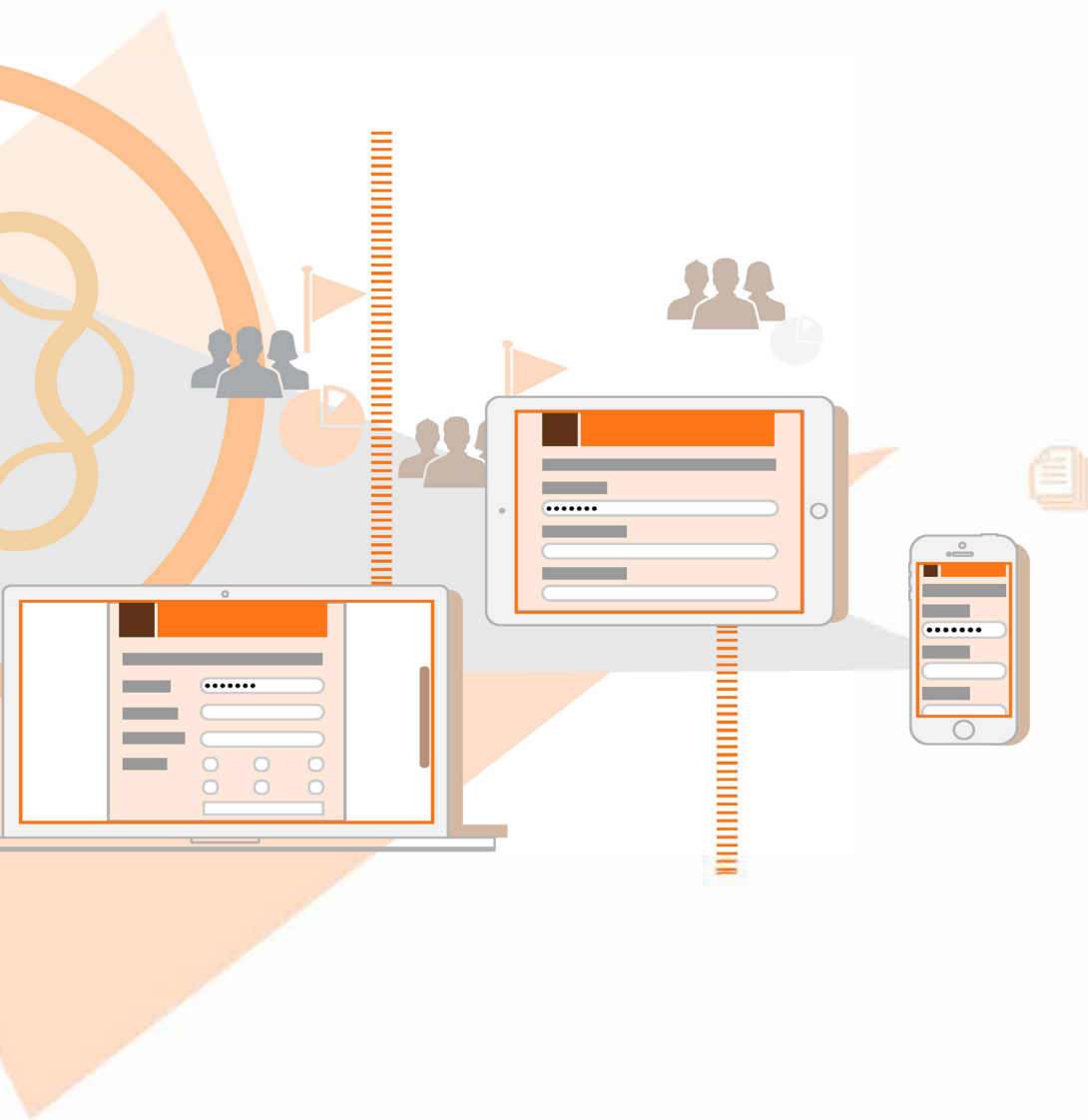# Preparing to install AEM Forms (Single Server)

AEM 6.3 Forms

Legal notices

For legal notices, see http://help.adobe.com/en_US/legalnotices/index.html.

Contents

1. About This Document

AEM Forms on JEE is an enterprise server platform that helps you automate and streamline business processes. AEM Forms comprises the following components:

• J2EE-based Adobe Experience Manager Forms provides server capabilities and runtime environment

• Tools to design, develop, and test AEM Forms on JEE applications

• Modules and Services are deployed on AEM Forms on JEE server and provide functional services

This document is part of a larger documentation set available at AEM Forms on JEE Documentation page. It is advised that you start with the preparing guide and then move on to installation and configuration guide depending on whether you are performing a fresh installation (single server or cluster setup) or upgrading your existing deployment. For Turnkey deployment, which is only for evaluation purposes, see Installing and Deploying AEM forms on JEEusingJBossTurnkey.

1.1. Conventions used in this document

The following naming conventions are used for common file paths.

Name	Description	Default value
[aem-forms root]	The installation directory that is used for all AEM Forms on JEE modules. The installation directory contains subdirectories for Configuration Manager, the SDK, and each AEM Forms on JEE module installed (along with the product documentation). This directory also includes directories relating to third-party technologies.	Windows: C:\Adobe\Adobe_Experience_Manager_Forms AIX, Linux, and Solaris: /opt/adobe/Adobe_Experience_Manager_Forms

Name	Description	Default value
[appserver root]	The home directory of the application server that runs the services that are part of AEM Forms.	JBoss on Windows: C:\jboss JBoss on Linux and Solaris: /opt/jboss JBoss Enterprise Application Platform on Windows: C:\jboss-eap-<*version*>\jboss-as JBoss Enterprise Application Platform on Linux and Solaris: /opt/jboss-eap-<*version*>/jboss-as WebSphere on Windows: C:\Program Files\IBM\WebSphere\AppServer WebSphere on Linux and Solaris: /opt/IBM/WebSphere/AppServer WebSphere on AIX: /usr/IBM/WebSphere/AppServer, or, /opt/IBM/WebSphere/AppServer WebLogic on Windows: C:\Oracle\Middleware\wlserver_<*version*> WebLogic on Linux and Solaris: /opt/Oracle/Middleware/wlserver_<*version*>
[WL_Home]	The install directory for WebLogic as specified for the `WL_HOME` environment variable.	WebLogic on Windows: C:\Oracle\Middleware\ WebLogic on Linux and Solaris: /opt/Oracle/Middleware
[appserverdomain]	The domain that you configured on WebLogic. The default domain is called *base_domain*.	WebLogic on Windows: C:\Oracle\Middleware\user_projects\domains\base_domain WebLogic on Linux and Solaris: /opt/Oracle/Middleware/user_projects/domains/base_domain
[WebSphere ND root]	The install directory for WebSphere Application Server, Network Deployment	WebSphere on Windows: C:\Program Files\IBM\WebSphere\AppServer WebSphere on Linux and Solaris: /opt/IBM/WebSphere/AppServer WebSphere on AIX: /usr/IBM/WebSphere/AppServer
[server name]	The name of the server configured on your WebLogic or WebSphere server	WebLogic: server1 WebSphere: server1
[profile_name]	The profile name for the JBoss application server.	Adobe pre-configured JBoss: domain_<*db-name*> Manual or Downlaoded Jboss: standard

Name	Description	Default value
[dbserver root]	The location where the database server is installed	Depends on the database type and your specification during installation

Most of the information about directory locations in this document is cross-platform (all filenames and paths are case-sensitive on AIX, Linux, and Solaris). Any platform-specific information is indicated as required.

1.2. Additional information

The resources in this table can help you learn about AEM Forms on JEE.

For information about	See
Introduction to AEM Forms	Overview
All documentation that is available for AEM Forms on JEE	Documentation

2. System requirements

2.1. Prepare your server environment

Do the following tasks to prepare your server environment:

1) Read the AEM Forms on JEE Supported Platforms document and ensure that your software, hardware, operating systems, application server, databases, JDKs, and other infrastructure are compliant.

2) Install and configure the operating system, and update with the necessary patches and service packs.

3) Install and configure database server.

4) Install and configure the application server.

2.2. Privileges required to install on Windows

When installing on Windows, you must use an account that has administrator privileges. If you run the installer using a non-administrator account, enter the credentials of an account that has administrator privileges. Turn off the UAC to run the installation and configuration process.

2.3. Synchronizing clock times

You must ensure that all computers in a horizontal cluster synchronize their clock times regularly. Your AEM Forms on JEE installation may encounter problems if the node times differ by more than a few seconds.

Apply the standard time synchronization practices employed by your network to all computers of the AEM Forms on JEE cluster.

2.4. (Optional) Additional system requirements

Certain capabilities/platforms have a few additional requirements. If you are using AEM Forms for these capabilities/platforms, see *Appendix-Additional system requirements* for detailed information about these additional requirements:

•   Linux- and UNIX-based platforms

•   PDF Generator

•   Central Migration Bridge service

•   AEM Forms IPv6 support

•   Connectors for IBM File Net, Documentum, and IBM Content Manager

•   Forms, Output, and ConvertPDF services

- AEM Forms on JEE with a Luna HSM cluster
- LDAP Configuration
- Processes with document form variables and digital signatures
- *AEMFormsCredentials and Certificates*

3. AEM Forms Credentials and Certificates

This section describes how to do the following tasks:
- Obtain the Document Security Rights credential.
- Obtain digital certificates for use with Digital Signatures.

3.1. Obtaining the Acrobat Reader DC extensions Rights credential

The Acrobat Reader DC extensions Rights credential is a digital certificate that is specific to Acrobat Reader DC extensions that enables Adobe Reader usage rights to be activated in PDF documents. If the credential is not installed, Acrobat Reader DC extensions users cannot apply usage rights to documents. You cannot use a standard digital certificate for this function; you must use the dedicated Rights credential.

The Rights credential extends the usage rights of each PDF file that Acrobat Reader DC extensions processes. It is a critical part of the software licensing and should be stored carefully in a secure environment.

The following types of Rights credentials are available:

Customer Evaluation: A credential with a short validity period that is provided to customers who want to evaluate Reader Extensions. Usage rights applied to documents using this credential expire when the credential expires. This type of credential is valid only for two to three months.

Production: A credential with a long validity period that is provided to customers who purchased the full product. Production credentials are unique to each customer but can be installed on multiple systems.

The Rights credential is delivered as a digital certificate that contains the public key, the private key, and the password used to access the credential.

If your organization orders an evaluation version of Acrobat Reader DC extensions, you receive an evaluation Rights credential from the sales representative you ordered the product from or from the website where you downloaded the evaluation product.

If your organization purchases a production version of Reader Extensions, the production Rights credential is delivered by Electronic Software Download (ESD). A production Rights credential is unique to your organization and can enable the specific usage rights that you require.

If you obtained Acrobat Reader DC extensions through a partner or software provider who integrated Acrobat Reader DC extensions into their software, the Rights credential is provided to you by that partner who, in turn, receives this credential from Adobe.

NOTE: The Rights credential cannot be used for typical document signing or assertion of identity. For these applications, you can use a self-sign certificate or acquire an identity certificate from a Certificate Authority (CA).

3.2. Obtaining digital certificates for use with Digital Signatures

Digital certificates are required for use with Digital Signatures. Although, you can configure and manage digital certificates after you install and configure AEM Forms, obtaining them before you install ensures that you are ready to use AEM Forms when it is deployed.

Digital certificates are obtained from a Certificate Authority (CA) and sent to you by email or over the web as a certificate file. This certificate file contains the public keys (also called *certificates*) and references to private keys (also called *credentials*) that are used for encrypting and signing documents. Certificates do not contain actual private keys; instead, they contain a reference to the identity of the user who keeps the private keys securely stored in an encrypted file or HSM.

You can use Internet Explorer (Windows) or OpenSSL (non-Windows operating system) to export PFX, P12, and CER files for certificates that are stored in any compatible certificate store that is available on your computer. PFX files can be exported only as the certificate store or the credential itself permits. CER files that hold the public key that corresponds to a credential can also be exported from PFX files by using either Internet Explorer or OpenSSL.

NOTE: You can configure and manage certificates, credentials, and Certification Revocation Lists (CRLs) for use with AEM Forms by using Trust Store Management, which is accessible through the web-based administration console. (See administration help.)

The CRL distribution point describes where you can download the CRL that corresponds to a particular CER or PFX file.

The following file types are supported:

Certificates: DER-encoded X509v3 and base64-encoded certificate (.cer) files. Certificates that verify the trust.xml file can be either DER-encoded or base64-encoded.

Credentials: RSA and DSA credentials up to 4096 bits in standard PKCS12 format (.pfx and .p12 files).

CRLs: Base64-encoded and DER-encoded CRLfiles.

Maintaining the security of private keys (credentials) is critical to ensuring the stability of sensitive information. A physical storage device, often called a *Hardware Security Module* (HSM), typically provides the maximum level of security for private keys. If you do not use a physical device, it is important to store highly sensitive private keys and certificates in encrypted files in a safe place.

Digital Signatures supports the industry-standard PKCS #11 interface to communicate with HSMs. An HSM vendor can provide the resources and tools that you need to install and configure an HSM storage system.

4. Create the AEM Forms Database

4.1. Database configuration requirements

This section describes how to set up the database for use with AEM Forms. This section describes special tables, indexes, and other properties that are required in the AEM Forms database that are not configured by Configuration Manager. The section has instructions for all the supported database; perform the instructions only for your database:

The database will contain these elements:

• AEM Forms services

• AEM Forms run-time configurations

• AEM Forms process data

• Customer process definitions and templates

• Application server managed data

Before you create the database, ensure that you read the pre-installation requirements and have the required software installed.

4.2. Minimum database user permissions

Database	Initialization permissions	Runtime permissions
Oracle	CREATE SESSION CREATE TABLE CREATE VIEW CREATE SEQUENCE UNLIMITED TABLE SPACE	CREATE SESSION UNLIMITED TABLE SPACE (only needed if you do not configure user quotas) CREATE TABLE

Database	Initialization permissions	Runtime permissions
MySQL	SELECT INSERT UPDATE DELETE CREATE DROP REFERENCES INDEX ALTER CREATE_TEMP_TABLE LOCK_TABLES	SELECT INSERT UPDATE DELETE
SQL Server - DB level	Create Table Create View Connect	Connect
SQL Server - Schema level	Alter Insert References Select Update Delete	Insert Select Update Delete
DB2	See *DB2useraccount* for a complete description	See *DB2useraccount* for a complete description

4.3. Creating an Oracle database

If you prefer not to use the default database that was created when you installed Oracle, create a new database by using the Database Configuration Assistant tool.

NOTE: You can use the Transaction Processing or General Purpose templates while configuring an Oracle database instance for AEM Forms. If you wish to use the Custom Database template for configuring a database instance, the minimum set of database components you must include are **Oracle JVM** and **Enterprise Manager Repository**.

Do the following when you create your Oracle database:

- Set the initial database size to a minimum of 500MB.
- Create user quotas to allow the database to grow to accommodate persistent data from applications.
- Enable support for UTF-8 encoding.
- Set the Database Character Set to Unicode (AL32UTF8), and the National Character Set to AL16UTF16 (Unicode UTF-16 universal character set).

- Set NLS_LENGTH_SEMANTICS to BYTE (if required). The database initialization fails if you set any other value.
- You must install Oracle using Transaction Processing and set the connection mode for the server to Dedicated Processing.

User account and rights

Create a new user account on the database and assign it the following system privileges:
- CREATE SEQUENCE
- CREATE VIEW
- UNLIMITED TABLESPACE
- CREATE TABLE
- CREATE CLUSTER
- CREATE SESSION

NOTE: For deployments on non-Windows operating systems, the username must not exceed eight characters; on Windows, it must not exceed 12 characters.

You need the following information when you configure the data source on the application server:
- SID (Service ID)
- Username and password of the Oracle user account
- Host name or IP address of the database server
- Oracle LISTENER port number (default is **1521**)

For information about using Oracle, see the appropriate Oracle's user documentation.

4.4. Creating an SQL Server database

You can create an SQL Server database that AEM Forms will use to store run-time and configuration data. For information about creating an SQL Server database, refer to the SQL Server documentation.

Create an SQL Server database, and create a user account and assign it DB_OWNER privileges for use when configuring the data source on the application server. For information about creating the database and user, see the SQL Server documentation.

You need the following information when you configure the data source on the application server:
- Database name
- Username and password of the SQL Server user account
- Host name or IP address of the database server
- SQL Server port number

Set up the SQL Server for AEM Forms

Before you create the AEM Forms database, optimize the SQL Server by changing these settings.

Increase memory

The default SQL Server settings do not aggressively allocate memory. This situation significantly affects performance on most deployments of an SQL Server database.

NOTE: This section is recommended but optional.

1) Using Microsoft SQL Server Management Studio, connect to the database server where you will host the AEM Forms database.

2) Right-click the database server connection and select **Properties**.

3) Select the **Memory** page and enter a size in the **Minimum Server Memory (in MB)** box that is equal to the size of the free memory on the server.

4) Restart the SQL Server database.

Set the processor priority

On dedicated database servers, which are recommended for production installations of AEM Forms, the SQL Server process is configured so that it does not consume too much of the system CPU resources.

NOTE: This section is recommended but optional.

1) Using Microsoft SQL Server Management Studio, connect to the database server where you will host the AEM Forms database.

2) Right-click the database server connection and select **Properties**.

3) Select the **Processors** page and select **Boost SQL Server Priority**.

4) Restart the SQL Server database.

Increase the recovery interval

This setting specifies the amount of time the deployment waits for recovery after a crash. The SQL Server default setting is one minute. Increasing this setting to a larger value improves performance because it causes the server to write changes from the database log to the database files less frequently. This setting does not compromise the transactional behavior; however, it affects the size of the log file that is replayed on startup.

NOTE: This section is recommended but optional.

1) Using Microsoft SQL Server Management Studio, connect to the database server where you will host the AEM Forms database.

2) Right-click the database connection and select**Properties**.

3) Select the **Database Settings** page and type 5 in the **Recovery Interval (Minutes)** box.

4) Restart the SQL Server database.

Integrated security

NOTE: This is an optional configuration.

If you are using SQL Server integrated security, you can set your SQL Server database to Mixed Mode or Windows Authentication Mode. However, if you are using Windows Authentication Mode, you must configure integrated security on Windows to establish a trusted connection with the SQL Server.

• For JBoss, see *ConfigureIntegratedSecurityonWindows* in Preparing to Install AEM forms on Single-Server.

Sizing your SQL Server database

The default database sizes that SQL Server provides are too small for AEM Forms. Even if the database is set to auto-grow, unintended effects can occur, such as reduced performance when the database grows or the growth begins to fragment the disk. It is best to preallocate the database size at creation to reflect your deployment requirements:

Medium-size deployments: Environments where the LDAP directory has approximately 100,000 users and 10,000 groups. Set Database Data Initial Size to 1GB, and set autogrowth to 250MB.

