



BEA WebLogic RFID Enterprise Server™

Multi-Tenancy

Version 2.0™
Revised: January, 2007

Copyright

Copyright © 1995-2007 BEA Systems, Inc. All Rights Reserved.

Restricted Rights Legend

This software is protected by copyright, and may be protected by patent laws. No copying or other use of this software is permitted unless you have entered into a license agreement with BEA authorizing such use. This document is protected by copyright and may not be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form, in whole or in part, without prior consent, in writing, from BEA Systems, Inc.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE DOCUMENTATION IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA SYSTEMS DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE DOCUMENT IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks and Service Marks

Copyright © 1995-2007 BEA Systems, Inc. All Rights Reserved. BEA, BEA JRockit, BEA WebLogic Portal, BEA WebLogic Server, BEA WebLogic Workshop, BEA WebLogic RFID Mobile SDK, Built on BEA, Jolt, JoltBeans, SteelThread, Top End, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA AquaLogic, BEA AquaLogic Data Services Platform, BEA AquaLogic Enterprise Security, BEA AquaLogic Interaction, BEA AquaLogic Interaction Analytics, BEA AquaLogic Interaction Collaboration, BEA AquaLogic Interaction Integration Services, BEA AquaLogic Interaction Process, BEA AquaLogic Interaction Publisher, BEA AquaLogic Interaction Studio, BEA AquaLogic Service Bus, BEA AquaLogic Service Registry, BEA AquaLogic BPM Designer, BEA AquaLogic BPM Studio, BEA AquaLogic BPM Enterprise Server – Standalone, BEA AquaLogic BPM Enterprise Server – BEA WebLogic, BEA AquaLogic BPM Enterprise Server – IBM WebSphere, BEA AquaLogic BPM Enterprise Server – JBoss, BEA AquaLogic BPM Process Analyzer, BEA AquaLogic Interaction Development Kit, BEA AquaLogic Interaction JSR-168 Consumer, BEA AquaLogic Interaction Identity Service – Active Directory, BEA AquaLogic Interaction Identity Service – LDAP, BEA AquaLogic Interaction Content Service – Microsoft Exchange, BEA AquaLogic Interaction Content Service – Lotus Notes, BEA AquaLogic Interaction Logging Utilities, BEA AquaLogic Interaction WSRP Consumer, BEA AquaLogic Interaction Portlet Framework – Microsoft Excel, BEA AquaLogic Interaction .NET Application Accelerator, AquaLogic Interaction Content Service – Documentum, BEA AquaLogic Interaction Content Service – Windows Files, BEA AquaLogic Interaction Portlet Suite – IMAP, BEA AquaLogic Interaction Portlet Suite – Lotus Notes, BEA AquaLogic Interaction Portlet Suite – Exchange, BEA AquaLogic Interaction Portlet Suite – Documentum, BEA AquaLogic Interaction IDK Extension, BEA AquaLogic HiPer Workspace for BPM, BEA AquaLogic HiPer Workspace for Retail, BEA AquaLogic Sharepoint Console, BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Kodo, BEA Liquid Data for WebLogic, BEA Manager, BEA MessageQ, BEA SALT, BEA Service Architecture Leveraging Tuxedo, BEA WebLogic Commerce Server, BEA WebLogic Communications Platform, BEA WebLogic Enterprise, BEA WebLogic Enterprise Platform, BEA WebLogic Enterprise Security, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Java Adapter for Mainframe, BEA WebLogic JDriver, BEA WebLogic Log Central, BEA WebLogic Mobility Server, BEA WebLogic Network Gatekeeper, BEA WebLogic Personalization Server, BEA WebLogic Personal Messaging API, BEA WebLogic Platform, BEA WebLogic Portlets for Groupware Integration, BEA WebLogic Real Time, BEA WebLogic RFID Compliance Express, BEA WebLogic RFID Edge Server, BEA WebLogic RFID Enterprise Server, BEA WebLogic Server Process Edition, BEA WebLogic SIP Server, BEA WebLogic WorkGroup Edition, BEA Workshop for WebLogic Platform, BEA Workshop for JSF, BEA Workshop for JSP, BEA Workshop for Struts, BEA Workshop Studio, Dev2Dev, Liquid Computing, and Think Liquid are trademarks of BEA Systems, Inc. Accelerated Knowledge Transfer, AKT, BEA Mission Critical Support, BEA Mission Critical Support Continuum, and BEA SOA Self Assessment are service marks of BEA Systems, Inc.

