



**KERNEL AUTHENTICATION &
AUTHORIZATION FOR J2EE (KAAJEE)
VERSION 1.1.0
and
SECURITY SERVICE PROVIDER
INTERFACE (SSPI)
VERSION 1.1.0
FOR WEBLOGIC VERSIONS 9.2 AND
HIGHER**

INSTALLATION GUIDE

March 2011


Department of Veterans Affairs
Office of Information and Technology
Product Development

Revision History

Documentation Revisions

The following table displays the revision history for this document. Revisions to the documentation are based on patches and new versions released to the field.

Table i. Documentation revision history

Date	Description	Author(s)
03/2011	<p>Software and documentation for KAAJEE 1.1.0.007 and KAAJEE Security Service Provider Interface (SSPI) 1.1.0.002, referencing VistALink 1.6 and WebLogic 9.2 and higher.</p> <p>Software Version: 1.1.0.007</p> <p>Security Service Provider Interface (SSPI) Version: 1.1.0.002</p> <p>Kernel Patch: XU*8.0*504</p>	<p>Product Development Services Security Program HWSC development team.</p> <p>Bay Pines, FL OIFO:</p> <ul style="list-style-type: none">• Development Manager—Charles Swartz <p>Oakland, CA OIFO:</p> <ul style="list-style-type: none">• Developer—Jose L. Garcia• Developer—Alan Chan• SQA—Gurbir Singh• Technical Writer—Susan Strack
05/2006	<p>Initial software and documentation for KAAJEE 1.0.0.019 and KAAJEE Security Service Provider Interface (SSPI) 1.0.0.010, referencing VistALink 1.5 and WebLogic 8.1.</p> <p>Software Version: 1.0.0.019</p> <p>Security Service Provider Interface (SSPI) Version: 1.0.0.010</p> <p> NOTE: For a description of the current KAAJEE software version numbering scheme, please review the readme.txt file distributed with the KAAJEE software.</p> <p>In the future, the Development Technology Advisory Committee (DTAC) will be the authoritative source for determining future version numbering schemes for all HealthVet-VistA software file and folder names.</p>	<p>ISS KAAJEE Development Team</p> <ul style="list-style-type: none">• Project Manager—Dan Soraoka• Lead Developer—Alan Chan• Developer—Jose Garcia• SQA—Matt Alderman• Technical Writer—Thom Blom

Patch Revisions

For a complete list of patches related to this software, please refer to the Patch Module on FORUM.

Contents

1	Pre-Installation Instructions.....	1-1
1.1	Purpose	1-1
1.2	Distribution Files	1-1
1.3	Installer/Developer Notes—KAAJEE Software First-Time Installations and Upgrades	1-1
1.4	Application Server Environment Requirements	1-2
2	Installation Overview	2-1
2.1	VistA M Server.....	2-1
2.2	WebLogic V 9.2 and Higher Server Preparation.....	2-1
2.3	KAAJEE SSPI Deployment	2-1
2.4	Configure Managed Server Settings	2-3
2.5	Configure SDS 13.0 (or higher) JDBC Connections with the WebLogic Server.....	2-4
2.6	Deploy a J2EE Web-Based Application With the KAAJEE "Plug-In"	2-4
3	VistA M Server Installation Instructions	3-1
3.1	Confirm/Obtain VistA M Server Distribution Files (<i>recommended</i>)	3-1
3.2	Site Configuration (<i>required</i>)	3-2
3.2.1	Validate User Division Entries	3-2
3.2.2	Validate Institution Associations.....	3-3
3.3	Do Not Run any KAAJEE-based Software During the Installation (<i>recommended</i>)	3-4
3.4	Verify KIDS Install Platform (<i>required</i>)	3-4
3.5	Retrieve and Install the KAAJEE-related VistA M Server Patch (<i>required</i>)	3-4
4	J2EE Application Server Installation Instructions	4-1
4.1	Create KAAJEE Server Domain on WebLogic Application Server (<i>required</i>)	4-2
4.1.1	(Linux: Admin Server) Open a Terminal	4-2
4.1.2	(Linux: Admin Server) Locate the WebLogic Configuration File	4-2
4.1.3	(Linux: Admin Server) Create a New WebLogic Configuration	4-3
4.1.4	(Windows: Admin Server) Start the WebLogic Configuration Wizard	4-4
4.1.5	(Windows: Admin Server) Create a New WebLogic Configuration.....	4-4
4.2	Install and Configure SSPI on the Application Server (<i>required</i>).....	4-11
4.2.1	Undeploy SSPI Software	4-11
4.2.2	Deploy SSPI Software	4-14

4.3	Configure SDS 13.0 (or higher) JDBC Connections with the WebLogic Server (<i>required</i>) ..	4-48
4.4	Ensure the Existence of, or Create, a KAAJEE User with Administrative Privileges (<i>required</i>)	4-48
4.5	Edit the KAAJEE Configuration File (<i>required</i>).....	4-50
4.5.1	Locate the kaajeeConfig.xml File (<i>required</i>)	4-50
4.5.2	Edit the Station Number List in the kaajeeConfig.xml File (<i>required</i>)	4-51
4.5.3	Redeploy and Test the Web Application with the Updated kaajeeConfig.xml File (<i>required</i>).....	4-52
4.6	(Linux/Windows) Configure log4j for All J2EE-based Application Log Entries (<i>required</i>)..	4-52
4.6.1	Configure Application for log4j	4-53
4.6.2	Edit the File Name and Location for All Log Entries.....	4-53
4.6.3	Add KAAJEE-specific Logger Tags	4-54
Appendix A: Installation Back-Out or Roll-Back Procedure		Appendix A-1

Figures

Figure 4-1. Linux Admin Server—Successful domain creation message	4-4
Figure 4-2. WebLogic Configuration Wizard: Select Domain Source	4-5
Figure 4-3. WebLogic Configuration Wizard: Configure Administrator Username and Password	4-6
Figure 4-4. WebLogic Configuration Wizard: Configure Server Start Mode and JDK	4-7
Figure 4-5. WebLogic Configuration Wizard: Customize Environment and Services Settings.....	4-8
Figure 4-6. WebLogic Configuration Wizard: Create WebLogic Domain.....	4-9
Figure 4-7. WebLogic Configuration Wizard: Start Admin Server.....	4-10
Figure 4-8. Linux Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.sh file (<i>Generic example with <Alias> placeholders</i>)	4-17
Figure 4-9. Linux Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.sh file (<i>Alias placeholders resolved with actual path names.</i>).....	4-18
Figure 4-10. Windows Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.cmd file (<i>Generic example with <Alias> placeholders</i>)	4-22
Figure 4-11. WebLogic Server Administration Console: Managed Server Start tab settings	4-26
Figure 4-12. Linux Managed Server—KAAJEE SSPI classpath additions on the Server Start tab (<i>Generic example with <Alias> placeholders</i>).....	4-28
Figure 4-13. Linux Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab (<i>Actual example without <Alias> placeholders</i>)	4-29
Figure 4-14. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (<i>Generic example with <Alias> placeholders</i>)	4-30
Figure 4-15. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (<i>Actual example without <Alias> placeholders</i>)	4-30
Figure 4-16. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (<i>Generic example with <Alias> placeholders</i>).....	4-31
Figure 4-17. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (<i>Actual example without <Alias> placeholders</i>)	4-31
Figure 4-18. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab (<i>Generic example with <Alias> placeholders</i>)	4-33
Figure 4-19. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab (<i>Actual example without <Alias> placeholders</i>).....	4-34
Figure 4-20. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (<i>Generic example with <Alias> placeholders</i>)	4-35
Figure 4-21. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (<i>Actual example without <Alias> placeholders</i>)	4-35
Figure 4-22. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (<i>Generic example with <Alias> placeholders</i>).....	4-36

Figure 4-23. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (<i>Actual example without <Alias> placeholders</i>)	4-36
Figure 4-24. Oracle Database—Sample SSPI SQL script for KAAJEE table definitions	4-37
Figure 4-25. Caché Database—Sample SSPI SQL script for KAAJEE table definitions	4-39
Figure 4-26. Sample KaajeeDatabase.properties file as delivered with KAAJEE	4-39
Figure 4-27. Oracle Database—Sample Driver and URL	4-40
Figure 4-28. Caché Database—Sample Driver and URL	4-40
Figure 4-29. WebLogic 9.2 and higher Server Administration Console—Select New to create new Authentication Provider	4-42
Figure 4-30. WebLogic 9.2 and higher Server Administration Console—Create a New Authentication Provider	4-43
Figure 4-31. WebLogic 9.2 and higher Server Administration Console—Optional Control Flag setting for KaajeeManageableAuthenticator	4-44
Figure 4-32. WebLogic 9.2 and higher Server Administration Console—Select and edit the default Authenticator	4-45
Figure 4-33. WebLogic 9.2 and higher Server Administration Console—Change the Control Flag from REQUIRED to SUFFICIENT	4-46
Figure 4-34. Sample excerpt from a web.xml file—Using the run-as and security-role tags	4-49
Figure 4-35. Sample excerpt from a weblogic.xml file—Using the run-as-role-assignment tag	4-49
Figure 4.5-1. Sample Station Number excerpt of the kaajeeConfig.xml file	4-51
Figure 4-37. Sample excerpt of the mylog4j.xml file—Editing common log file name and location (Windows)	4-53
Figure 4-3. Sample excerpt of the mylog4j.xml file—Adding KAAJEE logger information	4-54

Tables

Table i. Documentation revision history	iii
Table ii. Documentation symbol/term descriptions	ix
Table 1-1. Application server minimum software/network tools/documentation required for KAAJEE ..	1-2
Table 3-1. KAAJEE-related VistA M Server distribution files and environment configuration	3-1
Table 4-1. Application server directory <Alias> placeholders (for documentation purposes)	4-1
Table 4-2. kaajee-1.1.0.xxx—KAAJEE folder structure	4-14
Table 4-3. Oracle Database—KAAJEE SSPI SQL table definitions	4-37
Table 4-4. Caché Database—KAAJEE SSPI SQL table definitions	4-38

Orientation

How to Use this Manual

Throughout this manual, advice and instructions are offered regarding the installation and use of KAAJEE and the functionality it provides for HealthVet-Veterans Health Information Systems and Technology Architecture (VistA) software products.

The installation instructions for KAAJEE are organized and described in this guide as follows:

1. Pre-Installation Instructions.
2. Installation Overview
3. VistA M Server Installation Instructions
4. J2EE Application Server Installation Instructions



Where necessary, separate steps for the following two supported operating systems are provided:

- Linux (i.e., Red Hat Enterprise ES 3.0 or higher)
- Windows


There are no special legal requirements involved in the use of KAAJEE.

This manual uses several methods to highlight different aspects of the material:

- Various symbols/terms are used throughout the documentation to alert the reader to special information. The following table gives a description of each of these symbols/terms:

Table ii. Documentation symbol/term descriptions

Symbol	Description
	NOTE/REF: Used to inform the reader of general information including references to additional reading material.
	CAUTION or DISCLAIMER: Used to inform the reader to take special notice of critical information.
	UPGRADES/FIRST-TIME INSTALLATION: Used to denote Upgrade or First-time installation instructions only.
	Skip forward to the referenced step or procedure that is indicated.
	Instructions that only apply to the Linux operating systems (i.e., Red Hat Enterprise ES 3.0 or higher) are set off and indicated with this Linux "Tux" penguin icon.

Symbol	Description
	Instructions that only apply to Microsoft Windows operating systems (i.e., Microsoft Windows 2000 or XP) are set off and indicated with this stylized "Windows" icon.

- Descriptive text is presented in a proportional font (as represented by this font).
- "Snapshots" of computer online displays (i.e., roll-and-scroll screen captures/dialogues) and computer source code, if any, are shown in a *non*-proportional font and enclosed within a box.
 - User's responses to online prompts and some software code reserved/key words will be bold typeface.
 - Author's comments, if any, are displayed in italics or as "callout" boxes.