Large-size deployments: Environments where the LDAP directory has approximately 350,000 users and more than 10,000 groups. Set Database Data Initial Size to 2GB, and set autogrowth to 1GB.

NOTE: Database growth is always restricted to a certain size. Administrators should monitor the resource usage of the AEM Forms database to ensure that it does not lose its restricted space or the space available on the disks where the database resides.

Creating the AEM Forms database user, schema, and login

You are now ready to create the AEM Forms database user, schema, and login.

IMPORTANT: Ensure that you use the SQL_Latin1_General_CP1_CI_AS collation (or the Japanese_CI_AS collation if your database will run in a Japanese environment) when you create the database instance for AEM Forms. Any other collation may cause your database initialization to fail. The collation for your AEM Forms database instance can be different from the collation used when creating the SQL Server database.

1) Using Microsoft SQL Server Management Studio, click **Server**, and then right-click **Database** and select **New Database**.

2) Enter the database name of your choice.

   NOTE: The database name is very important, and the name chosen must be consistently used in the following procedures where a reference to `database_name` *exists.*

3) In the **Database Data Initial Size MB** box, enter the appropriate value:
   – For small development or small production systems, specify 200MB.
   – For larger systems, see *SizingyourSQLServerdatabase*.

4) In the **Database Data Autogrowth** box, enter 50%.

5) In the **Database Log Initial Size** box, enter the appropriate value:
   – For small development or small production systems, specify 20MB.
   – For larger systems, see *SizingyourSQLServerdatabase*.

6) In the **Database Log Autogrowth** box, enter 50%.

7) Click **OK** to create the database.

Create the AEM Forms user in SQL Server

In the following procedure, *[database_name]* represents the name you specified when you created your database, and *[database_username]* represents the name you must specify for the new user.

1) Using Microsoft SQL Server Management Studio, connect to the database server where you created the AEM Forms database.

2) Click **Server** > **Security**, and then right-click **Logins** and select **New Login**.

3) Enter the login name*[database_username]*, and then select **SQL Server Authentication**and type a new password.

4) Ensure that **Enforce Password Expiration, User must change password on next login** is also deselected.

5) Leave the default database as **Master**, and click **OK**.

6) Click **Server** > **Databases** > *[database_name]* > **Security**, and then right-click **Schemas** and select **New Schema**.

7) In the **Schema Name** box, type *[database_username]*, and click **OK**.

8) Click **Server** > **Databases** >*[database_name]* > **Security**, and then right-click **Users** and select **New User**.

9) In the New User dialog box, type the login name and username *[database_username]*.

10) Set the default schema to *[database_username]* and click **OK**.

   NOTE: The schema name should be the same as the *[database_username].*

11) Click **Server** > **Databases** > [*database_name*] > **Security**, right-click the *[database_username]* schema, and select **Properties** > **Permissions**.

   a) Click **Search** in Users or Roles and type [*database_username*] and click **OK**.

   b) In the **Explicit** tab, grant the following permissions:
   - Alter
   - Insert
   - Reference
   - Select
   - Update
   - Delete

   c) Click **OK**.

12) Right-click **Server** > **Databases** > [*database_name*], right-click the [*database_username*] schema, and select **Properties > Permissions**.

   a) In the **Explicit** tab, grant Create Table, Create View, and Connect permissions.

   b) Click **OK**.

Associate the AEM Forms user with the database

After you create the AEM Forms user, associate it with the AEM Forms database.

1) Click **Security** > **Logins**, and then right-click *[database_username]* and select **Properties**.

2) In Login Properties, on the General page, set the user's default database to *[database_name]*.

3) Select the **User Mapping** page and, in the Users Mapped To This Login section, verify that *[database_name]* is selected, **User** is set to *[database_username]*, and **Default Schema** is set to *[database_username]*.

4) Ensure that *[database_name]* is selected in the Users Mapped To This Login table, and ensure that **public**is selected in the **Database Role Membership For** *[database_name]* table and then click **OK**.

Set the isolation level for the AEM Forms database

AEM Forms requires a specific isolation level to manage deadlocking. The deadlocking occurs when long-running transactions and numerous shorter reads occur at the same time.

IMPORTANT: You must set the isolation level for MS SQL Server to avoid deadlocking issues.

1) Click **Databases**, and then right-click *[database_name]* and select **New Query**.

   NOTE: *[database_name] represents the name you specified when you created your database.*

2) In the Query panel, type the following text:

```
ALTER DATABASE [database_name]
SET READ_COMMITTED_SNAPSHOT ON
GO
```

3) Click **Execute**. A response is displayed in the messages panel.

4.5. Creating a MySQL database

NOTE: The information contained in this document is meant for users who are installing MySQL manually and not for a turnkey installation. See Installing and DeployingAEMformsforJBossusingTurnkey.

Use the MySQL tools to create a MySQL database for use with AEM Forms and a MySQL user account that the application server can use to connect to the database. You also must modify the MySQL database server configuration. For information about creating the database and user account, see the MySQL documentation.

NOTE: MySQL does not support the use of special characters or spaces in the username or password. Ensure that your username and password adhere to this restriction. Also, to avoid Scheduler errors, do not use a dash (-) in the MySQL database name.

NOTE: On non-Windows operating systems, you must set the lower_case_table_names system variable to 1 to ensure that table names are case-insensitive. On Windows systems, this parameter does not have any effect. For more information about setting table names for case-sensitivity, see http://dev.mysql.com/doc/refman/5.1/en/identifier-case-sensitivity.html.

NOTE: To set up a database schema and new users on MySQL by using a graphical user interface (GUI), you must install the MySQL Administrator tool.

When installing the MySQL database, you must specify UTF-8 character encoding in the Variables section of the MYSQL UI.

You need the following information when you configure the data source on JBoss:

- Username and password of the MySQL user account
- Host name or IP address of database server
- MySQL port number

MySQL user account

The MySQL user account that you create must have these privileges to access the tables in the AEM Forms database:

- SELECT
- INSERT
- UPDATE
- DELETE
- CREATE
- DROP
- REFERENCES
- INDEX
- ALTER
- CREATE_TMP_TABLE
- LOCK_TABLES

MySQL initial server configuration requirements

The following configuration setting is required in the `mysqld` section of the my.ini file (Windows) or /etc/my.cnf file (non-Windows operating system):

- max_allowed_packet=64M

    NOTE: Failure to configure this setting will result in "packet too large" errors that will cause module deployment to fail.

    These configuration settings are recommended in the `mysqld` section of the my.ini file (Windows) or /etc/my.cnf file (non-Windows operating system):

- key_buffer_size=16M
- port=3306
- socket = /tmp/mysql.sock
- skip-external-locking
- table_open_cache = 64
- innodb_log_file_size=170M
- sort_buffer_size = 512K

- net_buffer_length = 8K
- read_buffer_size = 1M
- read_rnd_buffer_size = 1M
- myisam_sort_buffer_size = 35M

5. Configure a JBoss Application Server

5.1. Configuration options

When using a JBoss Application Server, you have the following installation options:

- Adobe preconfigured JBoss

- Manual JBoss configuration (See *Appendix-ManuallyConfiguringJBoss*.)

This section covers the use of the Adobe preconfigured JBoss Application Server when you are not using the turnkey installation option. You need to use the non-turnkey method for any of the following scenarios:

- You are using a Linux or Solaris operating system.

- You are using Oracle or SQL Server, or if you do not want to use the MySQL that is installed by default in a turnkey setup (Partial Turnkey).

- You need advanced control of the JBoss configurations.

Adobe-preconfigured JBoss

The Adobe-preconfigured JBoss software is delivered as part of the AEM Forms on JEE media. This is the simplest option for installing JBoss because advanced knowledge of JBoss is not required.

If you intend to download a supported version of JBoss from the Internet, you must follow the configuration steps in *Appendix-ManuallyConfiguringJBoss*.

Adobe downloads JBoss archive from the Red Hat website and configures it with the options required to run AEM forms on JEE in a stand-alone configuration. You can review these configurations in the next section.

NOTE: JBoss deployment has distinct configuration files for the data sources for each database type under [appserver root]/standalone/configuration. Each database is prefixed with "lc_". Thus, data source files for MySQL, Oracle, and MS SQL databases will be lc_mysql.xml, lc_oracle.xml, and lc_sqlserver.xml, respectively. In the following sections, lc_<db-name> is used to designate these distinct files.

If you download the Electronic Software Distribution (ESD) from the Adobe website, extract the aemforms_server_6_3_0_jboss_all_win.zip (Windows) or aemforms_server_6_3_0_j-boss_all_unix.tar.gz (Linux or Solaris) file to your file system. After the main archive file is extracted, extract the following directories: merge-modules, server, and third_party.

NOTE: Be sure to keep the directory hierarchy unchanged from the original ESD file.

The pre-configured JBoss software is located on the installation media or ESD in the /third_party directory. The jboss.zip file contains the Adobe pre-configured JBoss.

Depending on your installation requirements, copy the JBoss directories and their contents to the location where you intend to install JBoss. The JBoss installation has all the configuration steps completed, except for the steps detailed in *Database connectivity for Adobe pre-configured JBoss*, which you must complete if you plan to use MySQL. If you do not intend to use a MySQL database, you must complete the appropriate configuration steps for your database detailed later in this chapter.

NOTE: The preconfigured JBoss includes a number of JSafe JAR files, which provide cryptography and Public Key Infrastructure (PKI) related FIPS-certified implementations in AEM Forms on JEE. When you install the product files, these files are deployed in the [*appserver root*]/standalone/configuration directory. Depending on the class loading, these JSafe JARs are used, even if your applications have their own version of JSafe JAR files.

NOTE: The standalone.sh file contains startup parameters for the application server running on the Linux operating system. Ensure that the file contains all the required parameters and opening/closing quotes (" ") are in-place for all the parameters.

5.2. Install JDK for JBoss

You must download and install Oracle JDK 8.0 or later updates to 8.0 versions. After installing JDK,

1) Create or set the `JAVA_HOME` environment variable to point to the location where the Java JDK is installed.

2) Set the PATH environment variable to point to JAVA_HOME/bin.

3) (Solaris Only) Modify the standalone.conf file.

   *For JVM to run in 64-bit mode, add the **-D64** JVM argument to the following line in the [appserver root]/bin/standalone.conf file.*

   ```
   JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx4096m -XX:MaxPermSize=768m
   -Dorg.jboss.resolver.warning=true
   -Dsun.rmi.dgc.client.gcInterval=3600000
   -Dsun.rmi.dgc.server.gcInterval=3600000
   -Dsun.lang.ClassLoader.allowArraySyntax=true"
   ```

Logon scripts for Linux and Solaris

To prepare the environment for production usage, it is recommended for production use cases that the environment be set up automatically during user login. This requires that the `JAVA_HOME` and `PATH` environment variables are set in the users login scripts for the user that the JBoss process will run as.

5.3. Database connectivity for Adobe pre-configured JBoss

To configure a connection to the AEM Forms on JEE database, you must complete the following tasks:

•   Configure the AEM Forms on JEE data source.

•   Configure JBoss to use your database as the default data source.

You must configure the data source to connect to the database. For JBoss, you can configure a MySQL, SQL Server, or Oracle data source.

NOTE: Before proceeding with the following tasks, ensure that Jboss is not running.

Encrypt database password

Encrypt the password in the lc_<db-name>. file. Use the following command to encrypt the password:

```
java -cp
%JBOSS_HOME%\modules\system\layers\base\.overlays\layer-base-jboss-eap-6.
4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar
org.picketbox.datasource.security.SecureIdentityLoginModule <password>
```

NOTE: If you do not have the picketbox-4.1.2.Final-redhat-1.jar file, which is used to encrypt datasource passwords for JBoss, then download it from Index of /techpreview/all/org/picketbox/pick-etbox/4.1.2.Final-redhat-1. Do not use the JAR file available at http://wiki.jboss.org.

Configuring MySQL for Adobe pre-configured JBoss

To enable Adobe pre-configured JBoss to connect to a database that stores AEM Forms on JEE data, you must create a data source file and deploy it to the instance of JBoss where you will deploy AEM Forms on JEE.

Encrypt the password in the lc_<db-name>.xml file. Use the following command to encrypt the password:

```
java -cp
%JBOSS_HOME%\modules\system\layers\base\.overlays\layer-base-jboss-eap-6.
4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar
org.picketbox.datasource.security.SecureIdentityLoginModule <password>
```

NOTE: If you do not have the picketbox-4.1.2.Final-redhat-1.jar file, which is used to encrypt datasource passwords for JBoss, then download it from Index of /techpreview/all/org/picketbox/pick-etbox/4.1.2.Final-redhat-1. Do not use the JAR file available at http://wiki.jboss.org.

Edit the lc_<db-name>.xml file

Before configuring the MySQL data source, you must have already created the database on MySQL as described in *CreatingaMySQL database*.

1) Open the *[appserver root]*\standalone\configuration\lc_<db-name>.xml file in a text editor and locate this line for IDP_DS, AEM_DS, and EDC_DS:

```
<connection-url>
|jdbc:mysql://localhost:3306/adobe
</connection-url>
<driver>mysql</driver>
```

2) Replace the following text in the file with values that are specific to your database:

   – **localhost:** The name, IP address, or fully-qualified path of the computer that hosts the database. The default is `localhost`.

- **3306:** The port used to access the database. The default port is `3306`.
- **adobe:** The name of the database that stores the data. Replace the default value, `adobe`, with your database name.

3) Ensure that the minimum and maximum values for the data source connections are set as follows:
- For IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

- For EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

- For AEM_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

NOTE: If your AEM Forms on JEE server handles heavy load, increase the maximum number of JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for IDP_DS, AEM_DS, and EDC_DS.

4) Save and close the file.

Edit the lc_<db-name>.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_<db-name>.xml file in a text editor and modify the following code within the `<authentication>` element:

```
<security-domain name="EncryptDBPassword">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
                        </login-module>
                    </authentication>
                </security-domain>
    <security-domain name="EncryptDBPassword_IDP_DS">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                        </login-module>
```

```
                </authentication>
            </security-domain>
    <security-domain name="EncryptDBPassword_AEM_DS">
                <authentication>
                    <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                        <module-option name="userName" value="adobe"/>
                        <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                    </login-module>
                </authentication>
            </security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
                <authentication>
                    <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                        <module-option name="userName" value="adobe"/>
                        <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
                    </login-module>
                </authentication>
            </security-domain>
```

2) Specify the values that are specific to your database so that the application server can access your database.

3) Save and close the file.

4) Start JBoss.

Configuring Oracle for Adobe-preconfigured JBoss

To enable JBoss to connect to the Oracle database that stores AEM Forms on JEE data, you must create a data source file and deploy it to the instance of JBoss where you will deploy AEM Forms on JEE.

Edit the lc_<db-name>.xml file

1) Open the *[appserver root]*\standalone\configuration\lc_<db-name>.xml file in a text editor and locate this line:

```
<connection-url>jdbc:oracle:thin@localhost:1521:adobe</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
```

2) Replace the following text from the above line with values that are specific to your database:

– **localhost:** The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost.

- **1521:** The port used to access the database. The default port is `1521`.
- **adobe:** The SID of the database that stores the AEM Forms on JEE data. Replace the default value, `adobe`, with your database SID.

NOTE: If you are using a pluggable database, replace the colon (:) after <port> with a forward slash (/).

3) In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the username and password that the application server uses to access your database.

4) Ensure that the minimum and maximum values for the data source connections are set as follows:

- For IDP_DS:
  ```
  <min-pool-size>1</min-pool-size>
  <max-pool-size>30</max-pool-size>
  ```
- For EDC_DS:
  ```
  <min-pool-size>1</min-pool-size>
  <max-pool-size>30</max-pool-size>
  ```
- For AEM_DS:
  ```
  <min-pool-size>1</min-pool-size>
  <max-pool-size>30</max-pool-size>
  ```

NOTE: If your AEM Forms on JEE server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for IDP_DS, AEM_DS, and EDC_DS.

5) Save and close the file.

Edit the lc_<db-name>.xml file

If you are running AEM Forms on JEE with an Oracle database, you must set Oracle as the default data source for JBoss. This procedure assumes that the Oracle JDBC driver is installed in the *[appserver root]*/standalone/configuration directory.

1) Open the *[appserver root]*/standalone/configuration/lc_<db-name>.xml file in a text editor and modify the `<datasource>` element with your Oracle connection settings:
   ```
   <jndi-name>DefaultDS</jndi-name>
   <connection-url>jdbc:oracle:thin@localhost:1521:adobe</connection-url>
   <driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
   <security-domain>OracleDbRealm</security-domain>
   ```

2) Replace the text in the above lines with values that are specific to your database:

   localhost:: *Replace this value with the host name of your Oracle server.*

   1521*: If Oracle is not using the default port, replace this value with the appropriate port number.*

   adobe*: Replace this value with your database SID.*

3) In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the username and password that the application server uses to access your database.

4) *(Only for Oracle RAC)* Replace the connection settings mentioned in the first step with the following connection URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS=
(PROTOCOL=TCP) (HOST=yourhost1) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP)
(HOST=yourhost2) (PORT=1521)) (LOAD_BALANCE=on) (FAILOVER=on))
(CONNECT_DATA=(SERVER=dedicated) (SERVICE_NAME=service.yourcompany.com)
(FAILOVER_MODE=(TYPE=session) (METHOD=basic) (RETRIES=10) (DELAY=3))))
```

5) **NOTE:** *Ensure that this entry appears as a single line in the oracle-ds.xml file.*

6) *(Only for Oracle RAC)* Replace the following text from the connection URL in the previous step with values that are specific to your database:

   – **yourhost1:** The name, IP address, or fully-qualified domain name of the first node in the cluster that hosts the database.

   – **yourhost2:** The name, IP address, or fully-qualified domain name of the second node in the cluster that hosts the database.

     NOTE: The cluster hosting the database could have *n* nodes. **yourhost1** and **yourhost2**are examples in the case of a two-node cluster.

   – **service.yourcompany.com:**The service name for the Oracle RAC database.

7) Save and close the file.

Edit the lc_<db-name>.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_<db-name>.xml file in a text editor and modify the following code within the `<authentication>` element:

```
Replace  :

<security-domain name="EncryptDBPassword">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="5c2f412d6fa61722"/>
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
                        </login-module>
                    </authentication>
                </security-domain>
        <security-domain name="EncryptDBPassword_IDP_DS">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="5c2f412d6fa61722"/>
```

```
                        <module-option name="managedConnectionFactoryName"
     value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                          </login-module>
                    </authentication>
                </security-domain>
<security-domain name="EncryptDBPassword_AEM_DS">
                    <authentication>
                    <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                        <module-option name="userName" value="adobe"/>
                         <module-option name="password"
value="5c2f412d6fa61722"/>
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                          </login-module>
                    </authentication>
                </security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
                    <authentication>
                    <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                        <module-option name="userName" value="adobe"/>
                         <module-option name="password"
value="5c2f412d6fa61722"/>
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
                          </login-module>
                    </authentication>
```

2) Specify the values that are specific to your database to enable the application server to access your database.

   NOTE: For Oracle RAC, replace **adobe** with the service name specific to your database.

3) Save and close the file.

4) Start JBoss.

Configuring SQL Server for Adobe-preconfigured JBoss

To enable JBoss to connect to the SQL Server database that stores AEM forms on JEE data, you must create an SQL Server data source file and deploy it to the instance of JBoss where you will deploy AEM Forms on JEE (for example, *[appserver root]*\standalone\configuration\).

Edit the lc_<db-name>.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_<db-name>.xml file in a text editor and locate this line:

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
```

2) Replace the following text from the above line with values that are specific to your database:
   – **localhost**: The name, IP address, or fully-qualified path of the computer that hosts the database. The default is `localhost`.
   – **1433**: The port used to access the database. The default port is `1433`.
   – **adobe**: The name of the database that stores the AEM Forms on JEE data. You will need to update the default value, `adobe`, with your database name.