All other names and marks are property of their respective owners.

Introduction and Roadmap

The following sections describe the audience for and organization of this document:

- [“Document Scope and Audience”](#) on page 1-1
- [“Related Documentation”](#) on page 1-1

Document Scope and Audience

This document describes [Guide to This Document](#)

This document is organized as follows:

- This chapter, [Introduction and Roadmap](#), describes the scope of this document and related information.

Related Documentation

This manual is part of the WebLogic RFID Enterprise Server documentation set, which includes the following documents and online help:

- [WebLogic RFID Enterprise Server Product Overview](#) provides an overview of the WebLogic RFID Enterprise Server components and architecture.
- [Installing WebLogic RFID Enterprise Server](#) describes how to install and configure WebLogic RFID Enterprise Server.
- [Understanding the Event, Master Data, and Data Exchange Services](#) describes the services that implement the EPC Information Services (EPCIS).

- [*Query Subscription Administration Console Online Help*](#) describes how to create and manage subscriptions: queries that run at specified times and send results to specific destinations.
- [*Master Data Administration Console Online Help*](#) describes how to create and work with master data types and master data entries.
- [*Generating WebLogic RFID Enterprise Server Reports*](#) describes how to use the RFID Enterprise Server Reporting Service to display predefined RFID reports in a Web browser.
- [*Edge Server Administration Console Online Help*](#) describes how to use the RFID Edge Server Administration Console to view and manage RFID Edge Servers in the enterprise.
- [*Using the Serial Number Assignment Service*](#) describes how to use the RFID Enterprise Server Serial Number Assignment service to provide pools of RFID serial numbers for assigning to EPC tags.
- [*Using the Telemetry Console Extension*](#) describes how to use the Telemetry Console Extension for graphically presenting real-time Edge Server and RFID device telemetry data. The Telemetry Console Extension is part of the RFID Edge Server Administration Console.
- [*Release Notes*](#) lists known problems and workarounds in this release of the RFID Enterprise Server.

Enterprise Server and Multi-Tenancy

The following sections explain how to install, and configure a multi-tenant database in your RFID Enterprise Server environment:

- [“Install and Configure WebLogic Server and Enterprise Server” on page 2](#)
- [“Create and Populate a Directory for Each Customer” on page 2](#)
- [“Create Web Service URLs” on page 4](#)
- [“Create JMS Queue Resources” on page 7](#)
- [“Configure Database Connections and Credential Maps” on page 8](#)
- [“Determine the Names Used to Deploy Applications” on page 11](#)
- [“Configure Security and Deploy Applications” on page 14](#)

Overview

You can configure Enterprise Server 1.1 to provide multi-tenancy; however, support and documentation for this feature is available only through BEA support services, and only to specific customers.

Multi-tenancy means hosting data for multiple customers on a single server, where customers can only access their data; no customer can view or modify another customer's data. Multi-tenancy is implemented using either a single database with a separate schema for each customer, or with

multiple discrete databases. The procedures in this document use a single database with separate schemas.

To implement multi-tenancy, perform the following tasks:

1. Install WebLogic Server 9.1 and Enterprise Server 1.1. See [“Install and Configure WebLogic Server and Enterprise Server” on page 2](#).
2. In the Enterprise Server directory, create a separate directory for each customer. Populate each directory with `epcis.ear` and `epc-provisioning.ear`. See [“Create and Populate a Directory for Each Customer” on page 2](#).
3. For each customer, create a unique Web service URL. See [“Create Web Service URLs” on page 4](#).
4. For each customer, create unique JMS queue names. See [“Create JMS Queue Resources” on page 7](#).
5. Set up the database connection details. See [“Configure Database Connections and Credential Maps” on page 8](#).
6. Configure security for the applications and the JMS resources, then deploy the applications. See [“Configure Security and Deploy Applications” on page 14](#).

Install and Configure WebLogic Server and Enterprise Server

Follow the instructions provided in the WebLogic RFID Enterprise Server Installation Guide. Note the remainder of the multi-tenancy instructions use the directory shortcuts listed in that manual.

Create and Populate a Directory for Each Customer

Each customer must have a directory which contains that customer’s copy of `epcis.ear` and `epc-provisioning.ear`.