NOTE: Callout boxes refer to labels or descriptions usually enclosed within a box, which point to specific areas of a displayed image.

- Java software code, variables, and file/folder names can be written in lower or mixed case.
- All uppercase is reserved for the representation of M code, variable names, or the formal name of options, field and file names, and security keys (e.g., the XUPROGMODE key).

Assumptions About the Reader

This manual is written with the assumption that the reader is familiar with the following:

- VistALink—VistA M Server and Application Server software
- Linux (i.e., Red Hat Enterprise ES 3.0 or higher) or Microsoft Windows environment
- Java Programming language—Java 2 Standard Edition (J2SE) Java Development Kit (JDK, a.k.a. Java Software Development Kit [SDK])
- WebLogic 9.2 and higher—Application server
- Oracle Database 10g—Database (e.g., Security Service Provider Interface [SSPI] or Standard Data Services [SDS] 13.0 (or higher) database/tables)
- Oracle SQL*Plus Software 9.2.0.1.0 (or higher)

This manual provides an overall explanation of the installation procedures and functionality provided by the Kernel Authentication & Authorization for J2EE (KAAJEE) on Weblogic Application Server Versions 9.2 and higher software; however, no attempt is made to explain how the overall HealtheVet-VistA programming system is integrated and maintained. Such methods and procedures are documented elsewhere. We suggest you look at the various VA home pages on the VA Intranet for a general orientation to HealtheVet-VistA at the following address:

<http://vista.med.va.gov/>

Reference Materials

Readers who wish to learn more about KAAJEE should consult the following:

- *Kernel Authentication & Authorization for J2EE (KAAJEE) Installation Guide* (KAAJEE 1.1.0.xxx, & SSPI 1.1.0.xxx), this manual
- *Kernel Authentication & Authorization for J2EE (KAAJEE) Deployment Guide* (KAAJEE 1.1.0.xxx, & SSPI 1.1.0.xxx)
- KAAJEE Web site: <http://vista.med.va.gov/kernel/kaajee/index.asp>
- *Kernel Systems Management Guide*
- *VistALink Installation Guide*
- *VistALink System Management Guide*
- *VistALink Developer Guide*



REF: For more information on VistALink, please refer to the Application Modernization Foundations Web site located at the following Web address:

<http://vaww.vista.med.va.gov/vistalink/>

HealthVet-VistA documentation is made available online in Microsoft Word format and Adobe Acrobat Portable Document Format (PDF). The PDF documents *must* be read using the Adobe Acrobat Reader (i.e., ACROREAD.EXE), which is freely distributed by Adobe Systems Incorporated at the following Web address:

<http://www.adobe.com/>

HealthVet-VistA documentation can be downloaded from the VHA Software Document Library (VDL) Web site:

<http://www.va.gov/vdl/>

HealthVet-VistA documentation and software can also be downloaded from the Enterprise Product Support (EPS) anonymous directories:

- Preferred Method download.vista.med.va.gov

This method transmits the files from the first available FTP server.

- Albany OIFO <ftp://ftp.fo-albany.med.va.gov/>
- Hines OIFO <ftp://ftp.fo-hines.med.va.gov/>
- Salt Lake City OIFO <ftp://ftp.fo-slc.med.va.gov/>



DISCLAIMER: The appearance of any external hyperlink references in this manual does not constitute endorsement by the Department of Veterans Affairs (VA) of this Web site or the information, products, or services contained therein. The VA does not exercise any editorial control over the information you may find at these locations. Such links are provided and are consistent with the stated purpose of this VA Intranet Service.

1 Pre-Installation Instructions

1.1 Purpose

The purpose of this guide is to provide instructions for installing the HealthVet-Veterans Health Information Systems and Technology Architecture (VistA) Kernel Authentication and Authorization for Java (2) Enterprise Edition (KAAJEE) and related software.

KAAJEE is *not* an application but a framework. Users of the software need to understand how it integrates in their working environment. Thus, installing KAAJEE means to understand what jars and files need to be put where and what are the configuration files that you need to have and edit.

KAAJEE provides secure sign-on architecture for HealthVet-VistA Web-based applications.

These HealthVet-VistA Web-based applications are able to authenticating against Kernel on the VistA M Server via an Internet Browser on the client workstation and a middle tier application server (e.g., WebLogic).

1.2 Distribution Files



NOTE: Please refer to "Table 1-1. Application server minimum software/network tools/documentation required for KAAJEE" for confirmation of all KAAJEE and related software and documentation files.



REF: For the KAAJEE software preview/test release, all distribution files are available at the following Web address:

http://vista.med.va.gov/kernel/kaajee/download_9-10.asp

1.3 Installer/Developer Notes—KAAJEE Software First-Time Installations and Upgrades

First-time KAAJEE installers *must* perform *all* installation steps/procedures, except where noted. Those installation steps/procedures that can be skipped during a first-time installation will be displayed as follows:



FIRST-Time INSTALLATION: *First-time installation-specific instructions or information that can be skipped will be found here.*

If you were a test site prior to the final release of KAAJEE, we have notated those installation steps/procedures that have special information based on the final software upgrades that may affect how you install the released version of KAAJEE or provide other pertinent information. The upgrade information will be displayed as follows:



UPGRADES: *Upgrade-specific instructions or information will be found here.*

In addition, we will use this section to also highlight any KAAJEE code changes from previous test/preview versions of the software to the released version of the software that may affect development teams coding KAAJEE-enabled applications.

1.4 Application Server Environment Requirements



NOTE: The information in this topic is directed at the systems management personnel responsible for maintaining the application servers.

The following minimum software tools and files are required to install the KAAJEE software and documentation for application servers running KAAJEE-based Web applications:

Table 1-1. Application server minimum software/network tools/documentation required for KAAJEE

Minimum Software/Configuration/Documentation	Version and Description
Operating System Software	One of the following operating systems: <ul style="list-style-type: none"> Linux (i.e., Red Hat Enterprise ES 3.0 or higher) Microsoft Windows XP or 2000
Application Server Software	WebLogic Versions 9.2 and higher application servers.
SSPI Software	KAAJEE SSPI 1.1.0.xxx REF: Installation and configuration instructions are included in the Chapter 3, "J2EE Application Server Installation Instructions," in this manual.
VistALink Software	Version 1.6
Vista Kernel Software	Patch XU*8*504
KAAJEE_1_1_RELEASENOTES.PDF	Release Notes describes the changes to KAAJEE 1.1 to include new features and enhancements.
KAAJEE_1_1_INSTALLGUIDE.PDF	Installation Guide.
KAAJEE_1_1_DEPLOYGUIDE.PDF	Deployment Guide outlines the details of KAAJEE-related software and gives guidelines on how the

Minimum Software/Configuration/ Documentation	Version and Description
	software is used within HealthVet-Veterans Health Information Systems and Technology Architecture (VistA). It contains the User Manual, Programmer Manual, and Technical Manual information for KAAJEE.
kaajee_security_provider_1.1.0.xxx.zip	Security Provider Interface (SSPI) Software. The KAAJEE SSPI software download Zip file for installation on the application server.
kaajee_security_provider_1.1.0.xxx.zip.MD5	Security Service Provider Interface (SSPI) Software Checksum. The MD5 checksum value for the KAAJEE SSPI software download Zip file.

2 Installation Overview

This section provides an overview of the installation procedures for the Kernel Authentication and Authorization for Java (2) Enterprise Edition (KAAJEE). The chapters that follow address the specific installations that comprise KAAJEE:

- VistA M Server Installation Instructions



NOTE: Instructions for the VistA M Server installation can also be found in the description for Kernel Patch XU*8*504, located in the Patch Module on FORUM.

- J2EE Application Server Installation Instructions

2.1 VistA M Server

Kernel Patch XU*8*504 is the custodial patch for the M server installation of the KAAJEE software. In addition, ensure that the M server system is current with patches for KERNEL, VistALink, and RPC Broker.



NOTE: For information on the minimum software tools and files that are required to install the KAAJEE software, see the section titled "Application Server Environment Requirements" in this documentation.

2.2 WebLogic V 9.2 and Higher Server Preparation

Follow the VistALink 1.6 instructions to deploy your VistALink connector(s).

2.3 KAAJEE SSPI Deployment

1. Unzip the Kaajee Security Provider zip distribution into a KAAJEE SSPI staging folder.
2. Create KAAJEE Schema & SSPI Tables.
 - a. Contact the **DBA to create the KAAJEE user ID, schema, and SSPI tables on the Oracle database.
 - b. Create KAAJEE User ID & Schema.
 - c. To create the SSPI tables, run the OracleTables.sql script, which can be found in the KAAJEE SSPI distribution zip file.
 - d. Validate/Verify the Creation of the KAAJEE Database Schema & Tables. In summary, the DBA will need to perform the following procedures:
 - Identify and create an Oracle Tablespace to hold the KAAJEE schema.
 - Create a user account KAAJEE.
 - Give "connect" and "resource" and "unlimited tablespace" privileges to the user account.

- The user account should have a "default" profile.
 - Set the default tablespace for the KAAJEE user to the one created earlier.
 - Set the default "TEMP" tablespace for the KAAJEE user.
3. Edit the KaajeeDatabase.properties File in the Props Directory.

Make sure the DriverName, db_URL, db_UserID, Password and schema is correct for your KAAJEE database.
 4. Using the following information, edit your admin server startup script to add to the classpath:
 - a. KAAJEE SSPI PROPS folder (i.e.:... /kaajee_security_provider_1.1.0.xxx/props)
 - b. KAAJEE SSPI directory/folder (i.e.:... /kaajee_security_provider_1.1.0.xxx)
 - c. KAAJEE SSPI JAR file (i.e.: w\KaajeeSecurityProviders-1.1.0.xxx.jar)
 - d. KAAJEE SSPI supporting Apache JAR files
 - commons-collections-3.1.jar
 - commons-dbc-1.2.1.jar
 - commons-pool-1.2.jar
 5. Add the following JVM arguments to your admin server startup script (instructions shown for Windows and Linux), where sspidir is the KAAJEE SSPI directory that contains the SSPI JAR file:
 - Windows ==> -Dweblogic.alternateTypesDirectory=%sspindir%
 - Linux ==> -Dweblogic.alternateTypesDirectory=\${sspidir}
 6. Start the admin server.
 7. Log onto admin console.
 8. Navigate to the Authentication Directory:
 - a. Select Security Realms under Domain Structure.
 - b. Navigate to the Providers tab, as shown below:
 - Home > Summary of Security Realms > myrealm > Providers
 - c. Click on New in the Authentication tab
 9. Create a New Authentication Provider:
 - a. From the Providers directory, as shown below:
 - Home > Summary of Security Realms > myrealm > Providersenter KaajeeManageableAuthenticator for the Name.
 - b. Select the same name in the Type pull-down menu.
 - c. Click the 'OK' button.
 10. Ensure Control Flag is set to 'OPTIONAL'.
 - a. When returned to the Authentication page, click on KaajeeManageableAuthenticator.



NOTE: You should now be on the Settings for KaaJeeManageableAuthenticator page.

- b. Check to ensure that the Control Flag is set to the default value of OPTIONAL.

11. Change Control Flag from 'REQUIRED' to 'SUFFICIENT'.

- a. When returned to the Authentication page, select and edit the DefaultAuthenticator Authentication Provider.
- b. Change Control Flag from 'REQUIRED' to 'SUFFICIENT'.
- c. Click 'SAVE'.