3) In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the username and password that the application server uses to access your database.

4) Ensure that the minimum and maximum values for the data source connections are set as follows:
   – For IDP_DS:
```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```
   – For EDC_DS:
```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```
   – For AEM_DS:
```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

   NOTE: If your AEM Forms on JEE server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase`<max-pool-size>`to 50 or more for IDP_DS, AEM_DS, and EDC_DS.

5) Save and close the file.

Edit the lc_<db-name>.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_<db-name>.xml file in a text editor and modify the following code within the `<authentication>` element:

```
Replace:

<security-domain name="EncryptDBPassword">
                <authentication>
                    <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                        <module-option name="userName" value="adobe"/>
```

```
                              <module-option name="password"
value="5c2f412d6fa61722"/>
                          <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
                          </login-module>
                      </authentication>
                  </security-domain>
        <security-domain name="EncryptDBPassword_IDP_DS">
                          <authentication>
                              <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                                  <module-option name="userName" value="adobe"/>
                              <module-option name="password"
value="5c2f412d6fa61722"/>
                          <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                              </login-module>
                      </authentication>
                  </security-domain>
<security-domain name="EncryptDBPassword_AEM_DS">
                          <authentication>
                              <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                                  <module-option name="userName" value="adobe"/>
                              <module-option name="password"
value="5c2f412d6fa61722"/>
                          <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                              </login-module>
                      </authentication>
                  </security-domain>

<security-domain name="EncryptDBPassword_EDC_DS">
                          <authentication>
                              <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                                  <module-option name="userName" value="adobe"/>
                              <module-option name="password"
value="5c2f412d6fa61722"/>
                          <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
                              </login-module>
                      </authentication>
                  </security-domain>
```

2)   Specify the values that are specific to your database to enable the application server to access your database.

3) Save and close the file.

4) Start JBoss.

Map the Windows login to the AEM forms on JEE database user

1) Using Microsoft SQL Server Management Studio, connect to the database server that hosts the AEM Forms on JEE database.

2) Set the Authentication mode to **Windows authentication**.

3) Under **Security > Logins**, create a new account for the Windows domain user and select **Windows authentication**.

4) Click **User Mapping** on the Login - New screen and set the database and default schema for the new user.

5) Select **db_owner** as the Database role and click **OK**.

To verify that you created the user, expand the AEM Forms on JEE database in the tree and open **Security > Users**. The new user appears in the list of users.

Install the SQL Server database driver

NOTE: If you have not done so already, download the SQL Server JDBC Driver (for all platforms) from Microsoft Download Center or copy the driver from [DVD root]/third_party/ to the [appserver root]\modules\system\layers\base\com\microsoft\main directory.

NOTE: See the Supported Platforms for AEM Forms on JEE document for the version of supported drivers.

Configure integrated security on Windows

NOTE: You must run the JBoss service as the Windows user configured in *Map the Windows login to the AEM forms on JEE database user*. If JBoss is not running as a service, you must log in to Windows as that particular user before you start JBoss from a command prompt.

1) Modify the lc_<db-name>.xml files located in *[appserver root]*\standalone\configuration\, to add `integratedSecurity=true` to the connection URL as shown in this example:

   jdbc:sqlserver://<host-name>:<port>;databaseName=<db-name>;integratedSecurity=true.

2) Replace the values shown in bold in the above example with values appropriate to your database server.

3) Add the sqljdbc_auth.dll file to the Windows systems path (C:\Windows) on the computer that is running JBoss. The sqljdbc_auth.dll file is located within the Microsoft SQL JDBC 4.0 driver installation. The default location is `[SQL_root]/sqljdbc_4.0/enu/auth/x86` for 32-bit operating systems and `[SQL_root]/sqljdbc_4.0/enu/auth/x64` for 64-bit operating systems.

   NOTE: See the Supported Platforms for AEM Forms on JEE document for the version of supported drivers.

4) Open the properties for the JBoss for Adobe Experience Manager Forms 6.3 service or the JBoss service that you configured, and click the **Log On** tab.

5)   Select **This Account** and type the value of the user account that you mapped in *Map the Windows login to the AEM forms on JEE database user*. If you are running JBoss from the command line, you must be logged in as this user.

6. Configuring a WebSphere Application Server

This chapter describes how to install and configure the WebSphere Application Server that will host your AEM Forms on JEE installation.

6.1. Installing WebSphere

You must install WebSphere Application Server for running AEM Forms on JEE products. For use with AEM Forms on JEE, WebSphere can be installed either as WebSphere Base or as WebSphere ND with one base profile. See the WebSphere documentation to determine which method is best for your system.

Installing a Fix Pack to WebSphere

Before deploying AEM Forms on JEE, ensure that you install and update to the supported WebSphere version mentioned in the Supported Platforms.

JAVA_HOME and PATH environment variables

As part of your WebSphere installation, a Java SDK (JDK) was installed. The `JAVA_HOME` and `PATH` environment variables can point to the JDK where AEM Forms on JEE will be deployed.

Set the JAVA_HOME environment variable (Windows)

1) Select **Start** > **Control Panel**>**System**.
2) Click the **Advanced**tab and click **Environment Variables**.
3) In the System Variables area, click **New**.
4) Type`JAVA_HOME` as the variable name and, as its value, specify the directory where the JDK bundled with WebSphere is installed. For example, type the following path:

   C:\Program Files\IBM\WebSphere\AppServer\java\8.0

Set the PATH environment variable (Windows)

1) Select **Start** > **Control Panel**>**System**.
2) Click the **Advanced**tab and click **Environment Variables**.
3) In the System Variables area, select **Path**, click **Edit**, and then append the following text to the beginning of the variable value:

   `%JAVA_HOME%\bin;`

Set the DISPLAY setting (AIX)

• If you are using a Windows machine to access the AIX console, type the following text at the command prompt:

```
export DISPLAY= <IP Address of the Windows machine telnet, or cygwin to
Linux or Solaris machine>:0.0
```

NOTE: If the DISPLAY setting is not set correctly, the following error occurs when you start Configuration Manager: "No JRE is found...".

Set the JAVA_HOME environment variable (AIX, Linux, and Solaris)

*   Set the `JAVA_HOME` variable for Borne and Bash shells as shown in the following example:

    ```
    JAVA_HOME=/opt/IBM/WebSphere/AppServer/java/8.0
    export JAVA_HOME
    ```

    NOTE: The specific path varies based on the installation directory you specified and the operating system you are installing on.

Set the PATH environment variable (AIX, Linux, and Solaris)

*   Set the `PATH` variable for Borne and Bash shells as shown in the following example:

    ```
    PATH=$JAVA_HOME/bin:$PATH
    export PATH
    ```

(Windows, AIX, Linux, or Solaris) Verify JAVA_HOME environment variables

(Optional) Open a command prompt and run the following command:

```
java -version
```

You should receive a response that displays the Java version installed.

Starting WebSphere Base and accessing WebSphere Administrative Console

Start WebSphere Base

1) If the server is not already running, start the WebSphere Administrative Console, from *[appserver root]*/profiles/<profile_name>/bin, type the appropriate command, replacing **server1** with the name of your server instance:
    *   (Windows) `startServer.bat`**server1**
    *   (AIX, Linux, and Solaris) `./startServer.sh`**server1**

Access the WebSphere Administrative Console

1) Start the WebSphere Application Server.
2) In the address bar of a web browser, type the appropriate URL:

    ```
    http://[host name]:[port]/ibm/console
    ```

If you are logged into the server that you are currently using, you can replace *[hostname]* with `actual IP address or hostname`. The port value depends on the application server and whether Adminis-

trative Security is enabled. The default port value for WebSphere is 9060. If Administrative Security is enabled, the default SSL port value is 9043.

1) If WebSphere Administrative Security is enabled, type the WebSphere user ID and password in the boxes provided.

2) Click **Log In**.

Starting WebSphere ND and accessing WebSphere Administrative Console

1) Open a command prompt and navigate to *[WebSphere ND root]*/profiles/Dmgr01/bin.

2) **(Windows)** Run `startManager.bat`.

   (AIX, Linux, and Solaris) Run `startManager.sh`.

3) Navigate to *[WebSphere ND root]*/profiles/*[profile name]*/bin and run `startNode.bat`.

4) Once the services have started, connect to the server by typing the URL `http://[host name]:[port]/ibm/console` in the address bar of a web browser.

5) If WebSphere Administrative Security is enabled, type the WebSphere user ID and password in the boxes provided and click **Log In**.

6) In the navigation tree of the WebSphere Administrative Console, click **Servers > Server Types > WebSphere application servers**.

7) Select the listed server and click **Start**. Notice that the server status changes as the server is started.

Directory permissions

The AEM Forms on JEE application will extract files to the *[appserver root]*/installedApps directory. Therefore, it is important that writable permissions be given to that directory. If writable permissions cannot be given, the section below describes how to modify the location for the extracted files.

NOTE: It is recommended that you modify the location of the extracted files to [appserver root]/profiles/<profile_name>/installedApps.

Modify the location for the extracted files

1) Log in to the WebSphere Administrative Console.

2) Click **Servers** > **Server Types** > **WebSphere Application servers** and click your server name, such as **server1**.

3) Under Server Infrastructure, click **Java and forms workflow**>**Process Definition**.

4) Under Additional Properties, click **Java Virtual Machine** and then click **Custom Properties**.

5) Click **New** and create a custom property named `adobeidp.RootDirectory`.

6) Set the value of `adobeidp.RootDirectory` to the path where Adobe native files should be extracted, such as *[appserver root]*/profiles/*<profile_name>*/installedApps.

7) Click **OK** or **Apply**.

8) In the Messages box, click **Save directly to master configuration**, and then restart the application server.

6.2. Preparing WebSphere Application Server

This section describes how to prepare and configure an application server instance for your AEM Forms on JEE deployment.

Increase the SOAP request time out

1) Go to the *[appserver root]* directory and search for all files named soap.client.props. Multiple files may have this name. For example, on an AIX, Linux, or Solaris server, the following files exist:
   - *[appserver root]*/profileTemplates/default/documents/properties/soap.client.props
   - *[appserver root]*/profileTemplates/management/documents/properties/soap.client.props
   - *[appserver root]*/profiles/<profile_name>/properties/soap.client.props
   - *[appserver root]*/profiles/<profile_name>/temp/soap.client.props

2) Open each soap.client.props file in a text editor, find the `com.ibm.SOAP.requestTimeout` property, and change the value from `180` to `1800`.

3) Save and close each soap.client.props file.

4) In the navigation tree of the WebSphere Administrative Console, click **Servers** > **Server Types** > **WebSphere Application servers** and, in the right pane, click the server name.

5) Under Server Infrastructure, click **Administration**>**Administration Services**.

6) Under Additional Properties, click **JMX Connectors**, and then click **SOAPConnector**.

7) On the next screen, click **Custom properties**, and then click **requestTimeout**.

8) If necessary, change `600` to `1800` in the **Value** box on the next screen. Click **OK** or **Apply**.

9) In the **Messages** box, click **Save directly to master configuration**.

Increase the Deployer heap size

You must increase the heap size in the `ejbdeploy.bat/sh` script to avoid time-out errors.

NOTE: In WebSphere 8 and later versions, the ejbdeploy.bat/sh script is not installed by default. Therefore, you must install it separately to avoid errors during installation. For more information about adding the script, see EJBDeploy - OptionalfeatureinWebSphereApplicationServerVersion8.

AIX, Linux, or Solaris

1) Go to the *[appserver root]*/deploytool/itp/ directory and open `ejbdeploy.sh` for editing.

2) **(Solaris only)**In the `SunOS` section, find the `EJBDEPLOY_JVM_OPTIONS` attribute and change the value of the `-XX:PermSize` option to `256m`, and ensure that the value of the `-Xverify` option is `none`.

3) Add the heap size parameter at the end of the `JAVA_CMD \` section:
   `-Xms256m -Xmx4096M`

4) Save and close the file.

Windows

1) Go to *[appserver root]*\deploytool\itp and open the ejbdeploy.bat file in a text editor.

2) Find the line beginning with %JAVA_HOME% and then find the argument -Xmx.

3) Change the argument to -Xmx512M.

1) Save and close the file.

Configure inbound and outbound communication

NOTE: Perform these steps only if you have Global Security enabled on your WebSphere application server.

1) In WebSphere administration console, navigate to **Security > Global Security**.

2) In the Authentication section, click **RMI/IIOP security**.

3) Click **CSIv2 inbound communication** and set **Transport** to **SSL-supported**. Click **OK**.

4) In the Messages box, click **Save directly to master configuration**.

5) Click **CSIv2 outbound communication** and set **Transport** to **SSL-supported**.

6) Click **OK**.

7) In the Messages box, click **Save directly to master configuration**. Click **OK**.

6.3. Configuring the AEM Forms on JEE database connectivity

Next, configure the database connectivity and install AEM Forms on JEE by following the instructions provided in Installing and DeployingAEMformsonJEEforWebSphere.

6.4. Configuring WebSphere Application Server if global security is enabled

If your installation uses global security, you must run WebSphere Application Server as a user with the appropriate roles. You can employ one of the following options to configure WebSphere Application Server to run if WebSphere global security is enabled:

*   Create a new user with the necessary roles, and run WebSphere Application Server as that user. If a user already exists to run WebSphere Application Server, assign the necessary roles to that user.

    IMPORTANT: Ensure that you start WebSphere Application Server as this user. Some WebSphere processes may fail if you start WebSphere Application Server as a different user while global security is enabled.

    In a secure environment, it is recommended that you employ this option.

*   Configure the EVERYONE group with the necessary roles.

To create a new WebSphere Application Server user:

1) In the WebSphere Administrative Console navigation tree, click **Environment** > **Naming** > **CORBA Naming Service Users**, and then in the right pane, click **Add**.

2) In **Roles**, select all the roles.

3) Under Search and Select Users, select the User Realm.

4) In the search box, type the search string and click **Search**. *NOTE: To retrieve all users, type an asterisk (*).*

5) From the Available text box, select the required users and click the right arrow to add them to the Mapped to role box.

6) Click **Save directly to master configuration**.

To configure an existing WebSphere Application Server user:

1) In the WebSphere Administrative Console navigation tree, click **Environment** > **Naming**> **CORBA Naming Service Users**, and then in the right pane, select the user.

2) In **Roles**, select the required roles.

3) Click **OK**or **Apply**.

4) Click **Save directly to master configuration**.

To configure the EVERYONE group

1) In the WebSphere Administrative Console navigation tree, click **Environment** > **Naming** > **CORBA Naming Service Groups**.

2) In **Roles**, select the required roles.

3) Enable **Select from special subjects**, and then from the Special subjects list, select the **EVERYONE** group.

NOTE: If the EVERYONE group is already configured, the group will not be shown in the Special subjects list. You only need to assign the required roles to this group if not already done so.

1) Click **OK**or **Apply**.

2) Click **Save directly to master configuration**.

Configure CSIv2 inbound transport

On the default Global Security enabled installation of IBM WebSphere, CSIv2 inbound transport option is set to SSL-required. This configuration causes Output and Forms components to fail. Ensure that you change CSIv2 inbound transport option to SSL-Supported: To change the option:

1) Log in to IBM WebSphere administration console.

2) Expand **Security**, and then click **Global security**.

3) In the Authentication section, expand **RMI/IIOP security**, and then click **CSIv2 inbound communications**.

4) In CSIv2 Transport Layer section, set value of **Transport** to **SSL-Supported**.

5) Click **Apply**.

NOTE: After enabling global security and configuring appropriate roles for Application server, additional configurations are required to enable OSGi functionality built within the JEE stack. These configurations are performed after installing and configuring AEM Forms on JEE on WebSphere application server. For detailed steps, see Enable WebSphere Global Administrative Security.

7. Configuring a WebLogic Server

7.1. AEM Forms on JEE requirements for WebLogic

Two primary startup scenarios are available for WebLogic:

- Using Node Manager and configuring through the Administration Server (recommended)
- Directly starting the Managed Servers with configurations in the StartManagedWebLogic scripts

To use the automatic configuration features of Configuration Manager (the recommended option), you must use the Node Manager and configure through the Administration Server.

If you are directly starting the Managed Servers, you can still use the automatic configuration features of Configuration Manager. However, you must manually enter the Configuration Manager changes into the StartManagedWebLogic scripts. For the appropriate steps, see Installing and Deploying AEM Forms on JEE for WebLogic.

7.2. Installing WebLogic Server

Obtain Oracle WebLogic Server for your operating system and install it according to the manufacturer's documentation (see http://download.oracle.com/docs/cd/E14571_01/wls.htm).

NOTE: You must install and run WebLogic Server by using a user account that has computer administrator privileges.

Configuring WebLogic on Solaris (64-bit Java)

Additional steps need to be performed while installing WebLogic to use 64-bit Java on Solaris.

1) Run the WebLogic installer using the `-d64` flag.
2) From a new command prompt, navigate to *[appserver_root]*/server/bin directory
3) Type `export JAVA_OPTIONS=-d64`.
4) Start Node Manager by using the script; type `startNodeManager.sh`.

NOTE: You must start the WebLogic Administration Server using 64-bit Java. Before starting the WebLogic Administration server using `startWeblogic.sh` script, ensure that `JAVA_OPTIONS=-d64` is set in the environment.

7.3. Installing JDK

Install JDK

Depending on your operating system, obtain and install the Oracle JAVA for your operating system. For information on the supported version, see Supported PlatformCombinations.

The `JAVA_HOME` and `PATH` environment variables must point to the JDK on the server where AEM Forms on JEE will be deployed.

Set the JAVA_HOME environment (Windows)

1) Select **Start** > **Control Panel**>**System**.

2) Click the **Advanced**tab and click **Environment Variables**.

3) In the System Variables area, click **New**.

4) Enter `JAVA_HOME` as the variable name and the directory where you installed the Java SDK. This directory is where WebLogic installed the Java SDK that contains the /bin subdirectory. For example, type this text:

```
C:\Program Files\Java\jdk1.8.0_74
```

NOTE: If you are using PDF Generator to convert OpenOffice.org files, you must set the `JAVA_HOME_32` environment variable to point to the directory where the Oracle JDK is installed. Download the Oracle JDK from http://www.oracle.com/technetwork/java/index.html.

Set the PATH environment variable (Windows)

1) Select **Start** > **Control Panel**>**System**.

2) Click the **Advanced**tab and click **Environment Variables**.

3) In the System Variables area, select the `PATH` variable and then click **Edit**.

4) Append the following text to the beginning of the variable value:

```
%JAVA_HOME%\bin;
```

NOTE: Verify that the last entry of the PATH environment variable has no trailing slash (\) otherwise the WebLogic server instance will fail on start-up. Remove the slash if it exists and save your modification.

Set the JAVA_HOME environment (Linux and Solaris)

Set the `JAVA_HOME` variable for Bourne and Bash shells as shown in this example:

Linux:

```
JAVA_HOME=/opt/Java/jdk1.8.0_74
export JAVA_HOME
```

Solaris:

```
JAVA_HOME=/usr/java
export JAVA_HOME
```

NOTE: The specific path varies based on the installation directory you specified and the operating system you are installing on.

NOTE: You must set the `JAVA_HOME_32` environment variable if you are using PDF Generator to convert Open Office files on a WebLogic Server environment.

Set the PATH environment variable (Linux and Solaris)

Set the `PATH` variable for Bourne and Bash shells as shown in this example:

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

Verify JAVA_HOME environment variable

(Optional) Open a command prompt and run the following command:

```
java -version
```

The command returns the Java version installed on your system.

7.4. Creating the WebLogic Server domain

To deploy applications on WebLogic Server, you must have a WebLogic Server domain. A *domain* is the basic administrative unit for WebLogic Server. Here are the two basic types of WebLogic Server domains:

Domain with Managed Servers (recommended): A production environment typically consists of an Administration Server with one or more Managed Servers. The Administration Server is used to perform management operations; the applications and resources are deployed to individual Managed Servers.

Standalone Server Domain: This type of domain can be used for development or test environments in which a single server instance acts as both Administration Server and Managed Server. This option is easier to manage and allows you to use the *hot deploy* feature in WebLogic on a development environment (not recommended).

NOTE: In production environments, Oracle recommends that you deploy applications only on Managed Servers in the domain; the Administration Server should be reserved for management tasks.

Create a WebLogic domain with Managed Server

1) From a command prompt, start the WebLogic Configuration Wizard by navigating to the *[appserver root]*/common/bin directory and typing the appropriate command:
    –   (Windows) `config.cmd`
    –   (Linux and Solaris) `./config.sh`

2) On the Welcome screen, select **Create a new WebLogic domain** and click **Next**.

3) On the Select Domain Source screen, select **Generate a domain configured automatically to support the following products**and click **Next**.

4) On the Create WebLogic Domain screen, either accept the default values or enter the Domain name and Domain location as required, and then click **Next**.

5) In the Configure Administrator Username and Password screen, type your WebLogic user name and password, confirm the password by retyping it, and then click **Next**. Record this user name and password because Configuration Manager will prompt you for them.

6) In the WebLogic Domain Startup Mode panel, select **Production Mode**.

   IMPORTANT: Using Development Mode is not recommended.

7) In the right pane, navigate to the location of the JDK according to your operating system and then click **Next**:

   – (Windows, Linux) Location of **Oracle JDK 1.8** or later. For example, C:\Program Files\Java\jdk1.8.0_XX (Windows) or /opt/Java/jdk1.8.0_XX (Linux)

   – (Solaris) Location of **Oracle JDK 1.8** or later.

8) On the Select Optional Configuration Screen, click **Next**.

9) On the Configuration Summary Screen, click **Create**.

10) On the Creating Domain screen, when the configuration creation is 100% complete, do the following tasks:

   – (Windows) Select **Start Admin Server** and click **Done**.

   – (Linux and Solaris) Click **Done**, and start the server by navigating from a command prompt to the *[appserverdomain]* and typing `./startWebLogic.sh`

11) (Windows, Linux, or Solaris) When prompted, enter the WebLogic user name and password that you entered in step 5.

   NOTE: The Admin Server starts in the command prompt. Enter the user name and password in the command prompt.

Configuring Admin Server Listen Address

1) In the WebLogic Server administration console, under Domain Structure, click **Environment** > **Servers**

2) Under **Change Center**, click **Lock & Edit**.

3) Under **Servers table**, click **AdminServer**.

4) Provide hostname or IP address in the **Listen Address**box.

5) Click **Save** and **Activate Changes**.

6) Restart the Admin Server.

NOTE: Provide specific IP address or host name of Admin Server. Do not leave it the Listen Address box blank. Leading the box blank can lead to managed server startup problems and configuration issues.

7.5. WebLogic Managed Server

You must deploy AEM Forms on JEE applications on Managed Servers in a domain; the Administration Server should be reserved for management tasks. For information about creating a server domain and about Administration and Managed Servers, see the WebLogic product documentation.

NOTE: You must start the WebLogic Administration Server using 64-bit Java. Before starting the Weblogic Administration server using `startWeblogic.sh` script, ensure that `JAVA_OPTIONS=-d64` is set in the environment.

NOTE: The following procedure highlights the changes that are required to the default properties. For properties that are not provided, accept the existing settings. For more information about these screens, see WebLogic Help in the WebLogic Server Administration Console.

Start Node Manager

You must start the Node Manager before you continue the remaining sections in this chapter.

*   From a new command prompt, navigate to the *[appserver root]*\server\bin directory and type the appropriate command:

    TIP: For WebLogic 12.1.3 and 12.2.1, navigate to the *[appserver root]*\bin directory.

    (Windows) `startNodeManager.cmd`

    (UNIX) `./startNodeManager.sh`

    NOTE: On Windows, if you have installed Node Manager as a Windows service, you must start the Node Manager service. If not, use the command line option to start the Node Manager.

Create a new WebLogic Managed Server

1) If WebLogic Administration Server is not already running, from a command prompt, navigate to the *[appserverdomain]* directory and type the appropriate command:

    –   (Windows) `startWebLogic.cmd`

    –   (Linux and Solaris) `./startWebLogic.sh`

2) To access the WebLogic Server administration console, type `http://[host name]:7001/console` in the URL line of a web browser.

3) Type the user name and password that were used to create this WebLogic configuration, and then click **Log In**.

4) Under Change Center, click **Lock & Edit**.

5) Under Domain Structure, click **Environment** >**Servers** and, in the right pane, click **New**.

6) In the Create New Server page, in the **Server Name** box, type a name (such as `server1`) for your Managed Server.

7) In the **Server Listen Address** box, type the computer name or the IP address.

    NOTE: Make note of this value as you must use the same one when you configure the application server and initialize the database when running Configuration Manager, which defaults to *localhost.*

8) In the **Server Listen Port** box, type a port number that is not currently in use, such as `8001`. The Administration Server is already using port 7001.

9) Click **Finish** and then click **Activate Changes**.

10) Under Change Center, click **Lock & Edit**.

11) Under Domain Structure, click **Environment** >**Machines** and, in the right pane, click **New**.

12) In the Create New Machine page, type a name for the machine in the **Name** box, select the operating system you are using from the **Machine OS** list, and then click **OK**.

13) Click **Environment>Servers** and click the name of the Managed Server you created.

14) In the **Machine** list, select the machine you just created.

15) Verify that the listen port is the same as the port number you entered in step 8.

16) Click **Save**.

Memory settings

On Solaris , Configuration Manager does not set the Maximum Heap Size settings for WebLogic, therefore you must configure AEM Forms on JEE manually using the WebLogic Server administration console.

NOTE: Configure this setting if you plan to upgrade to AEM Forms on JEEor later and using Oracle JDK 1.8 On Red Hat Enterprise Linux.

1) On the Configuration tab, click the **Server Start** `tab`.

2) In the **Arguments** box, type the appropriate text to set the memory size for the Managed Server:

   – `-XX:MaxPermSize=1024m`

3) Click **Save** and then click **Activate Changes**.

Complete the memory settings for the Managed Server on WebLogic

1) Move to the following directory: *[WL_HOME]*\user_projects\domains*[appserverdomain]*\bin.

2) Edit the following file in a text editor:

   – (Windows) `setDomainEnv.cmd`

   – (Linux and Solaris) `setDomainEnv.sh`

3) Search for the line `WLS_MEM_ARGS_64BIT=-Xms256m -Xmx512m`and change it to `WLS_MEM_ARGS_64BIT=-Xms256m -Xmx4096m`

4) Search for the line `-XX:MaxPermSize=256m` and change it to `XX:MaxPermSize=1024m`.

   NOTE: You need to set the MaxPermSize parameter only for 64-bit JVM for Solaris.

   NOTE: Configure MaxPermSize if you plan to upgrade to AEM Forms on JEE or later and using Oracle JDK 1.8 with Red Hat Enterprise Linux.

5) Save the changes and close the file.

7.6. Configuring WebLogic for Web Services

For AEM Forms on JEE to accept requests using web services, you must complete the following procedure to modify authentication of the servlet container.

1) Start the WebLogic Administration Server.

2) Open a command prompt and set the environment by using `setWLSEnv`, as shown in this example:

   – (Windows) From the *[WL_HOME]*\wlserver\server\bin\ folder, type `setWLSEnv.cmd`

   – (Linux and Solaris) From the*[WL_HOME]*/wlserver/server/bin/ directory, type `setWLSEnv.sh`

3) Start the WebLogic scripting tool by typing the following command:

   `java weblogic.WLST`

   If you see "`Exception in thread "main" java.lang.NoClassDefFoundError: weblogic/WLST`*", complete the following tasks:*

   – From a command prompt, set the environment by using `<WL_Home>/wlserver/server/bin/setWLSEnv.cmd`

   – From a command prompt, copy the classpath that is returned from the `setWLSEnv.cmd` or `./setWLSEnv.sh` command and append `export CLASSPATH`, as shown in this example:

   `CLASSPATH="C:\PROGRA~1\Java\JDK18~1.0_7\lib\tools.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\lib\weblogic_sp.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\lib\weblogic.jar;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\net.sf.antcontrib_1.1.0.0_1-0b3\lib\ant-contrib.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\modules\features\oracle.wls.common.nodemanager_2.0.0.0.jar;`

   – From a command prompt, copy the path returned from `setWLSEnv.cmd` or `./setWLSEnv.sh` and append `export PATH`, as shown in this example:

   `PATH=";C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\native\win\x64;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\bin;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\org.apache.ant_1.9.2\bin;C:\PROGRA~1\Java\JDK18~1.0_7\jre\bin;C:\PROGRA~1\Java\JDK18~1.0_7\bin;C:\PROGRA~3\Oracle\Java\javapath;C:\PROGRA~1\Java\JDK18~1.0_7\bin;C:\Windows\System32;C:\Windows;C:\Windows\System32\wbem;C:\Windows\System32\WINDOW~1\v1.0\;C:\PROGRA~1\Perforce;C:\PROGRA~2\IBM\db2cmv8\inso;C:\PROGRA~2\IBM\db2cmv8;C:\PROGRA~2\IBM\db2cmv8\bin;C:\PROGRA~2\IBM\db2cmv8\dll;C:\PROGRA~1\Perforce\Server;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\native\win\x64\oci920_8;C:\Oracle\MIDDLE~1\ORACLE~1\oracle_common\modules\org.apache.maven_3.0.5\bin"`

   – Type the following commands in WLST to update servlet container authentication:

   ```
   connect('<WebLogic username>','<WebLogic password>','<WebLogic URL>')
   edit()
   startEdit()
   ```

```
cd('SecurityConfiguration')
cd('<domain name>')
set('EnforceValidBasicAuthCredentials','false')
activate()
exit()
```

NOTE: The WebLogic URL will be in the format `t3://hostname:[port]`, where *[port]* is the administration server port (usually `7001`).

7.7. Stopping and restarting WebLogic

After you make all your configuration changes, restart WebLogic for the changes to take effect. The WebLogic Managed Server, Node Manager and the WebLogic Administration Server also need to be restarted.

You must start the WebLogic Administration Server using 64-bit Java. Before starting the Weblogic Administration server using `startWeblogic.sh` script, ensure that `JAVA_OPTIONS=-d64` is set in the environment.

Stop WebLogic Managed Server

1) In the WebLogic Server administration console, under Domain Structure, click the domain name.

2) Click the **Control** tab and select the check box beside the server you want to stop.

3) Click **Shutdown** and select one of these options:

When work completes: Initiates a graceful shutdown of the selected server, causing the Managed Server to notify its subsystems to complete all in-work requests. A graceful shutdown gives the WebLogic Server subsystems time to complete certain application processing that is currently in progress.

Force Shutdown Now: Initiates a forced shutdown, which causes the Managed Server to instruct subsystems to immediately drop in-work requests.

4) At the WebLogic Server administration console prompt, click **Yes** to confirm the command.

You can verify that the Managed Server has shut down by viewing the table at the bottom of the Control tab. The table displays a list of all the servers and indicates their current state.

Stop WebLogic Administration Server

1) From a command prompt, navigate to *[appserverdomain]\bin*.

2) Type the following command:

   – (Windows) `stopWebLogic.cmd`

   – (Linux, UNIX) `./stopWebLogic.sh`

3) Enter the WebLogic user name and password (if you enabled security when installing WebLogic).

Start WebLogic Administration Server

1) From a command prompt, navigate to *[appserverdomain]*.

2) Type the following command:
   - (Windows) `startWebLogic.cmd`
   - (Linux, UNIX) `./startWebLogic.sh`

3) Enter the WebLogic user name and password (if you enabled security when installing WebLogic).

Start WebLogic Node Manager

- From a command prompt, navigate to the *[appserver root]*/server/bin directory and type the appropriate command:

  TIP: For WebLogic 12.1.3, navigate to the *[appserver root]*\bin directory.

  (Windows) `startNodeManager.cmd`

  (Linux, UNIX). `/startNodeManager.sh`

Start WebLogic Managed Server

1) When the WebLogic Administration Server has started, log in to the WebLogic Server administration console.

2) Under Domain Structure, click **Environment** > **Servers** and, in the right pane, click the managed server.

3) On the next screen, click the **Control** tab and select the check box beside the managed server you want to start.

4) Click **Start** and then click **Yes**.

7.8. Creating JMX policies for database initialization

You must create JMX policies to ensure that the database for core AEM Forms on JEE components initializes correctly.

Complete the following procedures.

Delegating MBean authorization to the realm

Before creating JMX policies, ensure that the security realm is set up to control access to MBeans. For more information, refer to the WebLogic administration console documentation.

1) In the WebLogic administration console, click **Domain Structure > Security Realms.**

2) Click **myrealm** from the Realms list on the **Summary of Security Realms** page.

3) On the Configuration > General page, ensure that **Use Authorization Providers to Protect JMX Access**is selected. If this option is not selected, perform the following steps:

- Click **Lock & Edit** in the Change Center.
- Select **Use Authorization Providers to Protect JMX Access**.
- Click **Save**.
- In the Change Center, click **Activate Changes**.
- Restart the admin server and the managed server.

Create JMX policies

1) In the WebLogic administration console, click**Domain Structure > Security Realms.**
2) On the **Summary of Security Realms** page, click the name of the realm for which you want to modify JMX policies.
3) On the Settings page, click the **Roles and Policies**tab and then click the **Realm Policies** sub tab.
4) In the Name column of the **Policies** table, click **JMX Policy Editor**.
5) On the JMX Policy Editor page, ensure that the **GLOBAL SCOPE** option is selected. Click **Next**.
6) Ensure that the **ALL MBEANS TYPES** option is selected on the next page. Click **Next**.
7) Select the **Attributes: Permission to Write** option and click **Create Policy**.
8) On the Edit JMX Policies page, click **Add Conditions**.
9) Select **Role** from the **Predicate List** drop-down menu and click **Next**.
10) In the **Role Argument Name** box, enter **Anonymous** and click **Add**.
    NOTE: The *Anonymous role is a default WebLogic role for all runtime process users (for example, users required for bootstrapping an application).*
11) Click **Finish**.
12) On the Edit JMX Policies page, click **Save**.
13) Repeat steps 1 to 6.
14) On the **JMX Policy Editor - Attributes and Operations** page, select the **Unregister instances of this MBean using MBean server** option and click **Create Policy**.
15) Repeat steps 8 to 12.

Increase the Stuck Max Thread time

1) In the WebLogic administration console, click**Environment > Servers > Admin Server.**
2) In the right pane, click Admin Server. In the Configuration tab, open the Tuning tab.
3) In the left pane, click Lock and Edit and set value of Stuck Max Thread to 1200.
4) Click Save, and click Activate Changes.
5) Restart the Weblogic Admin server.

7.9. Configuring the AEM Forms on JEE database connectivity

Next, configure the database connectivity and install AEM Forms on JEE by following the instructions provided in Installing and DeployingAEMformsonJEEforWebLogic.

8. Preconfigured Environment Checklist

Before you advance to the Application Server Installation guides, ensure you have recorded the following information on your system configuration:

8.1. Server setup

Preconfiguration	Record data here	Required?
System Requirements met	<enter text here>	Yes
Temp directory		
JDK Install location	<enter text here>	Yes
Global Storage Directory created	<enter text here>	Only if you do not want to use the default locations.

8.2. Application Server Configuration

Preconfiguration	Record data here	Required?
Application Server hostname or IP Address	<enter text here>	Yes
Application Server credentials	<enter text here>	Yes
Application Server port number	<enter text here>	Yes
Preconfiguration completed as described in *ConfiguringaJBossApplicationServer* or *ConfiguringaWebLogicServer* or *ConfiguringaWebSphereApplicationServer*?	<enter text here>	Yes

8.3. Database Configuration

Preconfiguration	Record data here	Required?
Database hostname or IP Address	<enter text here>	Yes
Database account credentials	<enter text here>	Yes
Database port number	<enter text here>	Yes
Preconfiguration completed as described in *Create the AEM Forms Database*	<enter text here>	Yes

8.4. LDAP Server

Preconfiguration	Record data here	Required?
LDAP Server hostname or IP Address	<enter text here>	This is only required if you are using LDAP to authenticate users.
LDAP account credentials	<enter text here>	This is only required if you are using LDAP to authenticate users.

8.5. PDF Generator configuration (Windows)

Preconfiguration	Record data here	Required?
Install Microsoft Office	<enter text here>	Yes. Note that you must use a user account with administrator privileges to install Acrobat and Microsoft Office.

Preconfiguration	Record data here	Required?
Install Acrobat	<enter text here>	Yes. Note that you must use a user account with administrator privileges to install Acrobat and Microsoft Office.
Other native application software installed	<enter text here>	Yes.
Configure the Windows environment variables	<enter text here>	Yes.
32-bit JDK installed		

9. Appendix - Manually Configuring JBoss

This appendix describes the configuration that is required for JBoss EAP that you can download from Red Hat. This option should be considered for advanced installations only. Advanced knowledge of JBoss is typically required.

AEM Forms runs on JBoss Windows Server 2012 (Enterprise or Standard Edition), Red Hat Linux ES/AS 6.5 or 7, SUSE Linux ES 12 platforms, and Solaris 11.

NOTE: If you plan to use the AEM Forms turnkey installation for JBoss, you do not need to review this document. All the instructions that are required for preparing your environment for a turnkey installation are available in Installing and Deploying AEM Forms UsingJBossTurnkey.

9.1. Installing the JDK for JBoss

You must download and install Oracle JDK 8.0 or later updates versions from http://www.oracle.com/technetwork/java/javase/downloads/index.html

Create or set the JAVA_HOME environment variable to point to the location where the JDK is installed.

Set the JAVA_HOME environment variable (Windows)

1) Select **Start** > **Control Panel**>**System**.
2) Click the **Advanced** tab.
3) Click **Environment Variables** and, under System Variables, click **New**.
4) In the **New System Variable** box, type JAVA_HOME as the variable name and enter the directory where you installed the JDK. This directory is the directory that contains the /bin subdirectory. For example, type the following path:

   C:\Program Files\Java\jdk1.8.0_74

Set the PATH environment variable (Windows)

1) Select **Start** > **Control Panel**>**System**.
2) Click the **Advanced** tab and click **Environment Variables**.
3) In the System Variables area, select the PATH variable and then click **Edit**.
4) Append the following text to the beginning of the variable value:

   %JAVA_HOME%\bin;

Set the JAVA_HOME environment (Linux and Solaris)

• It is recommended that you set the JAVA_HOME variable for Bourne and Bash as shown in the following example:

```
JAVA_HOME=/usr/java
export JAVA_HOME
```

Set the PATH environment variable (Linux and Solaris)

- Set the `PATH` variable for Bourne and Bash as shown in the following example:

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

Verify JAVA_HOME environment variable setting (Windows, Linux, or Solaris)

(Optional) Open a command prompt and run the following command:

```
java -version
```

The command returns the Java version installed on your system.

9.2. Manually installing JBoss

You can download and install JBoss EAP from http://www.jboss.org/jbossas/downloads/.

9.3. Starting and stopping JBoss

Several procedures in this appendix require you to stop and start the instance of JBoss where you want to deploy the product.

Start JBoss

1) From a command prompt, navigate to *[appserver root]*/bin.
2) Start the application server by typing the following command:
   - (Windows) domain.bat -c [config_file_name] -b [server_IP_Address]
   - (Linux and Solaris) `./standalone.sh -c [profile_name] -b [server_IP_Address]`

   where `[profile_name]` *is the configuration required for your database and* `[server_IP_Address]` *is the IP address of the server.*

Stop JBoss

1) From a command prompt, navigate to *[appserver root]*/bin.
2) Stop the application server by typing the following command:
   - (Windows) `jboss-cli.bat --connect command=:shutdown`
   - (Linux and Solaris) `./jboss-cli.sh --connect command=:shutdown`

9.4. Modifying the JBoss configuration

The JBoss Application Server is configured using various XML configuration files. JBoss must be shut down before editing any of these configuration files. If JBoss is running and these files are changed, JBoss will probably crash. JBoss also has a few configuration files that are formatted as .property files. You must ensure that the .property files are saved as UNIX text files on Linux or Solaris if you edit these files on Windows environments at any time.

For single-server installations, you may use jboss profile located at *[appserver root]*\standalone\config-uration as a template. For cluster installations, use jboss profile located at *[appserver root]*\domain\ as a template.

It is recommended that you make a copy of the profile (all or standard) and make changes to the copied profile.

Modify the JBoss configuration

Perform the following steps to modify the JBoss configuration to customize JBoss for Adobe Experience Manager Forms.

- Modify EAR file class-loading isolation
- Modify the standalone.conf.bat file (Windows)
- Modify standalone.conf (Linux and Solaris)

Modify the standalone.conf.bat file (Windows only)

1) Open the *[appserver root]*/bin/standalone.conf.bat file in an editor.

2) Delete the text in bold in the following line and add the memory arguments for 64-bit JVM.

    set "JAVA_OPTS=-Xms64M -Xmx512M -XX:MaxPermSize=256M"

    – Memory argument for 64-bit JVM:

        set ""JAVA_OPTS=%JAVA_OPTS% -Xms1024M -Xmx2048M -XX:MaxPermSize=768M"

3) Add the following argument for 64-bit JVM:

        set "JAVA_OPTS=%JAVA_OPTS% -Dadobeidp.serverName=server1
        -Dadobe.cache.multicast-port=33891 -Dfile.encoding=utf8
        -Djava.net.preferIPv4Stack=true"
        set "JAVA_OPTS=%JAVA_OPTS% -DentityExpansionLimit=10000"
        set "JAVA_OPTS=%JAVA_OPTS% -XX:+HeapDumpOnOutOfMemoryError"
        set "JAVA_OPTS=%JAVA_OPTS% -Dorg.jboss.net.protocol.file.useURI=false
        -Dorg.jboss.as.logging.per-deployment=false"
        set "JAVA_OPTS=%JAVA_OPTS% -XX:+UseCompressedOop

4) (Optional) Modify JBoss Application Server to run in IPv6 mode as follows:

    – Locate and modify `-Djava.net.preferIPv4Stack=false`

      –   Insert the string `-Djava.net.preferIPv6Stack=true`

NOTE: If the application server log contains the following error on startup, remove the value for the IPv6 stack and set the IPV4 value back to `true` *:*

"13:37:44,488 WARN [HANamingService] Failed to start AutomaticDiscovery java.net.SocketException: bad argument for IP_MULTICAST_IF: address not bound to any interface at java.net.PlainDatagramSocketImpl.socketSetOption(Native Method)at java.net.PlainDatagramSocketImpl.setOption(PlainDatagramSocketImpl.java:260)"

5)   Save and close the file.

Modify the standalone.bat file (Windows only)

1)   Open the *[appserver root]*/bin/standalone.bat file in an editor.

2)   Modify the line immediately following `rem Setup JBoss specific properties` to set encoding to UTF-8:

```
set JAVA_OPTS=%JAVA_OPTS% -Dprogram.name=%PROGNAME%
```
NOTE: Ensure that no line breaks appear in this new line.

NOTE: These memory arguments are for both, domain controllers and process controllers, and should be set on each node of cluster.

1)   (Optional) Modify JBoss Application Server to run in IPv6 mode as follows:

      –   Locate and modify `-Djava.net.preferIPv4Stack=false`

      –   Insert the string `-Djava.net.preferIPv6Stack=true`

NOTE: If the application server log contains the following error on startup, remove the value for the IPv6 stack and set the IPV4 value back to `true` *:*

"13:37:44,488 WARN [HANamingService] Failed to start AutomaticDiscovery java.net.SocketException: bad argument for IP_MULTICAST_IF: address not bound to any interface at java.net.PlainDatagramSocketImpl.socketSetOption(Native Method)at java.net.PlainDatagramSocketImpl.setOption(PlainDatagramSocketImpl.java:260)"

2)   Save and close the file.

Modify jboss.cli.bat (windows)

1)   Open the [JBoss_root]\bin\jboss-cli.bat file for editing.

2)   Add double-quotes (" ") to the text in bold in the following line:
```
set "JAVA_OPTS=%JAVA_OPTS%
-Djboss.modules.system.pkgs=com.sun.java.swing
-Dlogging.configuration=file:
%JBOSS_HOME%\bin\jboss-cli-logging.properties"
```
3)   Save and close the file.

Modify the standalone.conf file (JBoss with Solaris 11, Red Hat 6.5 or 7, 64-bit only)

Solaris JDKs from Sun require an additional argument to use 64-bit features. Without this configuration change, the Sun JDK defaults to 32-bit support only.

NOTE: If you're running JBoss as a non-root user, use `-Djava.io.tmpdir="location"` to set the location of the temporary directory to a directory to which you have access.

1) Open the *[appserver root]*/bin/standalone.conf file in an editor.

2) Locate the section starting with:

   if ["x$JAVA_OPTS" = "x"]; then

   Modify the section to look like:

```
if [ "x$JAVA_OPTS" = "x" ]; then
    #JAVA_OPTS="-Xms1303m -Xmx1303m -XX:MaxPermSize=256m
-Dorg.jboss.resolver.warning=true
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.lang.ClassLoader.allowArraySyntax=true"
JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx2048m -XX:MaxPermSize=512m
-Dorg.jboss.resolver.warning=true
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsun.lang.ClassLoader.allowArraySyntax=true"
JAVA_OPTS="$JAVA_OPTS -Dadobeidp.serverName=server1 -Dfile.encoding=utf8
-Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -DentityExpansionLimit=10000"
JAVA_OPTS="$JAVA_OPTS -XX:+UseCompressedOops
-XX:+HeapDumpOnOutOfMemoryError"
```

   NOTE: Ensure that this entry appears as a single line in the standalone.conf file.

3) (optional)Modify JBoss Application Server to run in IPv6 mode as follows:

   – Locate and modify `-Djava.net.preferIPv4Stack=false`

   – Add `-Djava.net.preferIPv6Stack=true`

4) Save and close the file.

Modify domain.xml file (Windows and Unix)

1) Open the [JBoss_root]/domain/configuration/domain_<DBType>.xml file for editing.

2) Add the following system properties as a child of the <property name="java.net.preferIPv4Stack" value="true"/> node:

```
<property name="com.arjuna.ats.arjuna.allowMultipleLastResources"
value="true"/>
<property name="org.apache.catalina.connector.URI_ENCODING"
value="UTF-8"/>
```

3) The domain.xml file contains four profiles such as `default`, `ha`, `full`, and `full-ha`. AEM Forms on JEE configured on a Cluster for JBoss application server required only full profile. Search and delete the following nodes:

   – `<profile name="default">`

   – `<profile name="ha">`

   – `<profile name="full-ha">`

4) To add and update logging properties:

   a) In the node`<subsystem xmlns="urn:jboss:domain:logging:1.3">`, set the value of the child node `<append value="true"/>` to `false`.

   b) Add the following code after the `</periodic-rotating-file-handler>` node.

```
<logger category="com.adobe">
        <level name="INFO"/>
</logger>
<logger category="org.springframework">
        <level name="WARN"/>
</logger>
<logger category="org.mc4j.ems">
        <level name="WARN"/>
</logger>
```

   c) Add the following code as the child of the `<subsystem xmlns="urn:jboss:domain:ee:1.1">` node

```
<global-modules>
            <module name="org.jgroups" slot="main"/>
            <module name="org.jacorb" slot="main"/>
</global-modules>
```

5) Locate the following text and add/change the value that appears in bold to 100:

```
<strict-max-pool name="slsb-strict-max-pool" max-pool-size="20"
instance-acquisition-timeout="5"
instance-acquisition-timeout-unit="MINUTES"/>
```

6) locate the `<subsystem xmlns="urn:jboss:domain:jacorb:1.3">`node, replace the child node `<initializers transactions="spec" security="identity"/>` with `<initializers security="identity" transactions="spec"/>`

7) Locate the node

```
<orb socket-binding="jacorb" ssl-socket-binding="jacorb-ssl">
       <initializers security="identity" transactions="spec"/>
</orb>
```

   and add the following lines after it.

```
<properties>
<property name="jacorb.connection.client.pending_reply_timeout"
value="360000"/>
</properties>
```

8) Locate the following text and remove the text that appears in bold:

```
<!--<remoting-connector use-management-endpoint="false"/>-->
```

9) In the `<subsystem xmlns="urn:jboss:domain:messaging:1.4">`node, locate the child node `<address-settings>`. In the child node, after the line `<address-full-policy>PAGE</address-full-policy>`, add the following text:

```
<page-size-bytes>2097152</page-size-bytes>
```

10) In the <subsystem xmlns="urn:jboss:domain:remoting:1.1"> node, remove the value that appears in bold:

```
<connector name="remoting-connector" socket-binding="remoting"
security-realm="ApplicationRealm"/>
```

11) Add the following lines as a child of the <interfaces> node:

```
<interface name="any">
          <any-ipv4-address/> <!-- 0.0.0.0 -->
</interface>
```

12) Locate and delete all the child nodes of all the <socket-binding-group > nodes for all the profiles.

13) Locate and replace the child nodes of the <socket-binding-groups> nodes for all the profiles to

```
 <socket-binding-group name="full-sockets" default-interface="public">
          <!-- Needed for server groups using the 'default' profile  -->
           <socket-binding name="http" interface="any" port="8080"/>
           <socket-binding name="https" interface="any" port="8443"/>
<socket-binding name="jacorb" interface="public" port="3528"/>
           <socket-binding name="remoting" interface="any" port="4447"/>
</socket-binding-group>
```

14) Locate and delete all the child nodes of the <socket-binding-group name="ha-sockets" default-interface="public"> node for all the profiles.

15) Locate and delete all the child nodes of the <socket-binding-group name="full-ha-sockets" default-interface="public"> node for all the profiles.

16) Delete the following childs of the <server-groups> node:

```
<server-group name="other-server-group" profile="full-ha">
          <jvm name="default">
              <heap size="1303m" max-size="1303m"/>
              <permgen max-size="256m"/>
          </jvm>
          <socket-binding-group ref="full-ha-sockets"/>
</server-group>
```

17) Change the heap size from <heap size="1303m" max-size="1303m"/>to <heap size="1024m" max-size="4096m"/>.

18) Change the prem size from <permgen max-size="256m"/>to <permgen max-size="1024m"/>.

19) Save and close the file.

NOTE: The memory arguments specified in the domain_db.xml file are used for server process fork on each node under other-server-group unless some specific configuration parameters is not set in host.xml

Modify the host.xml file

1) Open the [JBoss_root]/domain/configuration/hosts.xml file for editing.

2) Locate the <jvms>node and change the values in bold:

```
<jvms>
    <jvm name="default">
          <heap size="1024m" max-size="2048m"/>
```

```
            <permgen size="256m" max-size="768m"/>
              <jvm-options>
                  <option value="-server"/>
              </jvm-options>
           </jvm> </jvms>
```

3)   Locate the `<server>`node and delete the following child nodes:

```
 <server name="server-two" group="main-server-group" auto-start="true">
        <!-- server-two avoids port conflicts by incrementing the ports
  in
            the default socket-group declared in the server-group -->
        <socket-bindings port-offset="150"/>
      </server>
      <server name="server-three" group="other-server-group"
  auto-start="false">
          <!-- server-three avoids port conflicts by incrementing the
  ports in
            the default socket-group declared in the server-group -->
        <socket-bindings port-offset="250"/>
      </server>
```

4)   Save and close the file.

NOTE: The memory arguments specified in the host.xml have more precedence over the memory arguments specified in the domain_db.xml file.

Modify the modules.xml file

1)   Open the module.xml file from the \Program Files\EAP-6.4.0\jboss-eap-6.4\modules\system\layers\base\sun\jdk\main directory for editing.

2)   Add the following tags to the file in the <paths> tag:

```
<path name="com/sun/org/apache/xerces/internal/jaxp"/>
<path name="sun/net/util"/>
```

3)   Save and Close.

9.5. About JVM Arguments

The memory arguments defined in the domain.conf.bat file are applicable to the processes of domain controller and process controller. Ensure that you set these memory arguments on each node of the cluster.

In a managed domain, the JVM settings are declared in the host.xml and the domain.xml configuration files. Domain controller components responsible for starting and stopping server processes use these settings. In a standalone server instance, the server startup processes can pass command line settings at startup. These settings can be declared from the command line or from the System Properties screen in the Management Console.

Managed Domain

An important feature of the managed domain is the ability to define JVM settings at multiple levels. You can configure custom JVM settings at the host level, by server group, or by server instance. The more specialized child elements override the parent configuration, allowing for the declaration of specific server configurations without requiring exclusions at the group or host level. This also allows the parent configuration to be inherited by the other levels until settings are either declared in the configuration files or passed at runtime.

JVM settings in the domain configuration file

The following example shows a JVM declaration for a server group in the domain_db.xml configuration file.

```
<server-groups>
<server-group name="main-server-group" profile="default">
<jvm name="default">
<heap size="64m " max-size="512m "/>
</jvm >
<socket-binding-group ref="standard-sockets"/>
</server-group>
</server-groups>
```

In this instance a server group called main-server-group declares a heap size of 64 megabytes, and a maximum heap size of 512 megabytes. Any server that belongs to this group inherits these settings. You can change these settings for the group as a whole, by the host, or the individual server.

The following example shows a JVM declaration for a server group in the host.xml configuration file.