1. Create a directory for each customer; the location of the directories is arbitrary. For example, if you have three customers represented by the `es_cust1`, `es_cust2`, and `es_cust3`, create the following directories:

```
BEA_HOME\customers\es_cust1
BEA_HOME\customers\es_cust2
BEA_HOME\customers\es_cust3
```

2. Copy the following files from the *BEA_HOME/user_projects/applications/your-domain/* directory into each of the customer directories:
 - companyprefix-lib.ear
 - enterprise-login.ear
 - enterprise-reports.ear
 - epcis.ear, epcis-lib.ear
 - epc-provisioning.ear
 - kodo.ear
 - loginconsole-lib.ear
 - masterdata.ear
 - masterdata-lib.ear directories from

For example, after copying the files, the *es_cust1* directory contains the following files:

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/companyprefix-lib.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/enterprise-login.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/enterprise-reports.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/epcis.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/epcis-lib.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/epc-provisioning.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/kodo.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/loginconsole-lib.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/masterdata.ear/...

BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/masterdata-lib.ear/...

Create Web Service URLs

There are four web services in the Enterprise Server: EPCIS, legacy EPCIS (1.1 version), Provisioning Service, and Master Data. Additionally, there are multiple Web consoles and servlets. Each company must have unique URLs for accessing copies of the applications. To create those URLs, edit the following `application.xml` and `web.xml` files, for each customer:

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/epcis
.ear/META-INF/application.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/epc-p
rovisioning.ear/META-INF/application.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/maste
rdata.ear/META-INF/application.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/enter
prise-reports.ear/META-INF/application.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/enter
prise-login.ear/META-INF/application.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/epcis
.ear/epcis-scheduler-servlet.war/WEB-INF/web.xml
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>/epcis
.ear/epcis-servlet.war/WEB-INF/web.xml
```

In each file, change the values of `<context-root>` elements to unique company-specific strings. The example below also includes updates to the `<context-root>` elements for web-based applications and servlets. Comment out the modules relating to the help consoles as these will be shared across customers

1. Edit the `customers/es_cust1/epcis.ear/META-INF/application.xml` file:

```
<module>
    <web>
        <web-uri>legacy-ws.war</web-uri>
        <context-root>legacyepcis-es_cust1</context-root>
    </web>
</module>
```



```

<module>
    <web>
        <web-uri>epcis-ws.war</web-uri>
        <context-root>epcis-es_cust1</context-root>
    </web>
</module>

<module>
    <web>
        <web-uri>epcis-servlet.war</web-uri>
        <context-root>EPCISServlet-es_cust1</context-root>
    </web>
</module>

<module>
    <web>
        <web-uri>epcis-console.war</web-uri>
        <context-root>epcis-console-es_cust1</context-root>
    </web>
</module>

<!--module>
    <web>
        <web-uri>epcis-console-help.war</web-uri>
        <context-root>querysubscriptionconsole-help</context-root>
    </web>
</module-->

<module>
    <web>
        <web-uri>epcis-scheduler-servlet.war</web-uri>

```

```
        <context-root>schedulerservlet-es_cust1</context-root>
    </web>
</module>
```

2. Edit the /es_cust1/epc-provisioning.ear/META-INF/application.xml file:

```
<module>
    <web>
        <web-uri>provisioning-service.war</web-uri>
        <context-root>epcprovisioning-es_cust1</context-root>
    </web>
</module>
<module>
    <web>
        <web-uri>epcps-webclient.war</web-uri>
        <context-root>epcps_webclient-es_cust1</context-root>
    </web>
</module>
```

3. Edit the customers/es_cust1/masterdata.ear/META-INF/application.xml file:

```
<module>
    <web>
        <web-uri>masterdata-ws.war</web-uri>
        <context-root>masterdata-es_cust1</context-root>
    </web>
</module>
<module>
    <web>
        <web-uri>masterdata-console.war</web-uri>
        <context-root>masterdataadminconsole-es_cust1</context-root>
```

```

        </web>
    </module>

    <!--module>
        <web>
            <web-uri>masterDataConsole-help.war</web-uri>
            <context-root>masterdataconsole-help</context-root>
        </web>
    </module-->

```

4. Edit the

customers/es_cust1/enterprise-reports.ear/META-INF/application.xml file:

```

<module>
    <web>
        <web-uri>reports.war</web-uri>
        <context-root>enterprise-reports-es_cust1</context-root>
    </web>
</module>

```

Create JMS Queue Resources

Each company must have its own JMS queue resources for the EPCIS and Provisioning Services.

1. In the WebLogic Server Administration console, create the JMS queue resources using customer-specific JNDI names for the resources. For example:

```

EPCISMessages-es_cust1
EPCISMessages-es_cust2
EPCISMessages-es_cust3
EPCISFailedMessages-es_cust1
EPCISFailedMessages-es_cust2
EPCISFailedMessages-es_cust3
epcpsBrokerMessages-es_cust1
epcpsBrokerMessages-es_cust2
epcpsBrokerMessages-es_cust3

```

2. Modify the customer's configuration to use its JMS queue name.

For each customer, edit the

`epcis.ear/epcis-mdb.jar/META-INF/weblogic-ejb-jar.xml` file.

Change the value of the `<destination-jndi-name>` element to match the customer's JMS queue JNDI name. Using `es_cust1` as an example:

Edit the `es_cust1/epcis.ear/epcis-mdb.jar/META-INF/weblogic-ejb-jar.xml` file:

```
<destination-jndi-name>EPCISMessages-es_cust1</destination-jndi-name>
```

Modify each customer's configuration to use its JMS queue name. Edit the

`epc-provisioning.ear/epcps-broker-embedded-mdb.jar/META-INF/weblogic-ejb-jar.xml` file:

```
<destination-jndi-name>epcpsBrokerMessages-es_cust1</destination-jndi-name>
```

3. Modify each customer's `weblogic-ejb-jar` files:

- `epcis.ear/epcis-mdb.jar/META-INF/weblogic-ejb-jar.xml`
- `epc-provisioning.ear/epcps-broker-embedded-mdb.jar/META-INF/weblogic-ejb-jar.xml`

Configure Database Connections and Credential Maps

To configure the database connection information for each customer, you need to edit the following files:

- `kodo.ear/kodo.rar/META-INF/ra.xml`
- `kodo.ear/kodo.rar/META-INF/weblogic-ra.xml`
- `companyprefix-lib.ear/companyprefix-ejb.jar/META-INF/ejb-jar.xml`
- `epc-provisioning.ear/epcps-broker-embedded-mdb.jar/META-INF/ejb-jar.xml`
- `epc-provisioning.ear/epcps-ejb.jar/META-INF/ejb-jar.xml`
- `epc-lib.ear/epcis-capture-ejb.jar/META-INF/ejb-jar.xml`
- `epc-lib.ear/epcis-ejb.jar/META-INF/ejb-jar.xml`
- `epc-lib.ear/epcis-reports-ejb.jar/META-INF/ejb-jar.xml`
- `loginconsole-lib.ear/loginconsole-ejb.jar/META-INF/ejb-jar.xml`
- `masterdata-lib.ear/masterdata-ejb.jar/META-INF/ejb-jar.xml`

1. In each customer's `kodo.ear/kodo.rar/META-INF/ra.xml` file, find the following `<config-property-name>` elements. Modify the associated `<config-property-value>` as required for the customer's database configuration:

```
<config-property-name>ConnectionURL</config-property-name><config-property-value>jdbc:pointbase:server://localhost:9092/ent_server</config-property-value>
```

```
<config-property-name>ConnectionDriverName</config-property-name>
```

```
<config-property-value>com.pointbase.jdbc.jdbcUniversalDriver
```

2. In each customer's kodo.ear/kodo.rar/META-INF/weblogic-ra.xml files, modify the <jndi-name> property so that it contains a unique customer-specific value. Using customer es_cust1 as an example:

```
<jndi-name>pmf-enterprise-es_cust1</jndi-name>
```

3. Edit each customer's ejb-jar.xml files to replace the associated <env-entry-value> with the string you specified for the <jndi-name> property in that customer's kodo.ear/kodo.rar/META-INF/weblogic-ra.xml file. The following files must be edited:

- companyprefix-lib.ear\companyprefix-ejb.jar\META-INF\ejb-jar.xml
- epc-provisioning.ear\epcps-broker-embedded-mdb.jar\META-INF\ejb-jar.xml
- epc-provisioning.ear\epcps-ejb.jar\META-INF\ejb-jar.xml
- epcis-lib.ear\epcis-capture-ejb.jar\META-INF\ejb-jar.xml
- epcis-lib.ear\epcis-ejb.jar\META-INF\ejb-jar.xml
- epcis-lib.ear\epcis-reports-ejb.jar\META-INF\ejb-jar.xml
- loginconsole-lib.ear\loginconsole-ejb.jar\META-INF\ejb-jar.xml
- masterdata-lib.ear\masterdata-ejb.jar\META-INF\ejb-jar.xml<env-entry-name>PMFJNDIName</env-entry-name>

Replace the associated <env-entry-value> with the string you specified for the <jndi-name> property in that customer's epcis.ear/kodo.rar/META-INF/weblogic-ra.xml file. Using customer es_cust1 as an example:

```
<env-entry>
```

```
    <env-entry-name>pmfJNDIName</env-entry-name>
```

```
    <env-entry-type>java.lang.String</env-entry-type>
```

```
    <env-entry-value>pmf-enterprise-es_cust1</env-entry-value>
```

```
</env-entry>
```

Configure the Enterprise Server Database

In the [Installing WebLogic RFID Enterprise Server](#) document, the section on configuring the Enterprise Server database configures and initializes the database for one user. This section provides the additional information required to configure multiple databases for access by several users, where each user has access to only one database. Start the PointBase server and console as described in the Administrator's Guide.

1. For each customer database, create a user for each customer with the username and password you assigned to that customer in [“Configure Database Connections and Credential Maps” on page 8](#). Do not assign any database roles or privileges that will allow one user to see another user's schema or data.
2. For each customer, edit `kodo.properties` and run the `initEventServiceDatabase` and `initSerialNumberDatabase` scripts. Because `kodo.properties` is in the `domain\config` directory, you will have to perform iterative customer-specific edits on the `\kodo.properties` file prior to running the scripts. For each customer:
 - a. Edit `kodo.properties` to modify the `ConnectionURL`, `ConnectionUserName`, `ConnectionPassword`, and `ConnectionDriverName` for that customer.
 - b. Copy the edited `kodo.properties` file to a customer-specific name for safekeeping. For example, after configuring the file for `es_cust1`, copy `kodo.properties` to `kodo.properties.es_cust1`.
 - c. Run the `initEventServiceDatabase` and `initSerialNumberDatabase` scripts.

Create Credential Map

For each customer create a Credential Map that stores the user name and password for that customer's database connection:

1. Open the WLS console.
2. Under Domain Structure, click on Security Realms.
3. In the right hand pane, select myrealm
4. Click on the Credential Mappings tab.
5. Press the New button.

6. In the Create New Credential Mapping screen, enter the customer's full Database Connection URL from the ra.xml file into the Remote Host field. Leave all of the other fields empty and click Next.
7. On the next screen, enter *rfiduser* into the following fields:
 - a. local user
 - b. Customer's database user name
 - c. Customer's database password
 - d. Remote user
 - e. Remote password
8. Click Finish. Note that you must restart WLS for the new Credential Mapping to take effect.

Determine the Names Used to Deploy Applications

When deploying the applications, give each a name that will make it easy to recognize which customer it is associated with; starting them all with the same prefix will result in their being grouped by customer in the Deployment tab of the WLS console. For example, cust1's applications might be prefixed with cust1- and so kodo.ear would be deployed as cust1-kodo, companyprefix-lib.ear would be deployed as cust1-companyprefix-lib, etc.

In the next two sections you will update references to application names, so be sure you are consistent in spelling and capitalization. You can use xref to record the names you intend to use for each application.

Table 1

Application (Deployed Name)	Sample (cust1)	Customer 1 Name	Customer 2 Name
kodo.ear (WLFRFID-KodoEnterpriseServer)			
companyprefix-lib.ear (WLFRFID-CompanyPrefixLib)			

Table 1

Application (Deployed Name)	Sample (cust1)	Customer 1 Name	Customer 2 Name
loginconsole-lib.ear (WLRFID-LoginConsoleLib)			
epcis-lib.ear (WLRFID-EPCISLib)			
masterdata-lib.ear (WLRFID-MasterData)			
Enterprise-login.ear (WLRFID-EnterpriseLogin)			
epcis.ear (WLRFID-EPCIS)			
masterdata.ear (WLRFID-MasterData)			
enterprise-reports.ear (WLRFID-EnterpriseReports)			
epc-provisioning.ear (WLRFID-EPCProvisioning)			

Update References to Named Applications

Two of the applications that will be deployed as libraries in WLS have manifests in their META-INF directory that specify the name with which the library will be deployed. Before deploying these libraries, you must update that file to contain the name you plan to use. Here are the files that must be updated. Change them to use the name specified in the table above:

```
BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/comp
anyprefix-lib.ear/META-INF/MANIFEST.MF
```

```
BEA_HOME/user_projects/applications/your-domain/customers/es_cust1/logi
nconsole-lib.ear/META-INF/MANIFEST.MF
```


You will need also to update references to the libraries in `weblogic-application.xml`. So, using the sample above, you would change references to the library `WLRfid-LoginConsoleLib` to `cust1-LoginConsoleLib`. Below is a list of the files and references that must be updated (examples are for `cust1`):

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>
/enterprise-login.ear/META-INF/weblogic-application.xml
```

```
<library-ref>
```

```
    <library-name>cust1-LoginConsoleLib</library-name>
```

```
</library-ref>
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>
/enterprise-reports.ear/META-INF/weblogic-application.xml
```

```
<library-ref>
```

```
    <library-name>cust1-EPCISLib</library-name>
```

```
</library-ref>
```

```
<library-ref>
```

```
    <library-name>cust1-MasterDataLib</library-name>
```

```
</library-ref>
```

```
<library-ref>
```

```
    <library-name>cust1-LoginConsoleLib</library-name>
```

```
</library-ref>
```

```
<library-ref>
```

```
    <library-name>cust1-CompanyPrefixLib</library-name>
```

```
</library-ref>
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>
/epc-provisioning.ear/META-INF/weblogic-application.xml
```

```
<library-ref>
```

```
    <library-name>cust1-LoginConsoleLib</library-name>
```

```
</library-ref>
```

```
<library-ref>
```

```
    <library-name>cust1-CompanyPrefixLib</library-name>
```

```
</library-ref>
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>  
/epcis.ear/META-INF/weblogic-application.xml
```

```
<library-ref>  
  <library-name>cust1-EPCISLib</library-name>  
</library-ref>  
<library-ref>  
  <library-name>cust1-MasterDataLib</library-name>  
</library-ref>  
<library-ref>  
  <library-name>cust1-LoginConsoleLib</library-name>  
</library-ref>
```

```
BEA_HOME/user_projects/applications/your-domain/customers/<customer>  
/masterdata.ear/META-INF/weblogic-application.xml
```

```
<library-ref>  
  <library-name>cust1-MasterDataLib</library-name>  
</library-ref>  
<library-ref>  
  <library-name>cust1-LoginConsoleLib</library-name>  
</library-ref>
```

Configure Security and Deploy Applications

1. Undeploy all of the default applications using the WLS console.

Modify the deployment descriptors for the Subscription and Master Data applications only as these will be shared across tenants.

In `BEA_HOME/user_projects/applications/your-domain/epcis.ear/epcis-console-help.war/WEB-INF/weblogic.xml`, add the following just above the closing `</weblogic-web-app>` tag:

```
<context-root>querysubscriptionconsole-help</context-root>
```

In *BEA_HOME/user_projects/applications/your-domain/master-data.ear/masterDataConsole-help.war/WEB-INF/weblogic.xml*, add the following just above the closing `</weblogic-web-app>` tag:

```
<context-root>masterdataconsole-help</context-root>
```

Re-deploy and start the two help console applications:

```
BEA_HOME/user_projects/applications/your-domain/epcis.ear/  
epcis-console-help.war
```

```
BEA_HOME/user_projects/applications/your-domain/master-  
data.ear/  
masterDataConsole-help.war
```

2. Create Security Roles

For each customer, you will need to create security role(s) for use with the customer's JMS resources and Web applications. Please review the discussion of Security in the [Installing WebLogic RFID Enterprise Server](#) document. By default there are seven roles that are used to access various part of the RFID Enterprise Server: `rfid_admin`, `epcis_admin`, `epcis_mgr`, `epcis_user`, `provisioning_bea`, `provisioning_mgr`, and `provisioning_user`. Depending on the requirements of each customer, they may desire a version of all these roles or just one or two. If, for example, `cust1` says that any user who has access to the system should have access to all parts of it, then you might just create one role: `rfid_admin-cust1_es`. If `cust2` wanted more fine-grained security, you might create customer-specific versions of all roles: `rfid_admin-cust2_es`, `epcis_admin-cust2_es`, `epcis_mgr-cust2_es`, `epcis_user-cust2_es`, `provisioning_bea-cust2_es`, `provisioning_mgr-cust2_es`, and `provisioning_user-cust2_es`.

