity for the server.

  In either case, the trusted CA certificates must include the trusted CA certificate that issued the peer's identity certificate. This certificate does not necessarily have to be the root CA certificate.

To acquire a digital certificate for your server, generate a public key, private key, and a Certificate Signature Request (CSR), which contains your public key. Send the CSR request to a certificate authority and follow their procedures for obtaining a signed digital certificate.

After you have your private keys, digital certificates, and any additional trusted CA certificates that you may need, you must store them so that WebLogic Server can use them to verify identity. Store private keys and certificates in keystores.

Logging Security

Oracle recommends a Logging level of ERROR for Logging Services. An explicit administrative action is required to change the log level. See "Overview" in *UIM Developer's Guide* for more information. When the log levels are set to DEBUG or lower, the log levels can contain raw exceptions and stack traces that could be exploited to compromise the security of your UIM system.

Oracle Security Documentation

UIM uses other Oracle products, such as Oracle Database and Oracle WebLogic Server. See the following documents, as they apply to UIM:

- *Oracle Database Security Guide*
- *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server*
- *Oracle Application Server Security Guide*

- *Oracle Application Server Administrator's Guide*

File Permissions

Generally, the permissions on a file should be restrictive. However, you should allow users to perform normal application operations on the file.

Table 1-1 File Permissions

Type of File	Linux Permission	Notes
Directories that should be available to all users	755	All Oracle Home Directories unless otherwise indicated.
Directories that should not be available to all users (for example, configuration, domains, servers, and so on)	750, 770	Linux permission values: • 750: ASInstance • 770: Jdk, Inventory and Cfgtoollogs Domain and System instance directories and Restricted directories.
General Binaries should be available to all users, including executable shell scripts (installation)	755	All Oracle Home binaries unless otherwise indicated.
General Binaries (configuration) and Instantiated Binaries (Instantiated with sensitive install specific information such as the OS dba group name)	700	Most of the binaries in Application Server suites are currently in this category. Instantiated binaries may contain sensitive installation-specific information. Therefore, they should be read-protected to the maximum extent.
Binaries and directories imported from a database component	Inherit database install component file permissions	For example, sqlplus (755).
Setuid Binaries	4711, 4710, 4701	This type of binary should include restricted permissions and it should be located in a directory with restrictive permissions.
Setguid Binaries	2711	This type of binary should include restricted permissions and it should be located in a directory with restrictive permissions.
Setguid and setgid Binaries	6711	You should configure this type of binary when it is absolutely necessary to set both user and group identities. Also, this type of binary should be located in a directory with restrictive permissions.
Libraries (for example, *.jar, *.a, *.so, *.war, *.ear)	640 or 750	Linux permission values: • 640: Most libraries belong to this category. • 750: For a few specific platforms, such as HP-UX and *.so must include execute permission as required by the O/. Therefore, set this file permission specific for the platform. **Note**: 640 is the recommended goal. However, until the recommended goal is achieved, 750 is acceptable for all Linux platforms.
Libraries (for example, *.a, *.so) (installation)	644 or 755	NA
Alert, log and trace files (for example, *.out, *.log, *.out.*)	640	Only the install owner can read/write and the group can only read these files due to the nature of sensitive information of these files.

Table 1-1 (Cont.) File Permissions

Type of File	Linux Permission	Notes
Configuration and metadata files (*.xml, *.properties, *.sql, *.pls)	640	Only the install owner should be allowed to read/write the super-sensitive information such as user passwords and the group can only read these files to copy files from ASHome to ASInstance.
Script files (*.pl, *.py, *.sh, e.g. root.sh)	700 or 750	The available scripts are: • 700: Deployment script and security configuration script and files. It includes Root.sh. • 750: Scripts that need to be available to OS group members.
Wallet files (*.sso, for example, cwallet.sso), and Keystore files, or other super sensitive non-executable files	600	Only the install owner can read/write these files due to the nature of super-sensitive information of these files.
All other files (install)	600, 640, 644	644: May allow more restricted files in some cases.
All other files (configuration)	600, 640	640, 600: For sensitive files.