```
<servers>
<server name="server-one" group="main-server-group" auto-start="true">
<jvm name="default">
<heap size="64m " max-size="256m "/>
</jvm >
</server>
</servers>
```

In this instance, a server named server-one belongs to the server group named main-servergroup, inheriting the JVM settings from the default JVM group. In the previous example, the main heap size for main-server-group was set at 512 megabytes. By declaring the lower maximum heap size of 256 megabytes, server-one can override the domain.xml settings to fine-tune performance to the required levels.

9.6. Copying jar files

1) Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_-root]\modules\system\layers\base\com\adobe\livecycle\main to the [*JBoss_-root*]\modules\system\layers\base\com\adobe\livecycle\main directory of your downloaded JBoss.

    –   certjFIPS.jar

- cglib.jar

- jsafeFIPS.jar

- jsafeJCEFIPS.jar

- module.xml

NOTE: The module.xml contains path entry for all the jar files.

2) On your server, create a folder structure [JBoss_root]\modules\system\layers\base\com\micro-soft\main.

Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_root]\modules\sys-tem\layers\base\com\microsoft\main to the [JBoss_root]\modules\system\layers\base\com\mic-rosoft\main directory of your downloaded JBoss.

- sqljdbc4.jar

- module.xml

NOTE: The module.xml contains path entry for all the jar files.

3) On your server, create a folder structure [JBoss_-root]\modules\system\layers\base\com\mysql\main..

Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_root]\modules\sys-tem\layers\base\com\mysql\main to the [JBoss_root]\modules\system\lay-ers\base\com\mysql\main directory of your downloaded JBoss.

- mysql-connector-java-bin.jar

- module.xml

NOTE: The module.xml contains path entry for all the jar files.

4) On your server, create a folder structure [JBoss_-root]\modules\system\layers\base\com\oracle\main.

Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_root]\modules\sys-tem\layers\base\com\oracle\main to the [JBoss_root]\modules\system\layers\base\com\ora-cle\main directory of your downloaded JBoss.

- ojdbc6.jar

- module.xml

NOTE: The module.xml contains path entry for all the jar files.

5) On your server, create a folder structure [JBoss_-root]\modules\system\layers\base\org\apache\commons.

Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_root]\modules\sys-tem\layers\base\org\apache\commons\validator\main to the [JBoss_root]\modules\system\lay-ers\base\org\apache\commons directory of your downloaded JBoss.

- commons-validator.jar

- commons-validator.jar.index

- module.xml

NOTE: The module.xml contains path entry for all the jar files.

6)   On your server, create a folder structure [JBoss_root]\modules\system\layers\base\org\hibernate\3.

Copy the following JAR files from [DVD root]\third_party\jboss.zip\[JBoss_root]\modules\system\layers\base\org\hibernate\3 to the [JBoss_root]\modules\system\layers\base\org\hibernate\3 directory of your downloaded JBoss.

–   hibernate-annotations.jar

–   hibernate-entitymanager.jar

–   module.xml

NOTE: The module.xml contains path entry for all the jar files.

9.7. AEM Forms database connectivity for manually installed JBoss

To configure the AEM Forms database connectivity, you must complete the following tasks:

•   Configure the AEM Forms data source.

•   Configure JBoss to use your database as the default data source.

You must install database drivers to the installation directories of the application server. Drivers are required to enable Configuration Manager and the application server to connect to the AEM Forms database. Install the drivers for the type of database that you use for the database.

You must configure the data source to connect to the database. For JBoss, you can configure an MySQL, Oracle, or SQL Server data source.

NOTE: Before proceeding with the following tasks, ensure that JBoss is not running.

Configuring MySQL for manually installed JBoss

To enable JBoss to connect to the MySQL database that stores AEM Forms data, you must complete these tasks.

•   Obtain and copy the MySQL JDBC driver to the instance of JBoss where you will deploy AEM Forms.

•   Create a data source file and deploy it to the instance of JBoss where you will deploy AEM Forms.

•   Encrypt the password in the lc_<db-name>.xml file. Use the following command to encrypt the password:

```
java –cp
%JBOSS_HOME%\modules\system\layers\base\.overlays\layer-base-jboss-ea
p-6.4.5.CP\org\picketbox\main picketbox-4.1.2.Final-redhat-1.jar
org.picketbox.datasource.security.SecureIdentityLoginModule
<password>
```

NOTE: If you do not have the picketbox-4.1.2.Final-redhat-1.jar file, which is used to encrypt datasource passwords for JBoss, then download it from Index of /techpreview/all/org/picketbox/picketbox/4.1.2.Final-redhat-1. Do not use the JAR file available at http://wiki.jboss.org.

Configuring the MySQL data source

Before you configure the MySQL data source, you must have the database created on MySQL. (See Creating a MySQL database.)

Set MySQL as the data source

1) Copy the database profiles from [DVD root]\third_party\jboss.zip\ [JBoss_root]\stand-alone\configuration to the [AppServer_root]\standalone\configuration directory.

2) Open the [JBoss_root]/domain/configuration/domain_oracle.xml file for editing.

3) Locate the <datasources> tag and delete all the child nodes.

```
<datasources>                         <datasource jta="true"
jndi-name="java:/IDP_DS" pool-name="IDP_DS" enabled="true"
use-java-context="true">
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<driver>mysql</driver>
<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>                         <pool>
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>                    </pool>
<security>                    <user-name>adobe</user-name>
<password>password</password>                    </security>
<validation>                          <valid-connection-checker
class-name="org.jboss.jca.adapters.jdbc.extensions.novendor.JDBC4ValidCo
nnectionChecker"/>
<check-valid-connection-sql>SELECT count(*) from
DUAL</check-valid-connection-sql>
<!--exception-sorter
class-name="com.mysql.jdbc.integration.jboss.ExtendedMysqlExceptionSorte
r"/-->                         <exception-sorter
class-name="org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLExceptionS
orter"/>              </validation>                    <timeout>
<blocking-timeout-millis>20000</blocking-timeout-millis>
<idle-timeout-minutes>2</idle-timeout-minutes>
</timeout>                         <statement>
<prepared-statement-cache-size>20</prepared-statement-cache-size>
</statement>              </datasource>
```

4) Replace the bold values with values that are specific to your database:

   – **MySqlDS :** Change to `DefaultDS`.

   – **mysql-hostname**, **3306**, **jbossdb**, **x**, and **y:** The database values that the application server uses to access the database.

5) Save and close the file.

Edit the standardjbosscmp-jdbc.xml file

1) Open the *[appserver root]*\server\<profile_name>\conf\<profile_name>jbosscmp-jdbc.xml file
    and change the following elements (not necessarily contiguous):

    ```
    <!-- optional since 4.0 <datasource-mapping>Hypersonic SQL
    </datasource-mapping> -->
    <fk-constraint>false</fk-constraint>
    ```

    – Remove the text *<!-optional since 4.0*and the trailing -->comment tagfrom the
       <datasource-mapping> tag to uncomment it.

    – Replace `Hypersonic SQL`with `mySQL`in the<datasource-mapping>element.

    The final <datasource-mapping> tag will look like the following line:

    ```
    <datasource-mapping>mySQL</datasource-mapping>
    ```

    – Replace `false`with`true`in the<fk-constraint>element.

2) Save and close the file.

Edit the lc_turnkey.xml file

1) Open the *[appserver root]*\server\<profile_name>\conf\login-config.xml file in a text editor and
    add the following text within the <authentication> element:

    ```
    <security-domain name="EncryptDBPassword">
                        <authentication>
                            <login-module
    code="org.picketbox.datasource.security.SecureIdentityLoginModule"
    flag="required">
                                <module-option name="userName" value="adobe"/>
                                <module-option name="password"
    value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
    value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
                            </login-module>
                        </authentication>
                    </security-domain>
        <security-domain name="EncryptDBPassword_IDP_DS">
                        <authentication>
                            <login-module
    code="org.picketbox.datasource.security.SecureIdentityLoginModule"
    flag="required">
                                <module-option name="userName" value="adobe"/>
                                <module-option name="password"
    value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
    value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                            </login-module>
                        </authentication>
                    </security-domain>
    <security-domain name="EncryptDBPassword_EDC_DS">
                        <authentication>
    ```

```
                                       <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                                           <module-option name="userName" value="adobe"/>
                                               <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                                   <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
                                           </login-module>
                                   </authentication>
                               </security-domain>
```

2)   In the file, specify the values that are specific to your database:

   NOTE: For Oracle RAC, replace **adobe** with the service name specific to your database.

3)   Save and close the file.

4)   Start JBoss.

Configuring Oracle for manually installed JBoss

Install the Oracle 12c database driver

Copy the ojdbc6.jar for JDK 1.8 driver file from the [aem-forms root]/lib/db/oracle directory to the [appserver root]/modules/system/layers/base/com/oracle/main directory. You can also download the Oracle 11g driver from the JDBC Driver Downloads site, see Supported PlatformCombinations for supported versions of Oracle 12c driver.

Edit lc_turnkey.xml file

1)   Copy the database profiles from [DVD root]\third_party\jboss.zip\ [JBoss_root]\stand-alone\configuration to the [AppServer_root]\standalone\configuration directory.

2)   Open the [appserver root]/standalone/configuration/lc_turnkey.xml file in a text editor and locate this line:

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
```

3)   Replace the following values with values that are specific to your database:

   –   *localhost* : The name, IP address, or fully-qualified path of the computer that hosts the database. The default is `localhost`.

   –   *1521* : The port used to access the database. The default port is `1521`.

   –   *adobe* : Change the default value, `adobe`, with your database SID.

   NOTE: If you are using a plauggable database, replace the colon (:) after <port> with a forward slash (/).

4)   *(Only for Oracle RAC)* Replace the connection URL mentioned in step 2 with the following connection URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS=
(PROTOCOL=TCP) (HOST=yourhost1) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP)
(HOST=yourhost2) (PORT=1521)) (LOAD_BALANCE=on) (FAILOVER=on))
(CONNECT_DATA=(SERVER=dedicated) (SERVICE_NAME=service.yourcompany.com
) (FAILOVER_MODE=(TYPE=session) (METHOD=basic) (RETRIES=10) (DELAY=3))))
```

NOTE: Ensure that this entry appears as a single line in the adobe-ds.xml file.

5) *(Only for Oracle RAC)*Replace the following text from the connection URL in step 5 with values that are specific to your database:

   – **yourhost1:** The name, IP address, or fully-qualified domain name of the first node in the cluster that hosts the database.

   – **yourhost2:**The name, IP address, or fully-qualified domain name of the second node in the cluster that hosts the database.

   NOTE: The cluster hosting the database could have *n* nodes. **yourhost1**and **yourhost2** *are examples in the case of a two-node cluster.*

   – **service.yourcompany.com:**The service name for the Oracle RAC database.

   NOTE: If you are using a plauggable database, replace the colon (:) after <port> with a forward slash (/).

6) Modify the minimum and maximum values for the data source connections:

   – IDP_DS:

   ```
   <min-pool-size>1</min-pool-size>
   <max-pool-size>30</max-pool-size>
   ```

   – EDC_DS:

   ```
   <min-pool-size>1</min-pool-size>
   <max-pool-size>30</max-pool-size>
   ```

   NOTE: If your forms server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase`<max-pool-size>`to 50 or more for both IDP_DS and EDC_DS.

7) Save and close the file.

Set Oracle as the data source

To use Oracle pluggable database, see *CreateandconfigureanOraclepluggabledatabase*.

If you are running LiveCycle with Oracle database, you must create three datasources `IDP_DS`, `EDC_DS`, and `DefaultDS`.

1) Copy the database profiles from [DVD root]\third_party\jboss.zip\ [JBoss_root]\stand-alone\configuration to the [AppServer_root]\standalone\configuration directory.

2) Open the [JBoss_root]/domain/configuration/domain_oracle.xml file for editing.

3) Locate the `<datasources>` tag and delete all the child nodes.

4) To create the IDP_DS datasource, in the <datasources> node, add the following lines:

   ```
    <datasource jta="true" jndi-name="java:/IDP_DS" pool-name="IDP_DS"
   enabled="true" use-java-context="true">
   ```

```
<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-
url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                              <pool>
                                  <min-pool-size>1</min-pool-size>
                                  <max-pool-size>30</max-pool-size>
                              </pool>
                              <security>
                                  <user-name>DB_NAME</user-name>
                                  <password>DB_PASSWORD</password>
                              </security>
<validation>
<exception-sorter
class-name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptio
nSorter"/>
</validation>

                              <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                                <idle-timeout-minutes>2</idle-timeout-minutes>
                              </timeout>
                              <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                              </statement>
                          </datasource>
```

5)   To create the EDC_DS datasource, after the IDP_DS datasource nodes, in the <datasources> node,
      add the following lines:

```
 <datasource jndi-name="java:/EDC_DS" pool-name="EDC_DS" enabled="true"
use-java-context="true">

<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-
url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                              <pool>
                                  <min-pool-size>1</min-pool-size>
                                  <max-pool-size>30</max-pool-size>
                              </pool>
```

```
<security>
    <user-name>DB_NAME</user-name>
    <password>DB_PASSWORD</password>
</security>
```
```
<validation>
<exception-sorter
class-name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptio
nSorter"/>
</validation>
```
```
                        <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                        <idle-timeout-minutes>2</idle-timeout-minutes>
                        </timeout>
                        <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                        </statement>
</datasource>
```

6) To create the DefaultDS datasource, after the EDC_DS datasource nodes, in the <datasources> node, add the following lines:

```
<datasource jndi-name="java:/DefaultDS" pool-name="DefaultDS"
enabled="true" use-java-context="true">

<connection-url>jdbc:oracle:thin:@localhost:1521:ORACLE_SID</connection-
url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<driver>oracle</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                        <pool>
                            <min-pool-size>1</min-pool-size>
                            <max-pool-size>30</max-pool-size>
                        </pool>
                        <security>
                            <user-name>DB_NAME</user-name>
                            <password>DB_PASSWORD</password>
                        </security>
<validation>
<exception-sorter
class-name="org.jboss.jca.adapters.jdbc.extensions.oracle.OracleExceptio
nSorter"/>
</validation>
                        <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                        <idle-timeout-minutes>2</idle-timeout-minutes>
```

```
                                </timeout>
                                <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                                </statement>
                        </datasource>
```

7) Set the following database specfic values for datasources IDP_DS, EDC_DS, and DefaultDS:

   a) **Localhost**: The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost

   b) **1521**: If Oracle does not uses the default port, provide the appropriate port number.

   c) **ORACLE_SID**: Replace the ORACLE_SID with your Oracle System Identifier.?

   d) **DB_USER**, **DB_PASSWORD**: Provide the credentials that the application server uses to access your database.

   e) Modify the minimum and maximum values for the data source connections:

   NOTE: If you are using a plauggable database, replace the colon (:) after <port> with a forward slash (/).

   • IDP_DS:
   ```
   <min-pool-size>1</min-pool-size>
   <max-pool-size>30</max-pool-size>
   ```

   • EDC_DS:
   ```
   <min-pool-size>1</min-pool-size>
   <max-pool-size>30</max-pool-size>
   ```

   NOTE: If your forms server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase <max-pool-size> to 50 or more for both IDP_DS and EDC_DS.

   f) Add the following lines as child of the <datasource> tag:
   ```
    <drivers>
   <driver name="oracle" module="com.oracle">
   <xa-datasource-class>oracle.jdbc.xa.client.OracleXADataSource</xa-da
   tasource-class>
   </driver>
   </drivers>
   ```

8) Save and close the file.

Edit the lc_turnkey.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_turnkey.xml file in a text editor and add the following text within the `<authentication>` element:
```
<security-domain name="EncryptDBPassword">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
```

```
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>
                        </login-module>
                    </authentication>
                </security-domain>
        <security-domain name="EncryptDBPassword_IDP_DS">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>
                        </login-module>
                    </authentication>
                </security-domain>
<security-domain name="EncryptDBPassword_EDC_DS">
                    <authentication>
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">
                            <module-option name="userName" value="adobe"/>
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>
                        </login-module>
                    </authentication>
                </security-domain>
```

2) In the file, specify the values that are specific to your database.
3) Save and close the file.
4) Start JBoss.

Configuring SQL Server for manually installed JBoss

Configuring the SQL Server database connectivity

Before you configure the SQL Server data source, you must have the AEM Forms database created on SQL Server. (See Creating a SQL Server database.)

Edit lc_turnkey.xml file

1) Copy the database profiles from [DVD root]\third_party\jboss.zip\ [JBoss_root]\stand-alone\configuration to the [AppServer_root]\standalone\configuration directory.

2) Open the adobe-ds.xml file in a text editor and modify the `<datasource>` element with your SQL Server connection settings:

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
```

3) Replace the following values with values that are specific to your database:

   – *localhost* **:** The name, IP address, or fully-qualified path of the computer that hosts the data-base. The default is `localhost`.

   – *1433* **:** The port used to access the database.

   – *adobe* **:** The name of the database that stores the AEM Forms data. You will need to update the default value, `adobe`, with your database name.

4) Change the `<driver-class>` element as follows:

```
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
```

5) In the lines that follow the `<driver-class>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database. Modify the minimum and maximum values for the data source connections:

   – IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

   – EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

   NOTE: If your forms server handles heavy load, increase the number of maximum JDBC connec-tions to ensure that all jobs are processed. In such cases, increase`<max-pool-size>`to 50 or more for both IDP_DS and EDC_DS.

6) Save and close the file.

Set MSSQL as the data source

If you are running AEM Forms with MSSQL database, you must create three datasources `IDP_DS`, `EDC_DS`, and `DefaultDS`.

1) Copy the database profiles from [DVD root]\third_party\jboss.zip\ [JBoss_root]\stand-alone\configuration to the [AppServer_root]\standalone\configuration directory.

2) Open the [JBoss_root]/domain/configuration/domain_mssql.xml file for editing.

3) Locate the `<datasources>` tag and delete all the child nodes.

4) To create the IDP_DS datasource, in the <datasources> node, add the following lines:

```xml
<datasources>
                    <datasource jta="true" jndi-name="java:/IDP_DS"
pool-name="IDP_DS" enabled="true" use-java-context="true">

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</conn
ection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class
>
                    <driver>sqlserver</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                    <pool>
                        <min-pool-size>1</min-pool-size>
                        <max-pool-size>30</max-pool-size>
                    </pool>
                    <security>
                        <user-name>DB_USER</user-name>
                        <password>DB_PASSWORD</password>
                    </security>
                    <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                      <idle-timeout-minutes>2</idle-timeout-minutes>
                    </timeout>
                    <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                    </statement>
                </datasource>
```

5) To create the EDC_DS datasource, after the IDP_DS datasource nodes, in the <datasources> node, add the following lines:

```xml
<datasource jndi-name="java:/EDC_DS" pool-name="EDC_DS" enabled="true"
use-java-context="true">

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</conn
ection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class
>
                    <driver>sqlserver</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                    <pool>
                        <min-pool-size>1</min-pool-size>
```

```
                              <max-pool-size>30</max-pool-size>
                          </pool>
                          <security>
                              <user-name>DB_USER</user-name>
                              <password>DB_PASSWORD</password>
                          </security>
                          <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                                <idle-timeout-minutes>2</idle-timeout-minutes>
                          </timeout>
                          <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                          </statement>
                      </datasource>
```

6) To create the DefaultDS datasource, after the EDC_DS datasource nodes, in the <datasources> node, add the following lines:

```
<datasource jndi-name="java:/DefaultDS" pool-name="DefaultDS"
enabled="true" use-java-context="true">

<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</conn
ection-url>

<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class
>
                          <driver>sqlserver</driver>


<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation
>
                          <pool>
                              <min-pool-size>1</min-pool-size>
                              <max-pool-size>30</max-pool-size>
                          </pool>
                          <security>
                              <user-name>DB_USER</user-name>
                              <password>DB_PASSWORD</password>
                          </security>
                          <timeout>

<blocking-timeout-millis>20000</blocking-timeout-millis>
                                <idle-timeout-minutes>2</idle-timeout-minutes>
                          </timeout>
                          <statement>

<prepared-statement-cache-size>20</prepared-statement-cache-size>
                          </statement>
                      </datasource>
```

7) Set the following database specfic values for datasources IDP_DS, EDC_DS, and DefaultDS:

a) **Localhost**: The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost

b) **1433**: If MSSQL does not use the default port, provide the appropriate port number.

c) **DB_USER**, **DB_PASSWORD**: Provide the credentials that the application server uses to access your database.

d) Modify the minimum and maximum values for the data source connections:

    *   IDP_DS:

        ```
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
        ```

    *   EDC_DS:

        ```
        <min-pool-size>1</min-pool-size>
        <max-pool-size>30</max-pool-size>
        ```

    NOTE: If your forms server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase <max-pool-size> to 50 or more for both IDP_DS and EDC_DS.