3. Configure security for each of the JMS Resources you created in "Create JMS Queue Resources" on page []. The suggested roles for these queues are:

EPCISCapturerfid_admin, epcis_admin, epcis_mgr

EPCISMessagesrfid_admin, epcis_admin, epcis_mgr

epcpsBrokerMessagesrfid_admin, provisioning_bea

Perform this procedure for each customer's JMS queue resources.

4. Deploy the Enterprise Server Application

Deploy the applications by modifying the config.xml file in your domain's config directory. For each client, add the following directly under the <configuration-version> tag, updating the values for <name> and <source-path> as appropriate (examples are for cust1). All the <app-deployment> tags for all customers must be together, followed by all the <library> tags for all customers:

```
<app-deployment>
  <name>cust1-EnterpriseLogin</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/enterprise-login.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</app-deployment>
<app-deployment>
  <name>cust1-KodoEnterpriseServer</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/kodo.ear</source-path>
  <deployment-order>90</deployment-order>
  <security-dd-model>DDOnly</security-dd-model>
</app-deployment>
```

```

<app-deployment>
  <name>cust1-EPCProvisioning</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/epc-provisioning.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</app-deployment>
<app-deployment>
  <name>cust1-EPCIS</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/epcis.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</app-deployment>
<app-deployment>
  <name>cust1-MasterData</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/masterdata.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</app-deployment>
<app-deployment>
  <name>cust1-EnterpriseReports</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

  <source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/enterprise-reports.ear</source-path>

```

```
<security-dd-model>CustomRolesAndPolicies</security-dd-model>
</app-deployment>
<library>
  <name>cust1-LoginConsoleLib</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/loginconsole-lib.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</library>
<library>
  <name>cust1-CompanyPrefixLib</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/companyprefix-lib.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</library>
<library>
  <name>cust1-EPCISLib</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/epcis-lib.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
</library>
<library>
  <name>cust1-MasterDataLib</name>
  <target>AdminServer</target>
  <module-type>ear</module-type>
```

```

<source-path>BEA_HOME/user_projects/applications/your-domain/customers/
es_cust1/masterdata-lib.ear</source-path>
  <security-dd-model>CustomRolesAndPolicies</secu-
rity-dd-model>

</library>s

```

5. Configure security for the web services and web applications using the custom security role(s) you created for each customer. The suggested roles for each are:

Table 1:

Item	Component	Role(s)
Login web GUI, enterprise-cust1_es, URL pattern: *.jsp	enterprise-login	rfid_admin, provisioning_mgr, provisioning_user, provisioning_bea
EPCIS web service	epcis	rfid_admin, epcis_admin, epcis_mgr
Legacy EPCIS web service	epcis	rfid_admin, epcis_admin, epcis_mgr
Subscription administration console epcis-console-cust1_es, URL pattern: *.portal	epcis	rfid_admin, epcis_admin
Master Data web service	masterdata	rfid_admin, epcis_admin, epcis_mgr
Master Data administration console masterdataadminconsole-cust1_es URL pattern: *.portal	masterdata	rfid_admin, epcis_admin, epcis_mgr
Reporting web GUI enterprise-reports-cust1_es URL pattern: *.faces	enterprise-reports	rfid_admin, epcis_mgr

Table 1:

Serial Number web service	epc-provisioning	rfid_admin, provisioning_user, provisioning_b provisioning_mgr
Serial number web GUI Epcps_webclient-cust1_es URL pattern: *.faces	epc-provisioning	rfid_admin, provisioning_mgr

The Enterprise Server Administrator's Guide describes how to create security roles, configure security for JMS resources, deploy applications, and configure security for Web applications. Perform the following procedures in the Administrator's Guide for each customer:

- **Create a Security Role:** When creating roles, create a unique role (or roles if you want to separate administration from access) for use with each customer's JMS resources and Web applications.
- **Configure Security for JMS Resources:** Perform this procedure for each customer's JMS queue resources. You created the resources in [“Create JMS Queue Resources” on page 7](#).
- **Deploy the Enterprise Server Applications:** Perform this procedure for each customer's `epcis.ear` and `epc-provisioning.ear` applications.
- **Configure Security and Start the Enterprise Server Web Applications:** Perform this procedure for each customer's `epcis`, `epc_client`, and `epcprovisioning` Web applications.