12. Bounce your admin server.

13. Verify all Changes Have Taken Place:

- a. Use the WebLogic console software (i.e., WebLogic Server 9.2 Console Login) to navigate to the following locations:
 - Home > Summary of Security Realms > myrealm > Users and Groups (Users tab)
 - Home > Summary of Security Realms > myrealm > Users and Groups (Groups tab)



NOTE: If this is a first-time install, you will not see users populated in the Oracle tables or in the WebLogic console.

2.4 Configure Managed Server Settings

1. Log onto the Admin console.
2. Use the WebLogic Server Console to navigate to the Server Start tab on the Configuration tab to update the Managed Server(s) KAAJEE SSPI-related classpath and arguments. For example, below is a sample navigation path:

Home > Summary of Servers > kjm92L_ManagedSvr1

where 'kjm92L_ManagedSvr1' is the name of the managed server. Replace 'kjm92L_ManagedSvr1' with the name of your managed server.

3. Edit the Class Path field to include the following paths:
 - a. KAAJEE SSPI PROPS folder (i.e.:... /kaajee_security_provider_1.1.0.xxx/props)
 - b. KAAJEE SSPI directory/folder (i.e.:... /kaajee_security_provider_1.1.0.xxx)
 - c. KAAJEE SSPI JAR file (i.e.: wlKaaJeeSecurityProviders-1.1.0.xxx.jar)
 - d. KAAJEE SSPI supporting Apache JAR files
 - commons-collections-3.1.jar
 - commons-dbcp-1.2.1.jar
 - commons-pool-1.2.jar

4. Add the following JVM argument to the Argument field:
 - a. Dweblogic.alternateTypesDirectory=<sspidir>
 - b. Replace <sspidir> with the path to the KAAJEE SSPI directory that contains the SSPI JAR file.
5. Start the managed server.

2.5 Configure SDS 13.0 (or higher) JDBC Connections with the WebLogic Server



NOTE: To configure the SDS tables for a J2EE DataSource, please refer to the "Configuring for a J2EE DataSource" topic in the SDS API Installation Guide.

The SDS API Installation Guide is included in the SDS software distribution ZIP files, which are available for download at the following Web address:

http://vawww.sts.infoshare.va.gov/STS_SDS/Project%20Artifacts/Forms/AllItems.aspx

2.6 Deploy a J2EE Web-Based Application With the KAAJEE "Plug-In"

For details how to deploy a J2EE web-based application with the KAAJEE "plug-in," refer to the Kernel Authentication & Authorization for J2EE (KAAJEE) Deployment Guide for Weblogic Application Server Versions 9.2 and higher.

3 VistA M Server Installation Instructions

The installation instructions in this section are directed at the Information Resource Management (IRM) staff located at a site and are applicable for the Test/Production accounts in the VistA Caché environment.



NOTE: For additional information on the VistA M server installation of the KAAJEE software, see the description for Kernel Patch XU*8*504 located in the Patch Module on FORUM.



NOTE: For information on the minimum software tools and files required to install the KAAJEE software in its entirety (i.e., covering the Java 2 Enterprise Edition [J2EE] and VistA M installations), see the section titled "Application Server Environment Requirements" in this documentation.

3.1 Confirm/Obtain VistA M Server Distribution Files (*recommended*)

The following files and environment configuration are needed to install the Kernel Authentication and Authorization Java (2) Enterprise Edition (KAAJEE)-related VistA M Server software:

Table 3-1. KAAJEE-related VistA M Server distribution files and environment configuration

Minimum Software/Configuration	Description
Operating System Software	InterSystems Caché
Fully Patched M Accounts	You should have both a development Test account and a Production account for KAAJEE software. The account(s) <i>must</i> contain the <i>fully</i> patched versions of the following software: <ul style="list-style-type: none">• Kernel 8.0• Kernel Toolkit 7.3• RPC Broker 1.1• VA FileMan 22.0• VistALink 1.6
VistALink Software	Version 1.6 (also listed above)
VistA Kernel Software	Patch XU*8*504
KAAJEE_1_1_RELEASENOTES.PDF	Release Notes describes the changes to KAAJEE 1.1 to include new features and enhancements.
KAAJEE_1_1_INSTALLGUIDE.PDF	Installation Guide describes in detail the installation procedures for KAAJEE.
KAAJEE_1_1_DEPLOYGUIDE.PDF	Deployment Guide outlines the details of KAAJEE-related software and gives guidelines on how the software is used

Minimum Software/Configuration	Description
	within HealthVet-Veterans Health Information Systems and Technology Architecture (VistA). It contains the User Manual, Programmer Manual, and Technical Manual information for KAAJEE.



REF: For the KAAJEE software release, all distribution files, unless otherwise noted, are available for download from the Enterprise VistA Support (EVS) anonymous directories:

- Albany OIFO ftp://ftp.fo-albany.med.va.gov/
- Hines OIFO ftp://ftp.fo-hines.med.va.gov/
- Salt Lake City OIFO ftp://ftp.fo-slc.med.va.gov/
- Preferred Method download.vista.med.va.gov

This method transmits the files from the first available FTP server.

3.2 Site Configuration (*required*)

The KERNEL SYSTEM PARAMETERS file (#8989.3) holds the site parameters for the installation of Kernel. This allows users to configure and fine tune Kernel for:

- Site-specific requirements and optimization needs.
- HealthVet-VistA software application requirements.

Some parameters are defined by IRM during the Kernel software installation process (e.g., agency information, volume set multiple, default parameters). Other parameters can be edited subsequent to installation (e.g., spooling, response time, and audit parameters). Priorities can also be set for interactive users and for TaskMan. Defaults for fields (e.g., timed read, auto menu, and ask device) are defined for use when not otherwise specified for a user or device. The values in the KERNEL SYSTEM PARAMETERS file (#8989.3) can be edited with the Enter/Edit Kernel Site Parameters option [XUSITEPARM].

3.2.1 Validate User Division Entries

During the authentication process for Web-based applications that are KAAJEE-enabled, KAAJEE displays a list of validated institutions to the user. KAAJEE uses the Standard Data Services (SDS) tables 13.0 (or higher) as the authoritative source to validate the list of station numbers that are stored in the <login-station-numbers> tag in the kaajeeConfig.xml file. After a user selects an institution from this validated list, the software follows the VistA authentication process (i.e., Kernel Signon).



NOTE: The validation of the VistA institution occurs *before* the actual login to the VistA M Server, but *after* the user selects the **Login** button on the KAAJEE Web login page. The selected institution is checked against the SDS 13.0 (or higher) tables for an entry and a VistA Provider. Also, KAAJEE checks that an entry exists in the KAAJEE configuration file.



REF: For more information on the <login-station-numbers> tag and/or the kaajeeConfig.xml file, please refer to the "Edit the KAAJEE Configuration File (*required*)" topic in the "J2EE Application Server Installation Instructions," chapter in this manual.

The VistA authentication process (i.e., Kernel Signon) requires that each user be associated with at least one division/institution. The local DUZ(2) variable on the VistA M Server stores the Internal Entry Number (IEN) of the login institution. Entries in the DIVISION multiple (#16) in the NEW PERSON file (#200) permit users to sign onto the institution(s) stored in this field. If there are *no* entries in the DIVISION multiple (#16) of the NEW PERSON file (#200) for the user signing on, information about the login institution comes from the value in the DEFAULT INSTITUTION field (#217) in the KERNEL SYSTEM PARAMETERS file (#8989.3).

Therefore, sites running any application that is used to sign onto VistA *must* verify that the institution(s) are set up correctly for the application user, as follows:

- **Multi-divisional Sites:** The DIVISION multiple (#16) in the NEW PERSON file (#200) *must* be set up for all users. This assures that the application users have access to only those stations for which they are authorized.
- **Non-multi-divisional Sites:** Sites *must* verify that the value in the DEFAULT INSTITUTION field (#217) in the KERNEL SYSTEM PARAMETERS file (#8989.3) is correct.

3.2.2 Validate Institution Associations

KAAJEE uses the Standard Data Services (SDS) tables 13.0 (or higher) as the authoritative source for institution data. Data in the ASSOCIATIONS Multiple field (#14) in the local site's INSTITUTION file (#4) is uploaded to FORUM, which is then used to populate the SDS tables. Thus, in order to sign onto VistA the data in the ASSOCIATIONS Multiple field (#14) *must* have correct information.

The ASSOCIATIONS Multiple is used to link groups of institutions into associations. The ASSOCIATIONS Multiple consists of the following subfields:

- ASSOCIATIONS (#.01)—This field is a pointer to the INSTITUTIONS ASSOCIATION TYPES file (#4.05).
- PARENT OF ASSOCIATION (#1)—This field points back to the INSTITUTION file (#4) to indicate the parent of the association. This field is cross-referenced to find the children of a parent for an association type.

In the ASSOCIATIONS Multiple, child facilities point to their administrative parent. All clinics point to a division parent, all divisions point to a primary facility parent, primary facilities point to an HCS parent or VISN parent. HCS entries point to a VISN parent. Thus, all parent relationships eventually resolve to a VISN. The first entry (IEN=1) in the ASSOCIATIONS Multiple references the VISN to which the division belongs, so that the PARENT OF ASSOCIATION field in that entry *must* point to a VISN in the INSTITUTION file (#4), and the second entry (IEN=2) references the actual parent of the current institution.

Therefore, sites running any application that is used to sign onto VistA *must* verify that the ASSOCIATION Multiple field (#14) in the INSTITUTION file (#4) has a file entry for their own

institution (and all child divisions if it's a multi-divisional site), and make sure that it is set up correctly. If changes are needed, use the IMF edit option [XUMF IMF ADD EDIT] to update those entries.



REF: For more information on the XUMF IMF ADD EDIT option as well as the ASSOCIATIONS Multiple and PARENT OF ASSOCIATION fields data requirements, please refer to the Institution File Redesign (IFR) supplemental documentation located on the VDL at the following Web address:

<http://www.va.gov/vdl/application.asp?appID=9>

3.3 Do Not Run any KAAJEE-based Software During the Installation (*recommended*)

No HealthVet-VistA Web-based and KAAJEE-enabled software should be running while the KAAJEE installation on the VistA M Server is taking place.

3.4 Verify KIDS Install Platform (*required*)

Verify that the Kernel Installation and Distribution System (KIDS) platform on your system is ready to install VistA M Server patches.

3.5 Retrieve and Install the KAAJEE-related VistA M Server Patch (*required*)

Kernel Patch XU*8*504 is the custodial patch for the M server installation of the KAAJEE software. All VistA M Server patches are distributed in Kernel 8.0 KIDS format. Follow the normal procedures to obtain and install released patches.



Make sure that the Kernel, Kernel Toolkit, RPC Broker, VA FileMan, and VistALink software is fully patched. Patches must be installed in their published sequence.



REF: For more information on these patches, please refer to the Patch Module on FORUM.



Congratulations! You have now completed the installation of KAAJEE-related software on the VistA M Server.

4 J2EE Application Server Installation Instructions

The installation instructions in this section are directed at the system administrators responsible for maintaining the application servers and are applicable for the WebLogic Application Server environment.



NOTE: Unless otherwise noted, all instructions apply to both the Linux (i.e., Red Hat Enterprise ES 13.0 or higher) and Microsoft Windows platforms.



REF: For application server platform requirements, please refer to the "Application Server Environment Requirements" topic in the "Pre-Installation Instructions" chapter in this manual.