8) Save and close the file.

Edit the lc_turnkey.xml file

1) Open the *[appserver root]*/standalone/configuration/lc_turnkey.xml file in a text editor and add the following lines within the `<authentication>` element:

```
Replace :?
?
<security-domain name="EncryptDBPassword">?
                    <authentication>?
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">?
                            <module-option name="userName" value="adobe"/>?
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>?
                        <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=DefaultDS,service=LocalTxCM"/>?
                        </login-module>?
                    </authentication>?
                </security-domain>?
    <security-domain name="EncryptDBPassword_IDP_DS">?
                    <authentication>?
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">?
                            <module-option name="userName" value="adobe"/>?
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>?
```

```
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=IDP_DS,service=LocalTxCM"/>?
                            </login-module>?
                        </authentication>?
                    </security-domain>?
<security-domain name="EncryptDBPassword_EDC_DS">?
                    <authentication>?
                        <login-module
code="org.picketbox.datasource.security.SecureIdentityLoginModule"
flag="required">?
                            <module-option name="userName" value="adobe"/>?
                            <module-option name="password"
value="-3bfaa32dfe43f65b207a6df87216de44"/>?
                            <module-option name="managedConnectionFactoryName"
value="jboss.jca:name=EDC_DS,service=LocalTxCM"/>?
                        </login-module>?
                    </authentication>?
                </security-domain>?
```

2)   In the file, specify the values that are specific to your database.

3)   Save and close the file.

4)   Start JBoss.

Configure Integrated Security on Windows

1)   Add the sqljdbc_auth.dll file to the Windows systems path (C:\Windows) on the computer that is running JBoss. The sqljdbc_auth.dll file is located within the Microsoft SQL JDBC 4.0 driver installation. The default location is `[SQL_root]/sqljdbc_3.0/enu/auth/x86` for 32-bit operating systems and `[SQL_root]/sqljdbc_3.0/enu/auth/x64` for 64-bit operating systems.

2)   Open the properties for the JBoss for Adobe Experience Manager Forms 6.3 service or the JBoss service that you configured, and click the **Log On** tab.

3)   Select **This Account** and type the value of a valid user account. This change is not required if you are running JBoss from the command line.

4)   Change SQL Server Security from Mixed mode to Windows Authentication only.

9.8. Next steps

Install AEM Forms by following the instructions provided in Installing and Deploying forms server.

10. Appendix - Additional system requirements

PDF Generator, Central Migration Bridge Service, AEM Forms IPv6 support, Connectors for IBM File Net, Documentum, IBM Content Manager, and Forms, Output and ConvertPDF services, and some other components require a few additional settings. Perform these settings only if you are configuring these capabilities.

10.1. Additional requirements for Linux and UNIX based platforms

NOTE: On Linux and UNIX platforms, AEM Forms on JEE installer uses the JDK installed on the machine. Therefore, you must ensure to install the supported JDK version. On other operating systems, the installer uses the JVM bundled with the installer.

Installing and configuring UTF-8

When installing AEM Forms on JEE on Linux and UNIX based operating systems, you must install and configure the US English version of UTF-8 locale if it is not already installed. You will need the install media (CDs or DVDs) for the operating system to perform this task.

NOTE: On Linux platforms, this locale is installed by default and is called en_US.utf8. It can be verified by using the `locale -a` command.

Installing UTF-8 on Solaris

1) Verify the US English UTF-8 locale is not installed by typing `locale -a` in a command prompt. Verify that the command output does not contain the entry EN_US.UTF-8.

2) Insert the Solaris install CD #1 in the disk drive and mount it to an appropriate location, for example:

3) TF-8 oTF-8 o

   `/cdrom/sol_10_807_sparc/s0`

4) Type the following command as root: `localeadm -a nam -d`
   `/cdrom/sol_10_807_sparc/s0`

   NOTE: This command installs all locales in the North America (nam) region even if you specify only the en_US.UTF-8 locale.

5) When the command is complete, type `locale -a` to verify that EN_US.UTF-8 is set as the locale.

   NOTE: For more information, see Installing addtional locales for Solaris

Solaris

NOTE: Ensure that X Window libraries are installed on your operating system. This is required for PDF Generator Forms Standard. See the documentation for your operating system for more information.

IMPORTANT: Do not use the Solaris `tar` command to extract files or errors (such as missing files) will occur. Download the GNU tar tool*and use it to extract all files on a Solaris environment.*

Linux

On Linux operating systems, ensure the following:

- **All Linux distributions:**
  - Ensure that X Window libraries are installed on your operating system. This is required for PDF Generator and Forms. See documentation for your operating system for more information.
  - Install the latest version of 32-bit libcurl, libcrypto, and libssl libraries.
  - Ensure that the directories /usr/lib/X11/fonts and /usr/share/fonts exists. If the directories do not exist, then use the ln command to create a symbolic link from /usr/share/X11/fonts to /usr/lib/X11/fonts and another symbolic link from /usr/share/fonts to /usr/share/X11/fonts. Also ensure that the courier fonts are available at /usr/lib/X11/fonts
  - Ensure that all the fonts (Unicode and non-unicode) are available in the /usr/share/fonts or /usr/share/X11/fonts directory.
  - On RedHat Enterprise Linux 6.x, the courier fonts are not available, download the font-ibm type1-1.0.3.zip archive. Extract the archive at /usr/share/fonts. Create a symbolic link from /usr/share/X11/fonts to /usr/share/fonts. Delete all the .lst font cache files from the Html2PdfSvc/bin and /usr/share/fonts directories.

- **SUSE Linux:** You must install the glibc-locale-32bit library that ships with SUSE Linux Enterprise Server; otherwise, AEM Forms on JEE will not generate PDF files. This library file is not installed by default, you must use YaST to install it. (See the SUSE Linux Enterprise Server documentation for details.)

  If you plan to install AEM Forms on JEE on SUSE Linux 11, you must also install the libstdc++-libc6.2-2.so.3 libraries. SUSE Linux 11 does not include these libraries by default. For more information, see this Novell Web page. These libraries are required for running Adobe Central Pro Output Server

Configuring the file limit values on non-Windows operating systems

To avoid StuckThread issues on a non-Windows operating systems environment, add or increase the rlim values in the /etc/system file.

1) (**Linux**) Locate and open the /etc/security/limits.conf file.

   *(**Solaris**) Locate and open the /etc/system file.*

2) (**Linux**) Add the following lines to the /etc/security/limits.conf file:

```
<app_group> soft nofile 65553
<app_group> hard nofile 65553
```

   Replace <app_group> with the user group who will run the application server. You may also replace < app_group > with an asterisk (*) to match all users and user groups.

   *(**Solaris**) Locate and modify the `rlim` values in the /etc/system file as follows:*

   set `rlim_fd_cur`: The initial (soft) maximum number of file descriptors per process. Set this value to `65553` or more.

set rlim_fd_max: The hard maximum number of file descriptors per process. Set this value to 65553 or more. (This modification is required only if the default value is lower than 65553). You must have super user privileges to change this value.

NOTE: The `rlim_fd_max` value must be equal to or greater than the `rlim_fd_cur` *value.*

3)   Save and close the file.

4)   Restart your computer.

Verify the updated settings

1)   Launch a new shell.

2)   Type `ulimit -n` and press **Enter**.

3)   Verify the value returned matches the `rlim` values you have set.

10.2. LDAP configuration

This configuration is optional and required only if you are using an LDAP directory to authenticate users.

If you do not have an existing LDAP server and database, install and configure your LDAP server and database according to the vendor's documentation. Make note of the LDAP administrator name and password to use during the AEM Forms on JEE configuration process. Configure AEM Forms on JEE to connect with the LDAP database after you install and deploy your services that are part of AEM Forms on JEE. This configuration is done by using the User Manager service.

See the Installing and Deploying AEM forms on JEE document for your application server.

10.3. Additional requirements for PDF Generator

NOTE: You cannot use the Shared Printer Protocol for the `SendToPrinter` API on Windows 2012 machines that have PDF Generator deployed on them. Use alternate protocols like CIFS or Direct IP.

User account for Windows

You must use a user account with administrator privileges for the following tasks:

•   Installing Microsoft Office

•   Installing PDF Generator

•   Installing Acrobat for PDF Generator

•   Running the application server process

NOTE: When you add a users for PDF Generator, grant the user running the application server with the Replace a process level token privilege.

User account for non-Windows operating systems

You must use a user account with administrator privileges for the following tasks:

*   Installing PDF Generator
*   Running the application server process
*   Running the `sudo` command

NOTE: When you add a users for PDF Generator, grant the user running the application server with the Replace a process level token privilege.

Using 64-bit application servers with PDF Generator

Ensure that a 32-bit Java 8 JDK is installed in addition to the 64-bit one the application server uses. Set the environment variable `JAVA_HOME_32`. This variable is required to point to a 32-bit JDK on systems where a 64-bit application server is in use. The specific path varies based on the installation directory you specified and the operating system you are installing on.

NOTE: You need to install the 32-bit Sun JDK and configure `JAVA_HOME_32` to point to the directory where it resides. Review Sun Java 8s Release Notes > Supported System Configurations and download the 32-bit version for your operating system.

IMPORTANT: Ensure that `JAVA_HOME_32` is set only as an environment variable and is not included in the PATH. If `JAVA_HOME_32` *is included in the PATH, Java core dumps may appear during EAR deployment or when you restart the server.*

Set the Windows JAVA_HOME_32 variable

1) Select **Start**> **Control Panel**>**System**.
2) Click the **Advanced System Settings**tab**.**
3) Click **Environment Variables** and, under System Variables, click **New.**
4) Enter the environment variable `JAVA_HOME_32`. This value is the directory that contains the JDK. For example, type the following:

```
C:\Program Files (x86)\Java\jdk1.8.0_74
```

Set the JAVA_HOME_32 variable on non-Windows operating systems

For Linux, set the JAVA_HOME_32 variable for the supported JDK for Bourne and Bash shells as shown in this example:

```
JAVA_HOME_32=/opt/jdk1.8.0_74
export JAVA_HOME_32
```

For Solaris, set the JAVA_HOME_32 variable for the supported JDK for Bourne and Bash shells as shown in this example:

```
JAVA_HOME_32=/opt/jdk1.8
export JAVA_HOME_32
```

Native file conversion software installation

Before you install PDF Generator, install the software that supports the file formats for which PDF conversion support is required and manually activate the licenses for the software using the user account that is used for running the application server process.

Refer to the individual licensing agreement for each native application that your AEM Forms on JEE deployment will support, and ensure that your AEM Forms on JEE deployment meets the licensing requirements specified. Typically, each AEM Forms on JEE user who will use native application support must also have an activated license on their own computer for the native application.

PDF Generator can be extended to convert additional file types to PDF files by using third party native file conversion applications. For the complete list of supported application and file formats, see Supported Platform Combinations document.

NOTE: PDF Generator uses native applications to convert the supported file formats to PDF. Unless otherwise indicated, only the German, French, English, and Japanese versions of these applications and platforms (operating systems) are supported. Also, ensure that the supported languages are installed on underlying platform (operating system).

NOTE: AEM Forms on JEE supports only 32-bit editions of all the above mentioned software.

NOTE: OpenOffice 3.3 or later must be installed on the server to convert the documents created in version 3.3.

NOTE: Native file conversion software might have initial registration/activation dialogs. Dismiss all the initial registration/activation dialogs for all the PDFG user accounts configured on the server.

NOTE: On Linux platform, OpenOffice must be installed under `/root` user. If OpenOffice is installed for specific users, PDFG might not be able to convert OpenOffice documents.

NOTE: End users should not use software applications used by PDF Generator on the server. This can lead to interference with PDF Generator conversions.

You do not need to install a native software application to convert the following native file formats:
*   Print files (PS, PRN, EPS)
*   Web files (HTML)
*   Image files (JPEG, GIF, BMP, TIFF, PNG)

Installing Acrobat for PDF Generator

Install Acrobat DC Pro before running the AEM Forms on JEE installer. Ensure that you launch Acrobat at least once after installing it to avoid PDF Generator configuration issues. Dismiss all modal dialog boxes that appear on launching Acrobat. *NOTE: Ensure that Acrobat is installed using the same user account that you will use to install AEM Forms on JEE.*

However, if AEM Forms on JEE is installed and Acrobat DC Pro is not installed, install Acrobat DC Pro and then run the Acrobat_for_PDFG_Configuration.bat script, located in the folder *[aem-forms root]*\pdfg_-config. Otherwise, PDF conversions will fail.

The Configuration Manager sets the Acrobat_PATH (case-sensitive) environment variable automatically. You can also choose to set it manually, see *Settingenvironmentvariables*. Restart your application server after setting the environment variable.

Configure Acrobat to use SHX fonts (Windows only)

NOTE: Perform these steps to configure Acrobat if you want PDF Generator to use SHX fonts to convert AutoCAD DWG files without installing AutoCAD. Also, these steps need to be performed for all user accounts configured in administration console.

1) Open Acrobat.
2) Select **Edit > Configurations**.
3) Select **Convert to PDF > Autodesk AutoCAD**.
4) Click **Edit Settings**.
5) Click **Configuration Preferences**.
6) Click **Browse** next to the SHX Font File Search Path and specify the path to the SHX font file.
7) Click **OK** on each opened dialog.

QuickTime 7

PDF Generator requires that QuickTime 7.7.9 or later (Player or Pro) be installed if you want to convert video embedded in files, such as PowerPoint presentations to PDF multimedia files. This application is available from the Apple Downloads site.

Setting environment variables

You must set the environment variables in Windows if you plan to create PDF documents from applications such as Photoshop and WordPerfect.

The names of these environment variables are listed here:

* `Notepad_PATH`
* `OpenOffice_PATH`
* `WordPerfect_PATH`
* `Acrobat_PATH`

These environment variables are optional and need to be set only if you plan to use the corresponding application to convert PDF files through PDF Generator. The value of the environment variable should contain the absolute path of the executable that is used to start the corresponding application.

Configuring PDF Generator on a Remote Machine

In case of a cluster, AEM Forms on JEE is installed only on one machine. Perform the following steps to configure PDF Generator on other machines in the cluster:

1) On the remote machine, if an earlier version of Acrobat is installed, uninstall it by using Add or Remove Programs in the Windows Control Panel.

2) Install Acrobat DC Pro by running the installer.

3) From the machine where AEM Forms on JEE is installed, copy `pdfg_config` and `plugins` folders to the remote machine under any directory.

4) On the remote machine, open /pdfg_config/ Acrobat_for_PDFG_Configuration.bat file for editing.

5) Locate and comment the `goto locationerror` line.

   Before

   ```
   goto locationerror
   ```

   After

   ```
   rem goto locationerror
   ```

6) Save and close the Acrobat_for_PDFG_Configuration.bat file.

7) Open the command prompt and run the following command:

   ```
   Acrobat_for_PDFG_Configuration.bat <Path of the pdfg_Configuration folder>
   ```

Service Control Manager command line tool

Before you complete an automatic installation of PDF Generator on Windows, ensure that the Service Control Manager command line tool, sc.exe, is installed in the Windows environment. Some Windows servers do not have this software preinstalled. By default, the sc.exe file is installed in the C:\Windows\system32 directory. Most OS installations have this tool installed. If you do not have the tool installed, it is available in the Windows Resource Kit for your specific version of Windows. To confirm that the tool is installed on your server, type `sc.exe` from a command prompt. The tools usage is returned.

NOTE: For PDF Generator to work properly, ensure that AEM Forms on JEE is running as a Windows service and the service must run under the Local System account.

Headless mode configuration

If you are running PDF Generator in a headless mode environment (that is, on a server without a monitor, keyboard, or mouse), the x11 libraries must be installed. Some flavors of Linux do not install these libraries by default; therefore, you must obtain the libraries and install them manually.

NOTE: *Activating x11 forwarding on a shell session causes the SOAP UI to create UI elements during SOAP requests, leading to request failures. To avoid request failures, you must add the* `-Djava.awt.headless=true` *JVM argument to application server startup parameters. For specific steps, see application server documentation.*

Enabling multi-threaded file conversions and multi-user support for PDF Generator

By default, PDF Generator can convert only one OpenOffice, Microsoft Word, or PowerPoint document at a time. If you enable multi-threaded conversions, PDF Generator can convert more than one of the documents concurrently by launching multiple instances of OpenOffice or PDFMaker (which is used to perform the Word and PowerPoint conversions).

NOTE: Multi-threaded file conversions (through Microsoft Office) are only supported for Microsoft Word 2007, 2010, 2013, or 2016 and PowerPoint 2007, 2010, 2013, or 2016.

NOTE: Microsoft Excel, Publisher, and Project files are not converted simultaneously. During conversion, EXCEL.exe, PUBLISHER.exe, and PROJECT.exe are watched in the task manager.

Each instance of OpenOffice or PDFMaker is launched using a separate user account. Each user account that you add must be a valid user with administrative privileges on the AEM Forms on JEE Server computer. For more information, see ConfiguringWindowsinstallation

After your AEM Forms on JEE Server is configured, add AEM Forms on JEE user accounts in administration console. See the User accounts for multi-threaded file conversions section in the AEM Forms on JEE installation guide for your application server. To enable multiuser support for native files and OpenOffice files on a Windows environment, add a minimum of three users with the following permissions.

When you add users for PDF Generator native conversions, grant the user running the application server with the Replace a process level token privilege.For more information, see *Granting the Replace a process level token privilege (Windows only)*

Dismiss initial dialogs and disable automatic updates for native applications

Converting native files from PDF Generator requires dismissing any initial registration, activation, and Improvement program dialogs with the option to not show them again. Automatic updates for these applications also needs to be disabled as these update dialogs can cause failures on a running server.

The dialogs and automatic update need to be disabled for the user running the server and all user accounts configured under PDFG Accounts for multi-user support. The dialogs need to be dismissed for all third-party applications if installed on the server:

NOTE: Ensure that you launch Adobe Acrobat Distiller at least once for all the PDFG user accounts configured on the server.