2

Performing a Secure UIM Installation

This chapter presents planning information for your Oracle Communications Unified Inventory Management (UIM) system and describes recommended installation scenarios that enhance security.

For more information about installing UIM, see "Unified Inventory Management Installation Overview " in *UIM Installation Guide*.

Installing UIM Securely

You can perform a custom installation or a typical installation. Oracle recommends that you perform a custom installation to avoid installing options and products you do not need. However, you can perform a typical installation, and remove or disable features you do not need after the installation is complete.

When installing UIM, do the following:

- When creating the WebLogic Server domain for UIM:
  - Make sure that SSL ports are being used on the Administration Server and all Managed servers.
  - If installing UIM on a cluster of servers, configure the cluster addresses to use SSL ports.
  - After you have created the WebLogic Server domain for UIM, start the Administration Server. Then, use t3s to start the Managed servers:

    ```
    startManagedWebLogic.sh ManagedServer_1 t3s://host_name:port
    ```

    where *ManagedServer_1* is the name of the first Managed server, and *port* is the SSL Port of the Administration server.
- Using the WebLogic Server Administration Console, configure Certificate Identity and trust store to use SSL. Do not use the default demonstration certificate that comes with WebLogic Server. See the WebLogic administrator's documentation for more information.
- When you complete the install and patch operations, you must remove the write access to the file system except for data and configuration files. The data and configuration files prevent overwriting of files. Only other operating system users can run the required services. Therefore, ensure to provide a minimum set of file system permissions to other operating system users.
- When you install UIM, avoid generating temporary files. If temporary files are required, ensure to install them with appropriate file permissions in properly protected directories. After the successful or unsuccessful installation or the failure of installation, erase the temporary files securely. If you require any additional privileges for the installation, revoke the temporary files immediately after the successful or unsuccessful installation. For temporary file storage, you can also use volatile memory-based file systems.
- Ensure that any files generated during application processing must have correct file system permissions. See "File Permissions" for more information.

Run the following command to verify files that have execute permissions:

```
find . -type f -perm +111 \! -iname '*.pm' \! -iname '*.so' \! -iname
'*.a' \! -iname '*.pl' \! -iname '*.sh' \! -iname '*.bin' -exec file '{}'
\; |grep --invert-match executable
```

About Password Policies

Oracle recommends having strong password policies for UIM and database schema users. Consider enforcing the following password policies:

- Minimum length of password is 8 characters.
- Password must contain at least one digit, one capital letter, and one special character. For example, `WebLogic@123`.
- The user name must not be part of the password.

Stricter rules can be set for the authentication provider using the WebLogic Server Administration Console. For details on authentication providers and their configuration, refer to WebLogic administrator documentation.

See "Unified Inventory Management System Administration Overview" in *UIM System Administrator's Guide* for information about changing and setting UIM passwords.

Post-Installation Configuration

This section explains security configurations to complete after UIM is installed.

Setting Up User Accounts to Lock and Expire

Create UIM user accounts to lock after a certain number of failed login attempts, and to expire after a certain amount of idle time.

See "Unified Inventory Management System Administration Overview" in *UIM System Administrator's Guide* for information about changing and setting UIM passwords.

3

Implementing UIM Security

This chapter explains the security features of Oracle Communications Unified Inventory Management (UIM).

Configuring and Using Authentication

Authentication is the mechanism by which users provide specific information as a proof of having access to a system. Authentication answers the question “Who are you?" using credentials such as user name and password.

In Oracle WebLogic Server, authentication providers are used to prove the identity of users or system processes. Authentication providers also remember, transport, and make identity information available to various components of a system when needed. During the authentication process, a principal validation provider provides additional security protection for the principals (users and groups) contained within the subject by signing and verifying the authenticity of those principals.