Because users can install the KAAJEE software in different root-level directories on the application server, we will use the following directory <Alias> placeholders when discussing KAAJEE file/folder locations:

Table 4-1. Application server directory <Alias> placeholders (for documentation purposes)

Directory <Alias> Placeholder	Description (and Document Default Directories)
<BEA_HOME>	The directory where you installed the WebLogic Servers 9.2 and 10.x software and where all the common programs used by all BEA software are stored. For the examples in this document, the default home directory is: <ul style="list-style-type: none">• Linux: /usr/local/BEA92• Windows: C:\AlsPlace\bea
<JAVA_HOME>	The directory where you installed the Java developer software. For the examples in this document, the default home directory is: <ul style="list-style-type: none">• Linux: /usr/local/BEA92/jdk150_04• Windows: C:\AlsPlace\bea\jdk150_04
<DOMAIN_NAME>	The name of your WebLogic domain. For the examples in this document, the directory is: <ul style="list-style-type: none">• Linux: kjm92Ldomain• Windows: kjm92domain
<USER_DOMAIN_HOME>	The directory where your user domain is located. For the examples in this document, the directory is: <ul style="list-style-type: none">• Linux: /usr/local/BEA92/user_projects/domains/kjm92Ldomain• Windows: c:\ALsPlace\bea\user_projects\domains\kjm92domain

Directory <Alias> Placeholder	Description (and Document Default Directories)
<SSPI_STAGING_FOLDER>	This is the staging directory where your KAAJEE SSPI zip distribution file is located.
<MANAGED_SERVER_NAME>	The name(s) of the Managed Server(s). For the examples in this document, the name is: <ul style="list-style-type: none"> • Linux: kjm92L_ManagedSvr1 • Windows: kjm92_ManagedSvr1
<HEV CONFIGURATION FOLDER>	This is the folder placed on the classpath of WebLogic Application Servers, containing the configuration files for all HealthVet-VistA J2EE applications.

4.1 Create KAAJEE Server Domain on WebLogic Application Server *(required)*



UPGRADES: Skip this step if you have already created a server domain on the WebLogic Application Server (e.g., with the installation of VistALink on the WebLogic Server).



BEGIN: Linux Instructions

To create a WebLogic Server Domain (e.g., kaajeewebdomain) on a Linux Admin Server, do the following:

4.1.1 (Linux: Admin Server) Open a Terminal

Open any X-Windows terminal server software or a secure character-based terminal emulator to access the Linux Admin Server.

Log onto the Linux server where you loaded the WebLogic Application Server.

4.1.2 (Linux: Admin Server) Locate the WebLogic Configuration File

Navigate to the following directory:

<BEA_HOME>/weblogic92/common/bin

4.1.3 (Linux: Admin Server) Create a New WebLogic Configuration

Perform the following steps on the Linux Admin Server to create a new WebLogic configuration:

1. Enter the following command:

./config.sh



NOTE: If you are using a secure character-based terminal emulator, proceed to Step #2 that follows.

If you are using an X-Windows terminal server, follow the instructions as shown in Step #4.1.5, "(Windows: Admin Server) Create a New WebLogic Configuration," that follows.

2. Enter **1** after the "Enter index number to select OR [Exit] [Next]>" prompt to create a new WebLogic configuration.
3. Enter **2** "Basic WebLogic Server Domain 9.2.," after the "Enter index number to select OR [Down] [Exit] [Previous] [Next]>" prompt.
4. Enter **1** after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt to run the wizard in Express Mode.
5. Enter **1**, "Modify 'User name'," after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
6. Enter the **user name** (e.g., weblogic). You can enter any user name of your choosing. If you want to use the default user name (i.e., WebLogic), enter **next** at the prompt.
7. Enter **2**, "Modify 'User password'," after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
8. Enter a **user password** (e.g., weblogic). You can enter any password of your choosing.
9. Enter **3**, "Modify 'Confirm user password'," after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
10. Re-enter the **user password** value you entered in Step #8 (e.g., weblogic) to confirm the user password.
11. Enter **next** after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
12. Enter **1**, "Development Mode," after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
13. Enter **1**, "Sun SDK....," after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt.
14. Enter **next** after the "Enter index number to select OR [Exit] [Previous] [Next]>" prompt to accept the default "Target Location".
15. Enter a domain name (e.g., kaajeewebdomain) after the "Enter value for 'Name' OR [Exit] [Previous] [Next]>" prompt.
16. Enter **next** after the "Enter option number to select OR [Exit] [Previous] [Next]>" prompt.

17. The system indicates that the domain was created successfully, as shown below:

Figure 4-1. Linux Admin Server—Successful domain creation message

```

<----- WebLogic Configuration Wizard ----->

Creating Domain...

0%          25%          50%          75%          100%
[-----|-----|-----|-----]
[ ***** ]

**** Domain Created Successfully! ****

```



END: Linux Instructions

▶▶ Linux users, skip to 4.2.



BEGIN: Microsoft Windows Instructions

To create a WebLogic Server Domain (e.g., kaajeewebdomain) on a Windows Admin Server, do the following:

4.1.4 (Windows: Admin Server) Start the WebLogic Configuration Wizard

On the Microsoft Windows server where the WebLogic Application Server is installed, go to:

Start > Programs > BEA Products > Tools > Configuration Wizard

4.1.5 (Windows: Admin Server) Create a New WebLogic Configuration

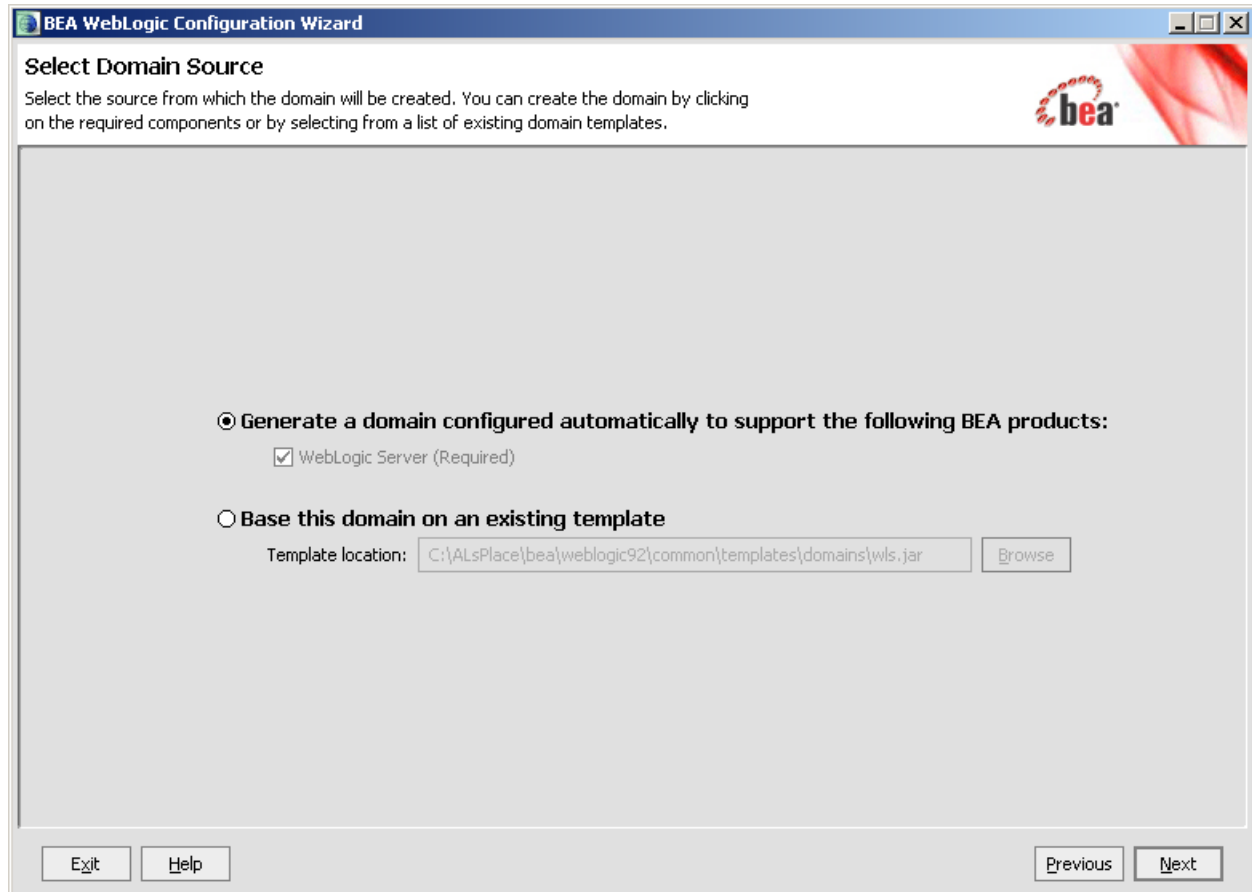


NOTE: Follow the WebLogic Configuration Wizard Prompts.

Perform the following steps on the Windows Admin Server to create a new WebLogic configuration:

1. Choose "Create a new WebLogic configuration" (default) on the first screen and click **Next**.
2. Accept the default "Generate a domain configured automatically to support the following BEA products", as shown below:

Figure 4-2. WebLogic Configuration Wizard: Select Domain Source



Click **Next**.

3. Enter the username and password (also confirm the password), as shown below:

Figure 4-3. WebLogic Configuration Wizard: Configure Administrator Username and Password

BEA WebLogic Configuration Wizard

Configure Administrator Username and Password

Create a user to be assigned to the Administrator role.
This user is the default administrator used to start development mode servers.

Discard Changes

*User name: weblogic

*User password: *****

*Confirm user password: *****|

Description: This user is the default administrator.

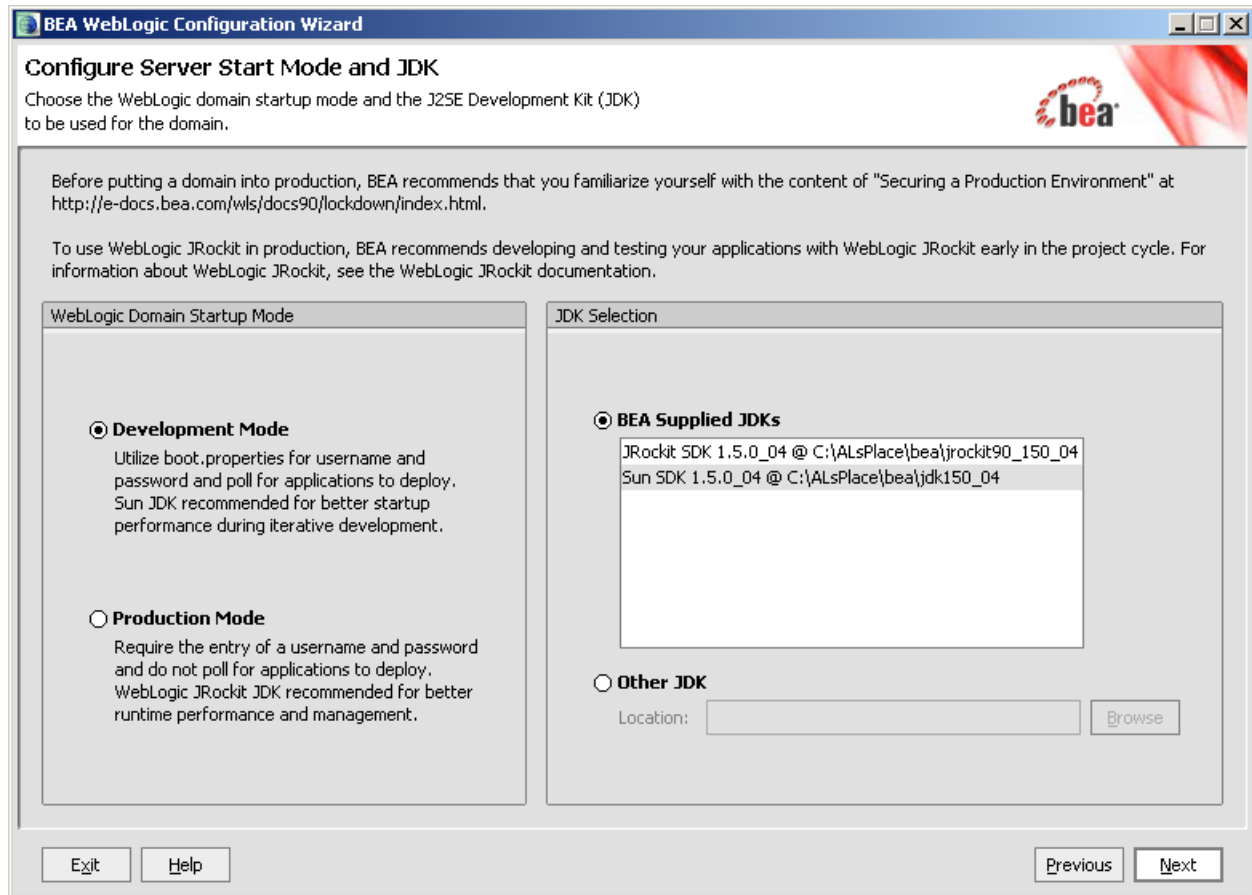
Exit Help Previous Next

In this example, the user entered "**weblogic**" as the username and "**weblogic**" as the password.