Disable error reporting on Windows Server 2012 (Optional but recommended)

While converting a document to PDF using PDF Generator on Windows Server 2012 Windows may report that the executable has encoutered a problem and needs to close. However, it does not impact the PDF conversion as it continues in the background.

To avoid receiving the error, you can disable the Windows error reporting. For more information on disabling error reporting, see https://technet.micro-soft.com/en-us/library/gg232692%28v=ws.10%29.aspx.

Additional configuration required for OpenOffice on non-Windows operating systems

1)   Add entries for additional users (other than the administrator who runs the AEM Forms on JEE Server) in the /etc/sudoers file. For example, if you are running AEM Forms on JEE as a user named

lcadm and a server named myhost, and you want to impersonate user1 and user2, add the following entries to /etc/sudoers:

```
lcadm myhost=(user1) NOPASSWD: ALL
```

```
lcadm myhost=(user2) NOPASSWD: ALL
```

This configuration enables lcadm to run any command on host 'myhost' as 'user1' or 'user2' without prompting for password.

2) Allow all the AEM Forms on JEE users to make connections to the AEM Forms on JEE Server. For example, to allow a local user named user1 the permission of making the connection to the AEM Forms on JEE Server, use the following command:

```
xhost +local:user1@
```

Ensure that the session with which the application server started should not get closed.

For more details, refer to xhost command documentation.

3) Restart the server.

Multi-user support for PDF Generator

To enable multi-user support for native files and OpenOffice files on a Windows environment, a minimum of three users with the following permissions must be added. On a non-Windows operating systems platform, create at least one user.

Platform	User permissions
Windows 2012 Server	Users with administrative privileges, Read/write permission on AEM Forms on JEE temporary directory, PDF Generator temporary directory and application server installation directory.
non-Windows operating systems	Users with sudo privileges Read/write permission on AEM Forms on JEE temporary directory, PDF Generator temporary directory, and application server installation directory.

When you add users for PDF Generator native conversions, you must grant the user running the application server with the *Replace a process level token* privilege. See *Granting the Replace a process level token privilege (Windows only)*.

Granting the Replace a process level token privilege (Windows only)

User account that are used to start the application server should be part of the local administrators group and requires the *Replace a process level token* privilege. To provide *Replace a process level token*privilege:

1) Click Start > Run, and then type `gpedit.msc`.

2) On the Group Policy dialog box, select **Computer Configuration** > **Windows Settings** > **Security Settings** > **Local Policies** > **User Rights Assignment**, and double click **Replace a process level token**.

3) Click **Add User or Group**, add the Windows user account that is used to open the command prompt from which the application server is started.

4) Restart Windows, and then start the application server.

Symbolic link on Linux platform

To substitute required fonts in a HTML-to-PDF conversion on the Linux platform, PDF Generator creates a symbolic link that point to the /usr/share/X11/fonts directory.

Sometimes the user running the application server might not possess permissions that are necessary to create a symbolic link. On such systems; create a symbolic link `/usr/lib/X11/fonts` that point to the `/usr/share/X11/fonts` directory.

Symbolic link on Solaris 11 platform

In Solaris 11 some of the fonts required for HTML to PDF conversion are moved from `/usr/openwin/lib/X11/fonts` location to `/usr/share/fonts` location. To allow PDF Generator to access these fonts, create a symbolic link at `/usr/openwin/lib/X11/fonts`, referring to `/usr/share/fonts` location. To perform HTML-to-PDF conversion on Solaris 11 platform, do the following:

1) Open the Terminal Window

2) Run the following command:

    `ln –s /usr/share/fonts /usr/openwin/lib/X11/fonts/usr_share_fonts`

Additional requirements for Red Hat Enterprise Linux 6 (RHEL6)

PDF Generator requires additional RPM packages and fonts to perform conversions on RHEL6. Perform the following steps to configure the PDF Generator on RHEL6:

1) Install the following RPM packages from RHEL6 installation media:
   – glibc-2.12-1.25.el6.i686.rpm
   – nss-softokn-freebl-3.12.9-3.el6.i686.rpm
   – libX11-1.3-2.el6.i686.rpm
   – libxcb-1.5-1.el6.i686.rpm
   – libXau-1.0.5-1.el6.i686.rpm

- zlib-1.2.3-25.el6.i686.rpm
- libXext-1.1-3.el6.i686.rpm
- fontconfig-2.8.0-3.el6.i686.rpm
- expat-2.0.1-9.1.el6.i686.rpm
- freetype-2.3.11-6.el6_0.2.i686.rpm
- libSM-1.1.0-7.1.el6.i686.rpm
- libICE-1.0.6-1.el6.i686.rpm
- libuuid-2.17.2-12.el6.i686.rpm
- libXrandr-1.3.0-4.el6.i686.rpm
- libXrender-0.9.5-1.el6.i686.rpm
- libXinerama-1.1-1.el6.i686.rpm

2)   In your browser, open website
     `http://cgit.freedesktop.org/xorg/font/ibm-type1/`

3)   Download the compressed file `font-ibm-type1-1.0.3.tar.gz`or
     `font-ibm-type1-1.0.3.zip.`The compressed file contains required fonts.

4)   Extract the downloaded zip file to the `/usr/share/fonts` directory.

Configuring user accounts for multi-threaded file conversions

By default, PDF Generator can convert only one OpenOffice, Microsoft Word, or PowerPoint document at a time. If you enable multi-threaded conversions, PDF Generator can convert more than one of the documents concurrently by launching multiple instances of OpenOffice or PDFMaker (which is used to perform the Word and PowerPoint conversions).

If you need to enable multi-threaded file conversion, you must first perform the tasks outlined in the "Enabling multi-threaded file conversions" section of the Preparing to Install or Upgrade guide available on the AEM Forms on JEE documentation.

For non-Windows operating systems users, you must create users and configure the system to remove the password prompts. The following section outlines the method to create a user and perform additional configurations.

Add user account

1)   In administration console, click **Services** > **PDF Generator**> **User Accounts.**

2)   Click **Add** and enter the user name and password of a user who has administrative privileges on the AEM Forms on JEE Server. If you are configuring users for OpenOffice, dismiss the initial OpenOffice activation dialogs.

     NOTE: If you are configuring users for OpenOffice, the number of instances of OpenOffice cannot be greater than number of user accounts specified in this step.

3)   Restart the AEM Forms on JEE Server.

Manual use of Acrobat restricted

If you installed the PDF Generator for native document conversion, use of the bundled Acrobat installation is restricted to the Generate PDF service and is not licensed for any other use.

10.4. Additional requirements for Connector for Documentum

If AEM Forms on JEE is connecting to Documentum, you must install Document Foundation Classes on machine hosting AEM Forms on JEE.

10.5. Additional requirements for Connector for IBM Content Manager

Connector for IBM® Content Manager requires the following software installed (both available from the IBM website):

•   DB2 Universal Database Client

•   IBM Information Integrator for Content (II4C)

See "Post-Deployment Activities" chapter in the Installing and Deploying AEM Forms on JEE document for your application server.

Configure the connection for a single IBM Content Manager datastore

1)   Start the DB2 Configuration Assistant.

2)   Click **Selected**>**Add Database Using Wizard**.

3)   Select **Manually Configure a Connection to a Database** and click **Next**.

4)   Select **TCP/IP** and click **Next**.

5)   Specify the following TCP/IP communication options and then click **Next**:

    –   In the **Host Name** box, type the host name of the server hosting DB2 Content Manager.

    –   Leave the Service Name box empty.

    –   In the **Port Number** box, type the port number. The default DB2 Content Manager port number is 50000.

6)   In the **Database Name** box, type the IBM Content Manager datastore name and, in the **Database Alias** box, type the alias name for the datastore and then click **Next**.

7)   Click **Next** to accept the default data source settings.

8)   In the **Operating System** list, select the operating system you are using and then click **Next**.

9)   Specify the following system options and then click **Next**:

    –   In the **System Name** box, type the server name hosting DB2. If you click Discover, DB2 Content Manager searches for the system name you specified and, if the system is not found, all of the DB2 instances are listed.

- In the **Host Name** box, type the name of the host, or click View Details to show the domain and IP address of the system you named in the previous step.

- In the **Operating System** list, select the operating system on which you deployed DB2 Content Manager.

10) (Optional) To specify Security options, select **Use Authentication Value in Server's DBM Configuration** and click **Finish**.

11) In the Test Connection dialog box, test the connection as required.

Configure connections for multiple IBM Content Manager datastores

1) Configure the initial connection by following the steps in *ToconfiguretheconnectionforasingleIBM-ContentManagerdatastore:*.

2) Add additional database connections by modifying the cmbicmsrvs.ini file (the file that stores the datastore information) as follows:

- From a command prompt window, change the directory to *[II4C home]*/bin (for example, C:\Program Files\db2cmv8\ on Windows **or** /opt/IBM/db2cmv8 on non-Windows operating systems).

- Run the cmbenv81.bat (Windows) or cmbenv81.sh (non-Windows operating systems) file to set the environment and the classpath for the Java Utilities of II4C.

- Change the directory to *[II4C working directory]*/cmgmt/connectors where *[II4C working directory]* is one of the following paths:

  (Windows) C:/Program Files/db2cmv8

  (Linux) /home/ibmcmadm

  (Solaris) /export/home/ibmcmadm

- Run the command

  ```
  java com.ibm.mm.sdk.util.cmbsrvsicm –a add –s <library server database
  name> -sm <database schema name>
  ```

  where `<library server database name>` *is the same as* `Database Alias` *configured in step 6 above.*

  NOTE: The following procedure allows users without DB2 rights to share the connection credentials through the cmbicmenv.ini file.

Configure a multiuser connection to the IBM Content Manager datastore

1) From a command prompt window, change the directory to *[II4C home]*/bin (for example, C:\Program Files\db2cmv8\ on Windows **or** /opt/IBM/db2cmv8 on non-Windows operating systems).

2) Run the cmbenv81.bat (Windows) or cmbenv81.sh (non-Windows operating systems) file to set the environment and the classpath for the Java Utilities of II4C.

3) Change the directory to *[II4C working directory]*/cmgmt/connectors, where *[II4C working directory]* is one of the following paths:

(Windows) C:/Program Files/db2cmv8

(Linux) /home/ibmcmadm

(Solaris) /export/home/ibmcmadm

4)   Run the command

```
java com.ibm.mm.sdk.util.cmbenvicm -a add -s <library server database
name> -u <database user ID> -p <database password>
```

where `<library server database name>` is the same as `Database alias` configured in step 6 above.

10.6. Additional requirements for Connector for IBM FileNet

These requirements are optional and required only if you are installing Connector for IBM® FileNet.

IBM FileNet 5.0

If AEM Forms on JEE is connecting to IBM FileNet 5.0 Content Engine, you must install the Content Engine Java Client. Use the IBM FileNet 5.0 content engine client installer located by default in C:\Program Files\FileNet\CEClient. During installation, select at least one of the components from Application Engine or Process Engine on the component selection screen.

For IBM FileNet 5.0 Process Engine, you must install the IBM FileNet 5.0 Process Engine Client located by default in C:\Program Files\FileNet\BPMClient. During installation, select the Other option on the component selection screen.

IBM FileNet 5.2

If AEM Forms on JEE is connecting to IBM FileNet 5.2 Content Engine, you must install the Content Engine Java Client. Use the IBM FileNet 5.2 content engine client installer located by default in C:\Program Files\FileNet\CEClient. During installation, select at least one of the components from Application Engine or Process Engine on the component selection screen.

For IBM FileNet 5.2 Process Engine, you must install the IBM FileNet 5.0 Process Engine Client located by default in C:\Program Files\FileNet\BPMClient. During installation, select the Other option on the component selection screen.

10.7. Central Migration Bridge service

The Central Migration Bridge service helps you to migrate existing applications from your Adobe Central Pro Output Server or Adobe Web Output Pak products to work with the Output service. The Central Migration Bridge service allows you to use your current IFD/MDF templates, data transformation scripts, and DAT files in a AEM Forms on JEE environment as part of an overall migration strategy.

NOTE: The Central Migration Bridge is useful only if you have existing Central Pro applications to migrate.

Authorized use of Central Migration Bridge

To use the Central Migration Bridge service, you must have a valid license for Central Pro Output Server 5.7 or an executed Central Pro Output Server 5.7 migration agreement. To install Central Pro Output Server 5.7, use your existing media and existing product authorization code (PAC). Your PAC is for a specific operating system platform; if this differs from the operating system platform on which you are installing AEM Forms on JEE, you will need to acquire a PAC for that operating system. Contact your Adobe representative for more information regarding migration, or acquiring Central Pro Output Server 5.7 media or PAC.

Installation considerations

The Central Migration Bridge service interacts directly with the Central Pro (version 5.7) executable files. You must have Central Pro installed on the same server as AEM Forms on JEE, but it is not a prerequisite to installing AEM Forms on JEE (that is, it can be installed before or after AEM Forms on JEE). Consult the Central Pro documentation set for installation instructions.

IMPORTANT: Do not start Central Pro or change its properties to run automatically.

On Windows, the Central Pro service *Adobe Central Output Server* is installed as a manual service. Do not run the service or change its properties to run automatically.

On non-Windows operating systems, do not start the Central Pro daemon *jfdaemon*. If you edited the start-up script of your computer to launch jfdaemon when restarting the computer, change the script to prevent the daemon from being launched automatically. (See the Central Pro installation documentation.) Do not start Central from the command line by launching the jfserver process.

NOTE: The AEM Forms on JEE User that invokes the Central Migration Bridge service must have access rights to the Central Pro install directory as well as execute permissions on the Central Pro executable files.

AEM Forms on JEE turnkey installations

If you are installing and configuring a AEM Forms on JEE turnkey environment using Express mode, the Central Migration Bridge service will be installed and configured by default - you will not be prompted for any input.

NOTE: Ensure that the Adobe Central Pro product is installed in the default directory.

Forms on JEE custom installations

If you are installing and configuring AEM Forms on JEE using the Custom mode (Partial turnkey or Manual), Configuration Manager will prompt you to include Central Migration Bridge in the deployment.

By default, the service will use the default Central Pro installation path. If Central Pro is installed in a different location, update the [Central Install Dir] configuration for the Central Migration Bridge Service by going to administration console.

When you have finished installing AEM Forms on JEE, if Central Pro is not installed in the default location, do the following steps to point AEM Forms on JEE to the correct directory:

1) Log in to administration console.

2) Click **Services > Applications and Services > Service Management**.

3) Click the **Central Migration Bridge:1.0** service.

4) Type the correct path to the Central Pro install directory.

5) Click **Save**.

NOTE: This configuration can also be made in Workbench. See Creating and managing processes in Workbench documentation.

10.8. AEM Forms on JEE IPv6 support

AEM Forms on JEE includes IPv6 support. The default configurations defined in the installation documentation for AEM Forms on JEE set IPv4 as the default IP protocol because this protocol has the most compatibility with third-party infrastructure.

Do not enable IPv6 unless your deployment must use it. The number of supported platform configurations is reduced when enabling IPv6 support with AEM Forms on JEE. You should verify that all third-party software, hardware, and networks that you plan to use have IPv6 support before you attempt to enable IPv6.

NOTE: If you are enabling CIFS in an IPv6 environment, you must explicitly enable IPv6 configuration after you configure your AEM Forms on JEE installation using Configuration Manager. See "Enabling CIFS in IPv6 mode" in the guide for your application server.

Supported IPv6 configurations

Not all infrastructure components support IPv6. For example, Oracle database does not support IPv6. You can use these databases by configuring the connection between the application server and the databases with IPv4, and the remaining communications over IPv6.

Check with your component vendor if IPv6 is supported.

IPv6 implementation guidelines

When you use IPv6 implementation either partially or fully, keep the following points in mind:

• After installing AEM Forms on JEE, do not use the option to start the Configuration Manager directly from the AEM Forms on JEE installer. Instead, navigate to the *[aem-forms root]*\configurationManager\bin\IPv6 directory, and run the IPv6-specific script (`ConfigurationManager_IPv6.bat` or `ConfigurationManager_IPv6.sh`) to launch the Configuration Manager.

• If you have chosen to validate the application server configuration using the Configuration Manager, the validation will fail after you enable IPv6 for the application server. You can ignore this error message during the process. After you restart the application server in the IPv6 mode, the application server can connect to the database.

- To have a pure IPv6 communication with the database server, modify EDC_DS, AEM_DS, and IDP_DS connection settings to use the hostname of the database which resolves to a numeric IPv6 address.

- Many software components such as database drivers do not completely support numeric IPv6 addresses. So, it is recommended that you use a DNS-resolved hostname instead of numeric IPv6 addresses.

- Ensure that name used for mapping IPv6 is added to the CSRF filter section. If the name is not added, see Preventing CSRF attacks section in administration help.

  NOTE: *Name used for mapping IPv6 must not contain square brackets ([]).*

- In an IPv6 environment, if you are using Microsoft SQL Server, you should specify the database server IP address in the following format. Note that in this string, *;serverName* is a keyword, and so must not be replaced with the actual server name.

  ```
  jdbc:sqlserver://;serverName=<IPv6 address>;
  portNumber=<port>;databaseName=<db_name>
  ```

  Here, instead of the numeric IPv6 address, you can specify the hostname of the SQL Server database.

Configuring IPv6 for JBoss

1) You can download and install JBoss from http://www.jboss.org/jbossas/downloads/ or obtain the jboss.zip file from the third-party directory on the installation media and extract the bundled JBoss.

2) Modify lc_turnkey.xml and the database-specific data source configuration file to connect to the AEM Forms on JEE database.

3) Modify the lc-turnkey.xml file to connect to the AEM Forms on JEE database.

4) Modify the following files to enable IPv6:

   – *(JBoss on Windows)* [appserver root]\bin\standalone.conf.bat

   (JBoss on other platforms) *[appserver root]\bin\standalone.conf*

   - Change `-Djava.net.preferIPv4Stack=true` to `-Djava.net.preferIPv6Stack=true`.

   - Add the `-Djava.net.preferIPv6Addresses=true` argument.

5) Launch Configuration Manager by invoking the *[aem-forms root]*\configurationManager\bin\IPv6\ ConfigurationManager_IPv6.bat or ConfigurationManager_IPv6.sh script.

6) In the Configuration Manager, select the steps to configure EAR files, bootstrap and deploy AEM Forms on JEE modules.

7) After the Configuration Manager process is completed, copy these EAR files to the *[appserver root]*\\standalone\deployments directory.

8) Start JBoss from a command line.

9) Provide Configuration Manager hostname of the computer that is mapped to its IPv6 address and then bootstrap the application server to deploy the AEM Forms on JEE modules.