Upon installation, UIM uses the WebLogic-embedded Lightweight Directory Access Protocol (LDAP) as the authentication provider. However, you can use any WebLogic-supported authentication provider, such as Oracle Internet Directory (OID), Relational Database Management System (RDBMS), or Security Assertion Markup Language (SAML).

> **Note**
>
> If your UIM environment requires high-grade security and your user base is high (over 10,000 users), Oracle recommends you use an external LDAP such as OID.

UIM uses user name and password authentication. See "Unified Inventory Management System Administration Overview" in *UIM System Administrator's Guide* for more information.

Whether UIM is configured to communicate with WebLogic Server over HTTP or HTTPS, login authentication is always sent over a secured HTTPS channel.

If you are using a Web services interface, authentication details are supplied with each request using the Username token header. See "Web Services Overview" in *UIM Web Services Developer's Guide* for more information.

Java Authentication and Authorization Service

WebLogic Server uses the Java Authentication and Authorization Service (JAAS) classes to authenticate to the client, whether the client is an application, applet, Enterprise JavaBean, or servlet that requires authentication.

JAAS implements a Java version of the Pluggable Authentication Module (PAM) framework, which permits applications to remain independent from underlying authentication technologies. Therefore, the PAM framework allows the use of new or updated authentication technologies without requiring modifications to the application.

About Callback Handlers

A callback handler is a flexible JAAS standard that allows a variable number of arguments to be passed as complex objects to a method.

There are three types of callback handlers: **NameCallback**, **PasswordCallback**, and **TextInputCallback**, all of which are part of the **javax.security.auth.callback** package. **NameCallback** and **PasswordCallback** return the user name and password, respectively. You can use **TextInputCallback** to access the data users enter into any additional fields on a login form (that is, fields other than those for obtaining the user name and password). When used, there should be one **TextInputCallback** per additional form field, and the prompt string of each **TextInputCallback** must match the field name in the form. WebLogic Server uses only the **TextInputCallback** for form-based Web application login.

An application implements a callback handler and passes it to underlying security services so that they may interact with the application to retrieve specific authentication data, such as user names and passwords, or to display certain information, such as error and warning messages.

Callback handlers are implemented in an application-dependent fashion. For example, implementations for an application with a UI may prompt users for requested information, or display error messages. An implementation may also choose to obtain requested information from an alternative source without asking the user.

Underlying security services make requests for different types of information by passing individual callbacks to the callback handler. The callback handler implementation decides how to retrieve and display information depending on the callbacks passed to it.

Configuring and Using Access Control

Authorization is used to control access by:

- Permitting only certain users to access a resource or action
- Applying varying limitations on user access or actions

Upon installation, UIM defines the **uimuser** role. This is a super role that grants access to all UIM resources, so the role should not be granted to everyone. Rather, Oracle recommends that you define your own application roles to restrict access to UIM resources.

The **uimuser** role is part of the **uim-users** WebLogic Server group. To access UIM, a user must be assigned a role that is part of the **uim-users** group. If a user is not assigned a role that is part of the **uim-users** group, after successful authentication, the user encounters the following error:

```
You do not have permission to access this page. Contact the Administrator.
```

Enabling Access for Specifications

In WebLogic Server Administration console, to enable access for a specification, you should add the corresponding user group to the specification.

To enable access for a specification:

1. Open WebLogic Server Administration console.
2. Create the user groups and assign users to the groups.

> **Note**
>
> UIM recognizes the user group only if the group name begins with **ora_uim_partition#**.

3. In **SPECIFICATION** table, update **PARTITION** column of the corresponding specification record with the user group name.

   For example, if the user group name is *mypartition*, update the **PARTITION** column with **/ mypartition**. After updating the table, only the users belonging to *mypartition* group can see this specification.