Click **Next**.

4. Choose the Java SDK, as shown below:

Figure 4-4. WebLogic Configuration Wizard: Configure Server Start Mode and JDK



In this example, the user chose "Sun SDK 1.5.0_04" Java SDK from the list of BEA Supplied SDK list.

Click **Next**.

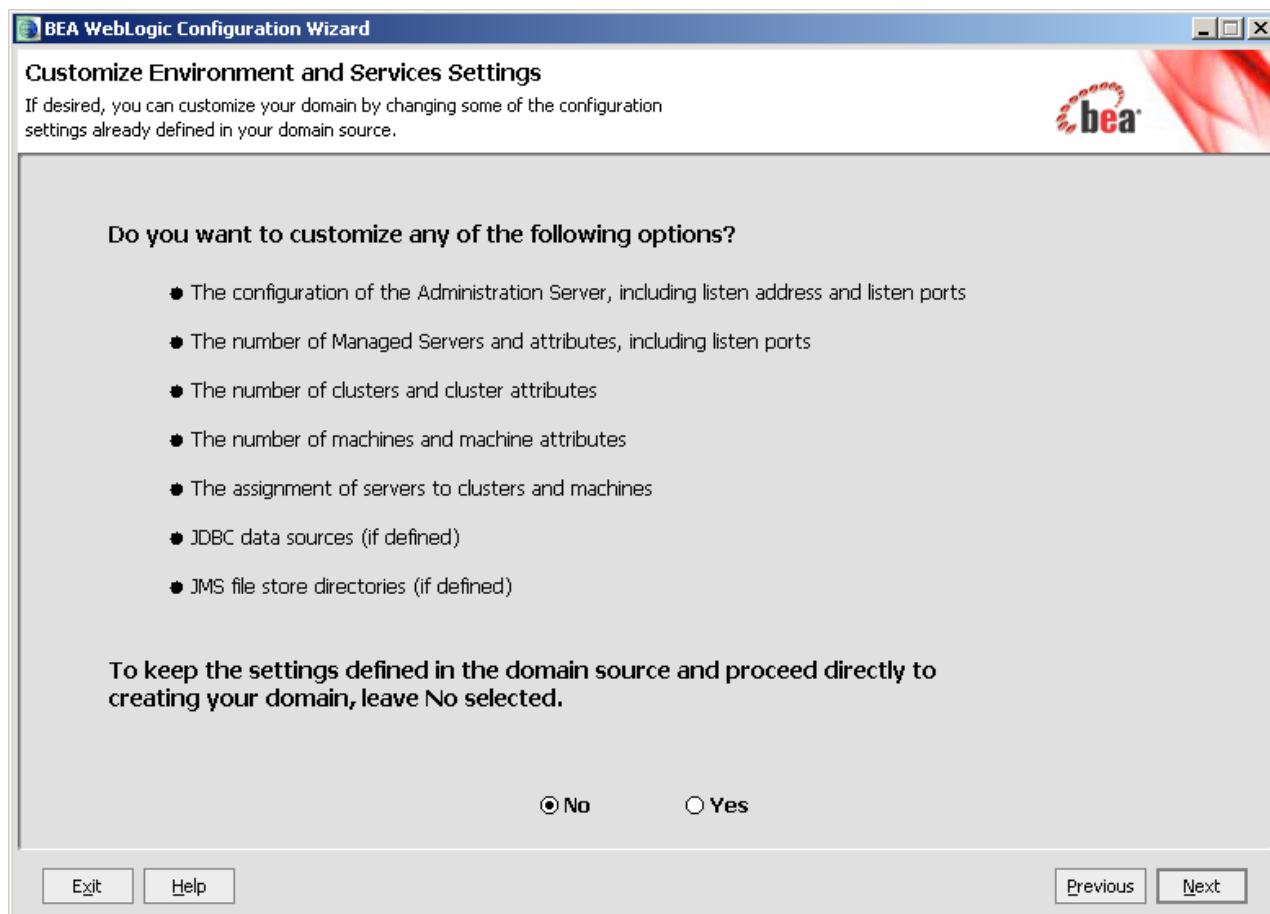


NOTE: The procedures/examples that follow will use Sun Java SDK-specific references.

5. Customize the environment and services settings:

Customize as needed to accommodate your requirements.

Figure 4-5. WebLogic Configuration Wizard: Customize Environment and Services Settings



If you select Yes, you will be prompted with additional dialogue to customize your environment before you create your domain.

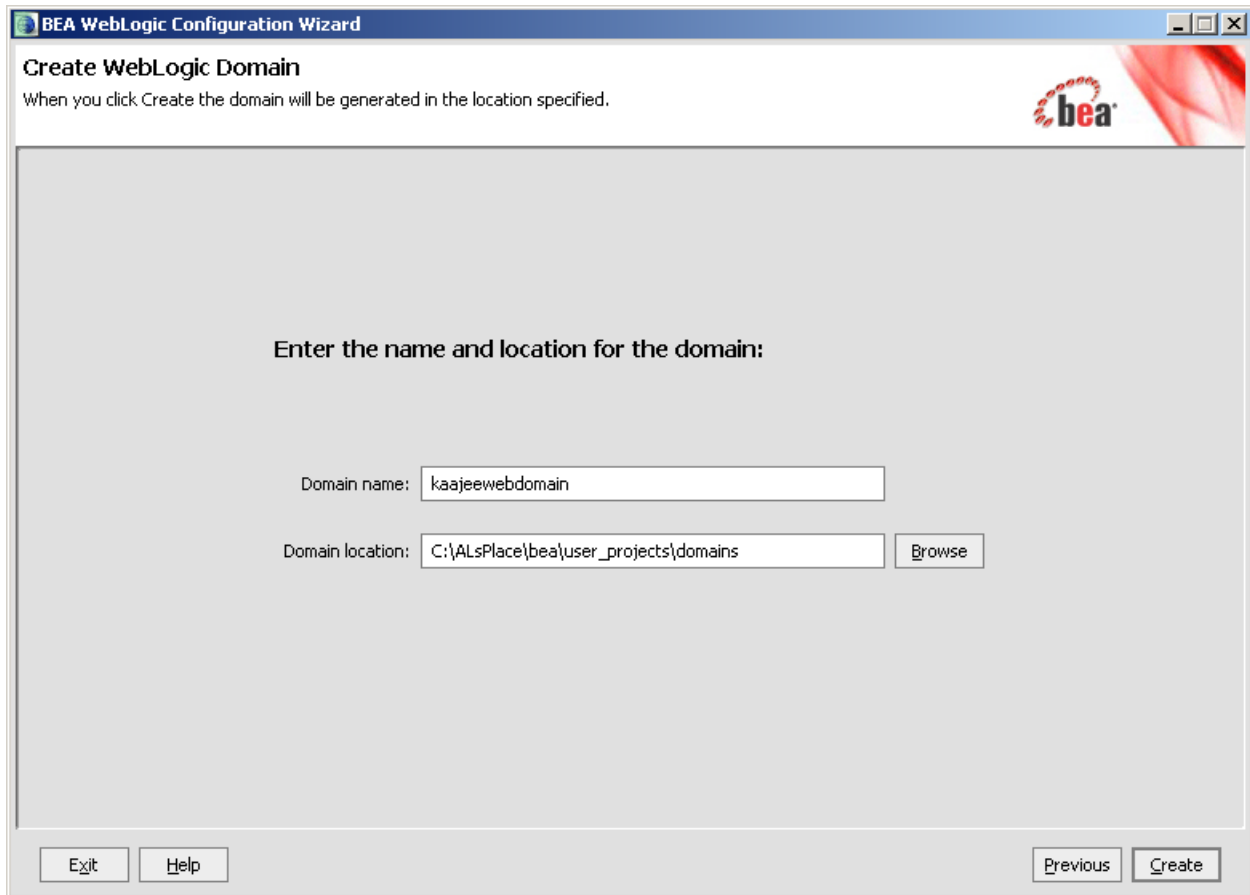


NOTE: Creating the new domain is outlined in the next step in the sequence as follows.

Click **Next**.

6. Create the new domain name, as shown below:

Figure 4-6. WebLogic Configuration Wizard: Create WebLogic Domain



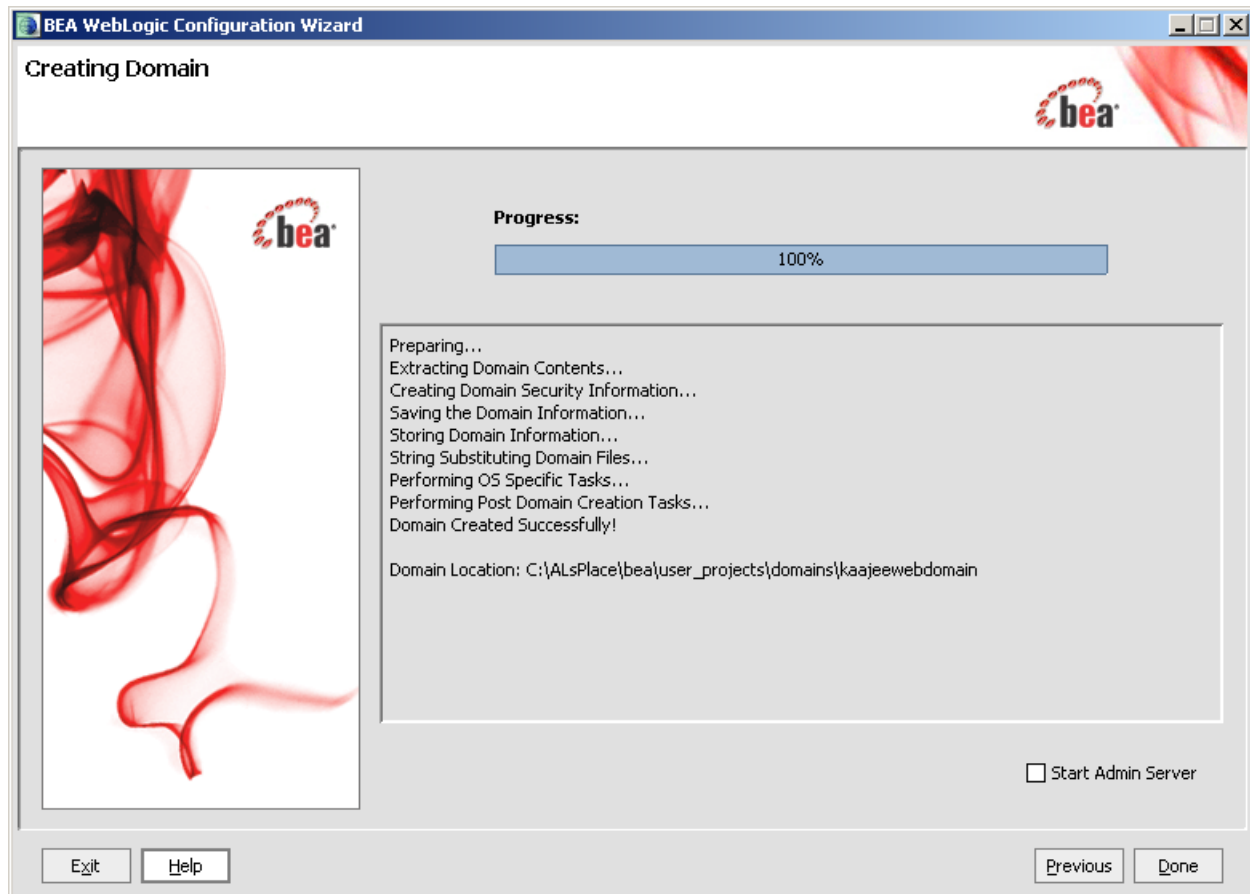
Enter a domain name (e.g., kaajeewebdomain) in the text box after the "Domain Name:" prompt and enter a location after the "Domain location:" prompt.

Click **Create**.

1. Check the "Start Admin Server"

This opens a DOS box on the workstation.

Figure 4-7. WebLogic Configuration Wizard: Start Admin Server



Check the "Start Admin Server" check box.

Click **Done**.

8. Open an Internet Browser and go to the following URL:
<http://localhost:7001/console>.
9. Verify the WebLogic configuration is complete by signing on to the console using the username and password that you entered.

Windows

END: Microsoft Windows Instructions

4.2 Install and Configure SSPI on the Application Server (*required*)

The developer or Application Server system manager *must* install and configure the Security Service Provider Interface (SSPI) software on the WebLogic 9.2 and higher Application Servers in order to develop, test, and run Web-based applications that are KAAJEE-enabled.

4.2.1 Undeploy SSPI Software



FIRST-TIME INSTALLATIONS: Skip this step and proceed to Step #4.2.2, "Deploy SSPI Software," if you have *never* deployed KAAJEE SSPIs on the WebLogic Application Server.



UPGRADES: You *must* perform this step if you have previously deployed KAAJEE SSPIs on the WebLogic Application Server and will be installing a newer version of the KAAJEE SSPIs.

Before installing any new version of the KAAJEE SSPIs on the WebLogic server, users *must* remove any previously installed KAAJEE SSPIs. To do this, perform the following procedures:



NOTE: Before starting, users should shut down all Managed Servers running on the WebLogic Application Server. Shutting down the server ensures that the domain server will refresh its configuration values, etc. upon startup and that the new configuration changes take effect.

4.2.1.1 Delete kaajeeManageableAuthenticator

On the WebLogic Admin Server, use the console to navigate to the following directory:

Home > Summary of Security Realms > myrealm > Providers

Delete kaajeeManageableAuthenticator. Confirm the delete when prompted.

4.2.1.2 Modify DefaultAuthenticator Control Flag

In the same directory, select DefaultAuthenticator. Use the dropdown box next to the Control Flag field to change the setting to **REQUIRED** and then click **Save**.

4.2.1.3 Shut Down the Admin Server on the Application Server

Users should shut down the Admin Server running on the WebLogic Application Server. Shutting down the server ensures that the domain server will refresh its configuration values, etc. upon startup and that the new configuration changes take effect.



BEGIN: Linux Instructions

4.2.1.4 (Linux: Admin Server) Edit the startWebLogic.sh File

On the application server, users need to edit the startWebLogic.sh file. This file is located in the following directory:

`<BEA_Home>/user_project/domains/<DOMAIN_NAME>/bin/`

For example:

`/u01/app/bean/user_project/domains/kaajeewebdomain/bin/`

In the startWebLogic.sh file, delete the following argument:

`-Dweblogic.alternateTypesDirectory=${sspidir}`

Save and close the file.



END: Linux Instructions

▶▶ Linux users, skip to 4.2.1.6.



BEGIN: Microsoft Windows Instructions

4.2.1.5 (Windows: Admin Server) Edit the startWebLogic.cmd File

On the application server, users need to edit the startWebLogic.cmd file. This file is located in the following directory:

`<BEA_Home>\user_project\domains\<DOMAIN_NAME>\bin\`

For example:

`C:\bean\user_project\domains\kaajeewebdomain\bin\`

In the startWebLogic.cmd file, delete the following argument:

`-Dweblogic.alternateTypesDirectory=%sspidir%`

Save and close the file.



END: Microsoft Windows Instructions

4.2.1.6 Start the Admin Server on the Application Server

Users should start the Admin Server on the WebLogic Application Server and then log into the WebLogic Console.

4.2.1.7 Verify Removal of the `kaajeeManageableAuthenticator`

Users should navigate to the following directory:

Home > Summary of Security Realms > myrealm > Providers

Verify that the `kaajeeManageableAuthenticator` is no longer listed.

4.2.1.8 Shut Down the Admin Server on the Application Server

Users should shut down the Admin Server running on the WebLogic Application Server. Shutting down the server ensures that the domain server will refresh its configuration values, etc. upon startup and that the new configuration changes take effect.

4.2.1.9 Move and Back Up the `wlKaajeeSecurityProviders-1.1.0.xxx.jar` File

On the application server, users should navigate to the `<SSPI_STAGING_FOLDER>` staging directory. To complete the cleanup and create a backup, locate and move the `wlKaajeeSecurityProviders-1.1.0.xxx.jar` file to a backup directory.

4.2.1.10 KAAJEE SSPI Successfully Undeployed

At this point, users are now ready to deploy the latest version of the KAAJEE SSPIs, proceed to Step #4.2.2, "Deploy SSPI Software."

4.2.2 Deploy SSPI Software

To install the KAAJEE SSPIs on the WebLogic server, perform the following procedures:

4.2.2.1 Download/Obtain SSPI Software

Download the `kaajee_security_provider_1.1.0.xxx.zip` software from the EVS anonymous directories.

4.2.2.2 Create SSPI Staging Area on the Application Server



UPGRADES: Skip this step if you have already created an SSPI staging area on the WebLogic Application Server.

Create a KAAJEE SSPI staging directory under the WebLogic Application Server:

<SSPI_STAGING_FOLDER>

4.2.2.3 Load/Install the SSPI Software on the Application Server

Extract all files/folders contained inside the **<SSPI_STAGING_FOLDER>** staging directory.

After unzipping/exploding the `kaajee_security_provider_1.1.0.xxx.zip` file in the **<SSPI_STAGING_FOLDER>** directory, you will see the following contents/folder structure:

Table 4-2. kaajee-1.1.0.xxx—KAAJEE folder structure

Folder/Structure	Description
..\kaajee_security_provider	This folder is the KAAJEE SSPI <root> level. This folder contains the following files: <ul style="list-style-type: none"> • <code>build.xml</code>—KAAJEE SSPI Ant build script. • <code>readme.txt</code>—KAAJEE SSPI documentation (manual), which includes an introduction, change history, any special installation instructions, and any known issues/limitations. • <code>wlKaajeeSecurityProviders-1.1.0.xxx.jar</code>—The KAAJEE SSPI software deployment jar file. • <code>wlKaajeeSecurityProviders-1.1.0.xxx.jar.MD5</code>—The MD5 checksum value for the KAAJEE SSPI software deployment jar file.
..\common_pool_jars	This folder contains the following files: <ul style="list-style-type: none"> • <code>commons-collections-3.1.jar</code> • <code>commons-dbcp-1.2.1.jar</code> • <code>commons-pool-1.2.jar</code>
..\props	This properties folder contains the following files:

Folder/Structure	Description
	<ul style="list-style-type: none"> • KaaJeeDatabase.properties • KaaJeeManageableAuthenticator.xml
..\sql	This folder contains the SQL scripts for the following databases: <ul style="list-style-type: none"> • CacheTables.sql • OracleTables.sql
..\src	This folder contains the KAAJEE SSPI Java source code (i.e., the application server software).



BEGIN: Linux Instructions

Use the "jar" command to decompress the kaaJee_security_provider_1.1.0.xxx.zip distribution file in the <SSPI_STAGING_FOLDER> staging directory:

```
<SSPI_STAGING_FOLDER> jar -xvf kaaJee_security_provider_1.1.0.xxx.zip
```



END: Linux Instructions

▶▶ Linux users, skip to 4.2.2.4.



BEGIN: Microsoft Windows Instructions

Unzip the kaaJee_security_provider_1.1.0.xxx.zip distribution file in the <SSPI_STAGING_FOLDER> staging directory.



END: Microsoft Windows Instructions

4.2.2.4 Configure the SSPI Software on the Application Server

Configure the SSPI software on the WebLogic 9.2 and higher Application Servers, in both the Admin and Managed Servers.



Shut down the WebLogic Admin and Managed Servers. Shutting down the servers ensures that the domain servers will refresh their configuration values, etc. upon startup and that the new configuration changes take effect.



BEGIN: Linux Instructions

4.2.2.4.1 (Linux: Admin Server) Modify the Startup Script

For Linux, the setDomainEnv.sh startup script needs to be modified in order for the classes contained in the SSPI, Apache connection pool jar files, and third party jar files to be found at run-time. This script is located in the following directory:

<BEA_Home>/user_project/domains/<DOMAIN_NAME>/bin/

For example:

/u01/app/bean/user_project/domains/kaajeewebdomain/bin/



NOTE: In the examples that follow, some of the directory paths are represented by their **<Alias>**, as described in Table 4-1. You can copy and paste these examples for your own use but *must* substitute the **<Alias>** placeholder with the directory information specific to your workstation.

Add SSPI Jar File to the SSPI Classpath

The KAAJEE SSPI jar file is named as follows (“xxx” is a placeholder for the build number which varies):

- **wlKaaJeeSecurityProviders-1.1.0.xxx.jar**

This file is located in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/

Add Apache Connection Pool Jar Files to the SSPI Classpath

The Apache connection pool jar files listed below are located in the directory named **<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars/**.

- commons-collections-3.1.jar
- commons-dbcp-1.2.1.jar
- commons-pool-1.2.jar

These files *must* be added to the SSPI classpath.

Edit the setDomainEnv.sh file – Create KAAJEE variables

Edit the setDomainEnv.sh file to include the classpath to the three files listed in the section above "Add Apache Connection Pool Jar Files to the SSPI Classpath," instructed as follows:



NOTE: Only the sections of setDomainEnv.sh script pertinent to demarcating file updates are displayed below.

Immediately following the standard “ADD EXTENSIONS TO CLASSPATHS” comment statement in the standard generated setDomainEnv script below,

```
# ADD EXTENSIONS TO CLASSPATHS
```

Add the following lines of code (Figure 4-8):

Figure 4-8. Linux Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.sh file
(Generic example with <Alias> placeholders)

```
# ADD EXTENSIONS TO CLASSPATHS

# Create KAAJEE variables
ApacheConnPool="<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars"
commonpool="${ApacheConnPool}/commons-pool-1.2.jar"
commondbcp="${ApacheConnPool}/commons-dbcp-1.2.1.jar"
commoncollection="${ApacheConnPool}/commons-collections-3.1.jar"
propertiesdir="<SSPI_STAGING_FOLDER>/kaajee_security_provider/props"
sspidir="<SSPI_STAGING_FOLDER>/kaajee_security_provider"
sspijar="<SSPI_STAGING_FOLDER>/kaajee_security_provider/
wlKaaJeeSecurityProviders-1.1.0.xxx.jar"
```

For the following example, we substituted the **<Alias>** placeholder as shown below:

```
<SSPI_STAGING_FOLDER> = /u01/app/bea/user_projects/domains/kaajeewebdomain
```

Figure 4-9. Linux Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.sh file
(Alias placeholders resolved with actual path names.)

```
# ADD EXTENSIONS TO CLASSPATHS

# Create KAAJEE variables
ApacheConnPool="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/common_pool_jars"
commonpool="{ApacheConnPool}/commons-pool-1.2.jar"
commondbcp="{ApacheConnPool}/commons-dbcp-1.2.1.jar"
commoncollection="{ApacheConnPool}/commons-collections-3.1.jar"
propertiesdir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/props"
sspidir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider"
sspijar="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/wlKaaJeeSecurityProviders-1.1.0.xxx.jar"
```

Add Variables to the PRE_CLASSPATH

Add the following variables (that you created in the previous steps) to the script's PRE_CLASSPATH variable:

- propertiesdir (this directory points to the KaaJeeDatabase.properties file)



NOTE: For more information on the KaaJeeDatabase.properties file, please refer to the "Edit the KaaJeeDatabase.properties File in the Props Directory" topic in this chapter.