4. Update the **PARTITION** column for all specifications, for which the users need access.
5. Restart the application servers after clearing *tmp* and *cache* as the above steps modify the data in database.
6. Set the following property in system-config.properties file:

   ```
   uim.security.filter.enabled=true
   ```

Configuring and Using Security Audit Logs

For information about configuring and using security audit logs in Oracle WebLogic Server, see:

http://docs.oracle.com/middleware/1221/wls/SECMG/toc.htm

Secure Access to UIM Web Services

The Web services API is standards based using JAX-RPC over HTTPS. The UIM Web services API uses the same security access level as the UIM UI. So any user able to login to UIM can also use the Web Service API.

Managing UIM Security

UIM System Administrator's Guide contains information on the following security management topics:

- Authentication
  - Password management
  - Authenticating Web services
- Authorization
  - Security roles and policies
  - Taskflow and resource permissions
  - Application role management
  - Application policy management
  - Enabling Web service authentication at runtime
  - Partitioning the database

4

Security Considerations for Developers

This chapter provides information for developers about how to create secure applications for Oracle Communications Unified Inventory Management (UIM) and how to extend UIM without compromising its security.

About UIM Security Policies

UIM uses ADF security for its UI resources (JSDD or JSPX), and protects them with the **uimuser** role. Users having this role can run create, read, update, and delete operations on these pages. These policies can be customized in Oracle Fusion Middleware Enterprise Manager.

About Securing UIM APIs

By default, UIM APIs are not secured. To secure an API, you must extend UIM security to include the APIs. This can be done by:

- Securing APIs through the SecurityValidation Aspect
- Securing APIs through rulesets and extension points

See "Overview" in *UIM Developer's Guide* for more information.

About Securing Entity Data

By default, UIM entity data is not secured. To secure entity data, you must extend UIM security to control data access to individual entities. This is done by creating custom rulesets that run at specified extension points. The custom rulesets set permissions or partitions for an entity, enforces any permissions or partitions that are set for an entity, and logs error messages whenever a security violation is detected.

See "Overview" in *UIM Developer's Guide* for more information.

About Securing Web Services

By default, the Service Fulfillment Web service has security enabled upon installation. Specifically, the HTTP and JMS Web service ports are associated to the default WebLogic security policy file, **Auth.xml**. As a result, a user name and password must be sent in clear text over a secure tunnel (HTTPS/t3s). You can modify the default security settings through the WebLogic Server Administration Console. See *UIM Web Services Developer's Guide* for more information.

When you create a custom Web service, it is up to you secure the Web service. How you secure the Web service depends upon how you created the Web service. For example, if your custom Web service deploys with the **custom.ear** file, you need to create your own deployment plan; if your custom Web service deploys with the **inventory.ear** file, you need to modify the **inventory.ear** deployment plan that is part of the UIM installation. See "Web Services Overview" in *UIM Web Services Developer's Guide* for more information.

A

UIM Secure Deployment Checklist

The following security checklist lists guidelines to help you secure Oracle Communications Unified Inventory Management (UIM) and its components.

Secure Deployment Checklist

- Install only the components you require.
- Lock and expire default user accounts.
- Enforce strong password management.
- Enable data dictionary protection on the Oracle Database for UIM.
- Restrict, control, and revisit user privileges:
  - Grant only the necessary privileges to each user.
  - Revoke unnecessary privileges from the PUBLIC user group.
  - Restrict permissions on run-time facilities.
- Enforce the use of access controls by using the Authorization Policies.
- Require clients to authenticate.
- Restrict network access by doing the following:
  - Use firewalls.
  - Never leave an unnecessary hole in a firewall.
  - Password-protect the Oracle listener against remote access.
  - Monitor listener activity.
  - Monitor who accesses your systems.
  - Restrict system access by IP addresses.
  - Encrypt network traffic.
- Apply all security patches and workarounds.
- Encrypt sensitive information.
- Contact Oracle Security Products if you discover a vulnerability in any Oracle product.