- sspidir (this directory points to the location where you decompressed the SSPI software.)
- sspijsar (this includes the directory path listed in sspidir above plus the SSPI JAR file: wlKaaJeeSecurityProviders-1.1.0.xxx.jar)
- commonpool
- commondbcp
- commoncollection



NOTE: KAAJEE allows users to locate the file(s) pointed to by the propertiesdir and sspidir as follows:

- Co-located together in the same directory—Only one classpath is required.
- Located in separate directories—Two separate classpaths are required.

For these examples, the propertiesdir and sspidir classpaths are listed separately because they are located in separate directories.

Edit PRE_CLASSPATH

In the `setDomainEnv.sh` script, immediately after the “ADD EXTENSIONS TO CLASSPATHS” comment, and following the KAAJEE-specific variables that you set up in the preceding steps, append the following KAAJEE-specific items to the `PRE_CLASSPATH` variable:

```
# ADD EXTENSIONS TO CLASSPATHS

# Create KAAJEE variables
ApacheConnPool="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/common_pool_jars"
commonpool="${ApacheConnPool}/commons-pool-1.2.jar"
commondbcp="${ApacheConnPool}/commons-dbcp-1.2.1.jar"
commoncollection="${ApacheConnPool}/commons-collections-3.1.jar"
propertiesdir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/props"
sspidir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider"
sspijar="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/wlKaaJeeSecurityProviders-1.1.0.xxx.jar"
#
# Append KAAJEE items to PRE_CLASSPATH
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commonpool}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commondbcp}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commoncollection}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${propertiesdir}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${sspidir}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${sspijar}"
```

If you've already installed VistALink V 1.6, you may already have an addition to the `PRE_CLASSPATH` containing the directory location where the VistALink `connectorConfig.xml` file resides.

Add the sspidir Argument

Add the following `sspidir` argument:

```
-Dweblogic.alternateTypesDirectory=${sspidir}
```

This Java Virtual Machine (JVM) argument is significant because it allows WebLogic to find the appropriate directory where the custom SSPIs are located. Otherwise, WebLogic assumes that the custom SSPIs are located in the `mbeantypes` directory (e.g. `<BEA_Home>/weblogic92/server/lib/mbeantypes`). Classpaths are used by the HealthVet-VistA applications.

Somewhere AFTER the script lines setting the KAAJEE variables in the `setDomainEnv.sh` script (but before the final “export JAVA_OPTIONS” statement in the script), add the following lines of code:

```
# for KAAJEE
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.alternateTypesDirectory=${sspidir}"
```

SetDomainEnv.sh Script Changes Summary

After completing the previous steps, the complete section of modified script should look similar to the following:

```
# ADD EXTENSIONS TO CLASSPATHS

# Create KAAJEE variables
ApacheConnPool="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/common_pool_jars"
commonpool="${ApacheConnPool}/commons-pool-1.2.jar"
commondbcp="${ApacheConnPool}/commons-dbcp-1.2.1.jar"
commoncollection="${ApacheConnPool}/commons-collections-3.1.jar"
propertiesdir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/props"
sspidir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider"
sspijar="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider/wlKaaJeeSecurityProviders-1.1.0.xxx.jar"
#
# Append KAAJEE items to PRE_CLASSPATH
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commonpool}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commondbcp}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${commoncollection}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${propertiesdir}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${sspidir}"
PRE_CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${sspijar}"

# for KAAJEE
JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.alternateTypesDirectory=${sspidir}"
```



END: Linux Instructions

▶▶ Linux users, skip to 4.2.2.4.3.



BEGIN: Microsoft Windows Instructions

4.2.2.4.2 (Windows: Admin Server) Modify the setDomainEnv.cmd File

For Windows, the setDomainEnv.cmd file needs to be modified in order for the classes contained in the SSPI, Apache connection pool jar files, and third party jar files to be found at run-time. This file is located in the following directory:

`<BEA_Home>\user_project\domains\<DOMAIN_NAME>\bin\`

For example:

`C:\bea\user_project\domains\kaajeewebdomain\bin\`



NOTE: In the examples that follow, some of the directory paths are represented by their `<Alias>`, as described in Table 4-1. You can copy and paste these examples for your own use but *must* substitute the `<Alias>` placeholder with the directory information specific to your workstation.

Add SSPI Jar File to the SSPI Classpath

The KAAJEE SSPI jar file is named as follows (“xxx” is a placeholder for the build number which varies):

- `wlKaaJeeSecurityProviders-1.1.0.xxx.jar`

This file is located in the following directory:

`<SSPI_STAGING_FOLDER>\kaaje_security_provider\`

Add Apache Connection Pool Jar Files to the SSPI Classpath

The Apache connection pool jar files listed below are located in the directory named `<SSPI_STAGING_FOLDER>\kaaje_security_provider\common_pool_jars\`.

- `commons-collections-3.1.jar`
- `commons-dbc-1.2.1.jar`
- `commons-pool-1.2.jar`

These files *must* be added to the SSPI classpath.

Edit the setDomainEnv.cmd file – Create KAAJEE variables

Edit the setDomainEnv.cmd file to include the classpath to the three files listed in the section above "Add Apache Connection Pool Jar Files to the SSPI Classpath," instructed as follows:



NOTE: Only the sections of setDomainEnv.cmd script pertinent to demarcating file updates are displayed below.

Immediately following the standard “ADD EXTENSIONS TO CLASSPATHS” comment statement in the standard generated setDomainEnv script below,

```
@REM ADD EXTENSIONS TO CLASSPATHS
```

Add the following lines of code (Figure 4-10):

Figure 4-10. Windows Admin Server—KAAJEE SSPI classpath additions to the setDomainEnv.cmd file
(Generic example with <Alias> placeholders)

```
@REM ADD EXTENSIONS TO CLASSPATHS

@REM Create KAAJEE variables
set
ApacheConnPool=<SSPI_STAGING_FOLDER>\kaajee_security_provider\common_pool_jars
set commonpool=%ApacheConnPool%\commons-pool-1.2.jar
set commondbcp=%ApacheConnPool%\commons-dbcp-1.2.1.jar
set commoncollection=%ApacheConnPool%\commons-collections-3.1.jar
set propertiesdir=<SSPI_STAGING_FOLDER>\kaajee_security_provider\props
set sspidir=<SSPI_STAGING_FOLDER>\kaajee_security_provider
set sspi.jar=<SSPI_STAGING_FOLDER>\kaajee_security_provider\
wlKaaJeeSecurityProviders-1.1.0.xxx.jar
```

<Alias> placeholders.

Add Variables to the PRE_CLASSPATH

Add the following variables (that you created in the previous steps) to the script's PRE_CLASSPATH variable:

- propertiesdir (this directory points to the KaaJeeDatabase.properties file)



NOTE: For more information on the KaaJeeDatabase.properties file, please refer to the "Edit the KaaJeeDatabase.properties File in the Props Directory" topic in this chapter.

- sspidir (this directory points to the location where you unzipped the SSPI software.)
- sspijar (this includes the directory path listed in sspidir above plus the SSPI JAR file: wlKaaJeeSecurityProviders-1.1.0.xxx.jar)
- commonpool
- commondbcp
- commoncollection



NOTE: KAAJEE allows users to locate the file(s) pointed to by the propertiesdir and sspidir as follows:

- Co-located together in the same directory—Only one classpath is required.
- Located in separate directories—Two separate classpaths are required.

For these examples, the propertiesdir and sspidir classpaths are listed separately because they are located in separate directories.

Edit PRE_CLASSPATH

In the setDomainEnv.cmd script, immediately following the “ADD EXTENSIONS TO CLASSPATHS” comment, and following the KAAJEE-specific variables that you set up in the preceding steps, append the following KAAJEE-specific items to the PRE_CLASSPATH variable:

```
@REM ADD EXTENSIONS TO CLASSPATHS
@REM Create KAAJEE variables
set
ApacheConnPool=<SSPI_STAGING_FOLDER>\kaaJee_security_provider\common_pool_jars
set commonpool=%ApacheConnPool%\commons-pool-1.2.jar
set commondbcp=%ApacheConnPool%\commons-dbcp-1.2.1.jar
set commoncollection=%ApacheConnPool%\commons-collections-3.1.jar
set propertiesdir=<SSPI_STAGING_FOLDER>\kaaJee_security_provider\props
set sspidir=<SSPI_STAGING_FOLDER>\kaaJee_security_provider
set sspijar=<SSPI_STAGING_FOLDER>\kaaJee_security_provider\
wlKaaJeeSecurityProviders-1.1.0.xxx.jar
```

```
@REM Append KAAJEE items to PRE_CLASSPATH
set PRE_CLASSPATH=%PRE_CLASSPATH%;%propertiesdir%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%sspidir%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%sspijar%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commonpool%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commondbcp%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commoncollection%
```

If you've already installed VistALink 1.6, you may already have an addition to the PRE_CLASSPATH variable containing the directory location where the VistALink connectorConfig.xml file resides.

Add the sspidir Argument

Add the following sspidir argument:

```
-Dweblogic.alternateTypesDirectory=%sspidir%
```

This Java Virtual Machine (JVM) argument is significant because it allows WebLogic to find the appropriate directory where the custom SSPIs are located. Otherwise, WebLogic assumes that the custom SSPIs are located in the mbeantypes directory (e.g. <BEA_Home>\weblogic92\server\lib\mbeantypes). Classpaths are used by the Health_Vet-VistA applications.

Somewhere AFTER the script lines setting the KAAJEE variables in the setDomainEnv.cmd script (but before the final “set JAVA_OPTIONS=%JAVA_OPTIONS%” statement in the script), add the following lines of code:

```
@REM for KAAJEE
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dweblogic.alternateTypesDirectory=%sspidir%
```

SetDomainEnv.cmd Script Changes Summary

After completing the previous steps, the complete section of modified script should look similar to the following:


```

@REM ADD EXTENSIONS TO CLASSPATHS
@REM Create KAAJEE variables
set
ApacheConnPool=<SSPI_STAGING_FOLDER>\kaahee_security_provider\common_pool_jars
set commonpool=%ApacheConnPool%\commons-pool-1.2.jar
set commondbcp=%ApacheConnPool%\commons-dbc-1.2.1.jar
set commoncollection=%ApacheConnPool%\commons-collections-3.1.jar
set propertiesdir=<SSPI_STAGING_FOLDER>\kaahee_security_provider\props
set sspidir=<SSPI_STAGING_FOLDER>\kaahee_security_provider
set sspijar=<SSPI_STAGING_FOLDER>\kaahee_security_provider\
wlKaaheeSecurityProviders-1.1.0.xxx.jar

@REM Append KAAJEE items to PRE_CLASSPATH
set PRE_CLASSPATH=%PRE_CLASSPATH%;%propertiesdir%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%sspidir%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%sspijar%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commonpool%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commondbcp%
set PRE_CLASSPATH=%PRE_CLASSPATH%;%commoncollection%

@REM for KAAJEE
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dweblogic.alternateTypesDirectory=%sspidir%

```



END: Microsoft Windows Instructions



Windows users, skip to 4.2.2.4.4.



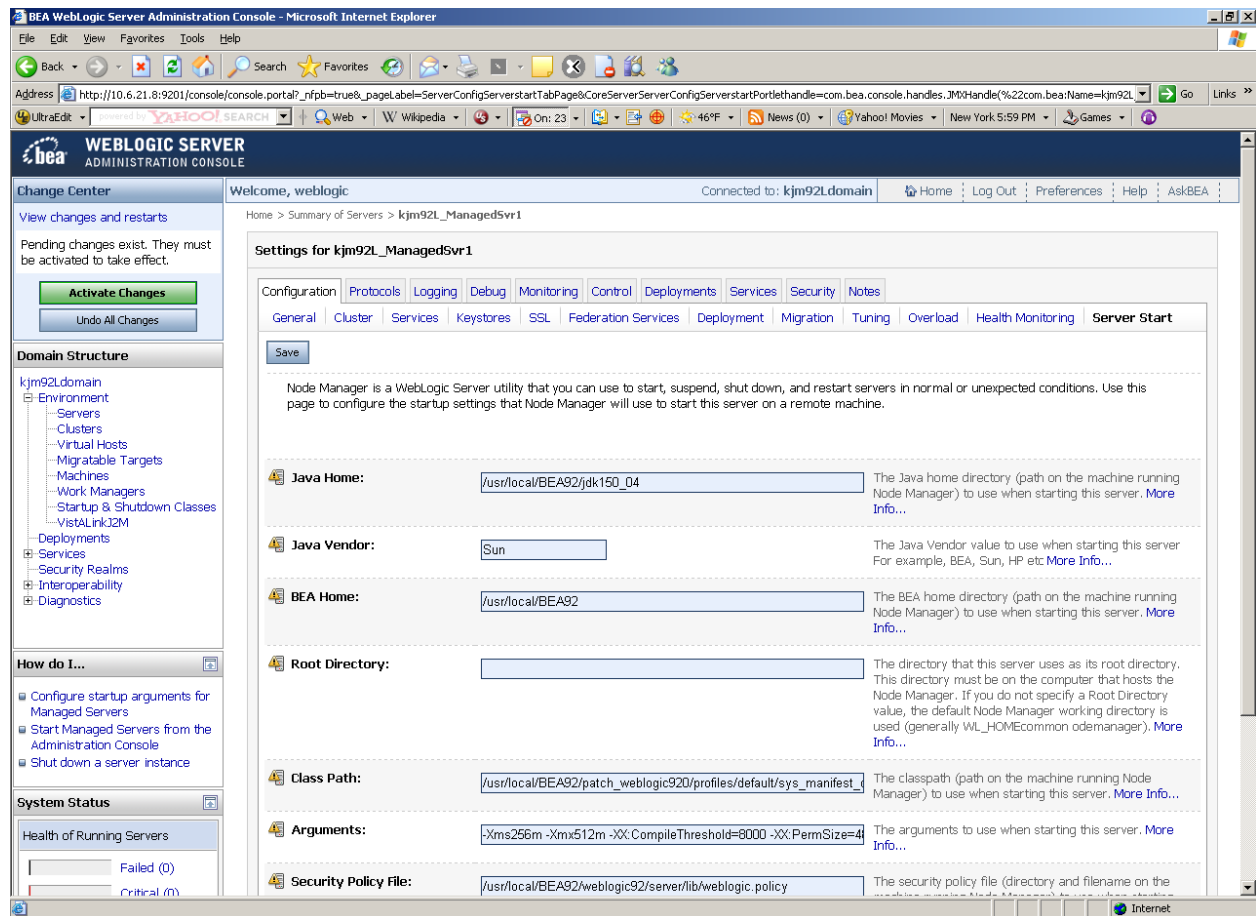
BEGIN: Linux Instructions

4.2.2.4.3 (Linux: Managed Servers) Modify the KAAJEE SSPI-related Classpath, Arguments, and Security Policy

Use the WebLogic Server Console to navigate to the **Server Start** tab on the **Configuration** tab to update the Managed Server(s) KAAJEE SSPI-related classpath and arguments.

Home > Summary of Servers > kjm92L_ManagedSvr1

Figure 4-11. WebLogic Server Administration Console: Managed Server Start tab settings





NOTE: In the examples that follow, some of the directory paths are represented by their **<Alias>**, as described in Table 4-1. You can copy and paste these examples for your own use but *must* substitute the **<Alias>** placeholder with the directory information specific to your workstation.

Users *must* repeat the following procedures for *each* Managed Server.

Add/Replace the KAAJEE SSPI Directories/Files to the Managed Server Classpath

Add or replace the following KAAJEE SSPI-related classpaths in the **Class Path** field (i.e., the classpath used to start the Managed Server) on the **Server Start** tab on the Managed Server(s):

- propertiesdir (this directory points to the KaajeeDatabase.properties file)



NOTE: For more information on the KaajeeDatabase.properties file, please refer to the "Edit the KaajeeDatabase.properties File in the Props Directory" topic in this chapter.

- sspidir (this directory points to the location where you decompressed the SSPI software.)
- wlKaajeeSecurityProviders-1.1.0.xxx.jar (SSPI JAR file)
- commons-pool-1.2.jar (file)
- commons-dbcp-1.2.1.jar (file)
- commons-collections-3.1.jar (file)



NOTE: KAAJEE allows users to locate the file(s) pointed to by the propertiesdir and sspidir as follows:

- Co-located together in the same directory—Only one classpath is required.
- Located in separate directories—Two separate classpaths are required.

For these examples, the propertiesdir and sspidir classpaths are listed separately because they are located in separate directories.

Figure 4-12. Linux Managed Server—KAAJEE SSPI classpath additions on the Server Start tab
(Generic example with <Alias> placeholders)

```

/usr/local/BEA92/patch_weblogic920/profiles/default/sys_manifest_classpath/weblog
ic_patch.jar:<JAVA_HOME>/lib/tools.jar:<BEA_HOME>/weblogic92/server/lib/weblogic_
sp.jar:<BEA_HOME>/weblogic92/server/lib/weblogic.jar:/usr/local/BEA92/weblogic92/
server/lib/webservices.jar:/usr/local/BEA92/weblogic92/common/eval/pointbase/lib
/pbclient51.jar:/usr/local/BEA92/weblogic92/server/lib/xqrl.jar:<SSPI_STAGING_FO
LDER>/kaajee_security_provider_1.1.0.xxx/props:<SSPI_STAGING_FOLDER>/kaajee_secur
ity_provider_1.1.0.xxx:<SSPI_STAGING_FOLDER>/kaajee_security_provider_1.1.0.xxx/w
lKaaJeeSecurityProviders-1.1.0.xxx.jar:
<SSPI_STAGING_FOLDER>/kaajee_security_provider_1.1.0.xxx/common_pool_jars/commons
-pool-
1.2.jar:<SSPI_STAGING_FOLDER>/kaajee_security_provider_1.1.0.xxx/common_pool_jars
/commons-dbcp-
1.2.1.jar:<SSPI_STAGING_FOLDER>/kaajee_security_provider_1.1.0.xxx/common_pool_ja
rs/commons-collections-3.1.jar: <BEA_STAGE>:
.
.
.
.

```

Other Managed Server
classpaths will follow.



NOTE: Other VistALink- and WebLogic-specific classpaths will also be displayed in this field.

For the following example, we substituted the **<Alias>** placeholders with the values as shown below:

- **<JAVA_HOME>** = /usr/local/BEA92/jdk150_04
- **<BEA_HOME>** = /usr/local/BEA92
- **<SSPI_STAGING_FOLDER>**= /usr/local/BEA92/user_projects/domains/kjm92Ldomain
- **<MANAGED_SERVER_NAME>** = kjm92L_ManagedSvr1
- **<BEA_STAGE>** = /usr/local/BEA-STAGE/kjm92Ldomain
(Staging area for applications, JCA Connectors, and configuration files)

Figure 4-13. Linux Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab
(Actual example without **<Alias>** placeholders)

```
/usr/local/BEA92/patch_weblogic920/profiles/default/sys_manifest_classpath/weblogic_patch.jar:/usr/local/BEA92/jdk150_04/lib/tools.jar:/usr/local/BEA92/weblogic92/server/lib/weblogic_sp.jar:/usr/local/BEA92/weblogic92/server/lib/weblogic.jar:/usr/local/BEA92/weblogic92/server/lib/webservices.jar::/usr/local/BEA92/weblogic92/common/eval/pointbase/lib/pbclient51.jar:/usr/local/BEA92/weblogic92/server/lib/xqrl.jar:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx/props:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx/wlKaaJeeSecurityProviders-1.1.0.xxx.jar:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx/common_pool_jars/commons-pool-1.2.jar:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx/common_pool_jars/commons-dbc-1.2.1.jar:/usr/local/BEA92/user_projects/domains/kjm92Ldomain/kaajee_security_provider_1.1.0.xxx/common_pool_jars/commons-collections-3.1.jar:/usr/local/BEA-STAGE/kjm92Ldomain:
.
.
.
.
```

Other Managed Server classpaths will follow.



NOTE: Other VistALink- and WebLogic-specific classpaths will also be displayed in this field.

Add/Replace the KAAJEE SSPI-related Arguments on the Managed Server(s)

Add or replace the following KAAJEE SSPI-related arguments on the Managed Server(s):

- -Xmx256m -Dweblogic.Name="**<MANAGED_SERVER_NAME>**"
- -Dgov.va.med.environment.srvtype=WEBLOGIC
- -Dgov.va.med.environment.production=false
- -Dlog4j.configuration=file: **<BEA_STAGE>**/log4j_managed_J2EEConfig.xml
- -Dweblogic.alternateTypesDirectory=**<SSPI_STAGING_FOLDER>**/kaajee_security_provider
- -Dweblogic.ProductionModeEnabled=""

The KAAJEE SSPI-related arguments are added/replaced in the **Arguments** field (i.e., the arguments used to start the Managed Server) on the **Server Start** tab on the Managed Server(s). The arguments are added or replaced in one long string, as shown below:

Figure 4-14. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab
(Generic example with <Alias> placeholders)

```
-Xms256m -Xmx512m -XX:CompileThreshold=8000 -XX:PermSize=48m -XX:MaxPermSize=128m
-Xverify:none -da -Dplatform.home=/usr/local/BEA92/weblogic92 -
Dwls.home=/usr/local/BEA92/weblogic92/server -
Dwli.home=/usr/local/BEA92/weblogic92/integration -
Dweblogic.management.discover=true -Dwlw.iterativeDev= -Dwlw.testConsole= -
Dwlw.logErrorsToConsole= -
Dweblogic.ext.dirs=/usr/local/BEA92/patch_weblogic920/profiles/default/sysexm_
ifest_classpath -Dgov.va.med.environment.servertype=weblogic -
Dgov.va.med.environment.production=false -Dlog4j.configuration=file://
<BEA_STAGE>/log4j_managed_J2EEConfig.xml -
Dweblogic.alternateTypesDirectory=<SSPI_STAGING_FOLDER>/kaajee_security_provider_
1.1.0.xxx -Dweblogic.Name=<MANAGED_SERVER_NAME>
```

For the following example, we substituted the <Alias> placeholders as shown below:

- <MANAGED_SERVER_NAME> = kjm92L_ManagedSvr1
- <USER_DOMAIN_HOME> = /usr/local/BEA92/user_projects/domains/kjm92Ldomain
- <SSPI_STAGING_FOLDER> = /usr/local/BEA92/user_projects/domains/kjm92Ldomain

Figure 4-15. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab
(Actual example without <Alias> placeholders)

```
-Xms256m -Xmx512m -XX:CompileThreshold=8000 -XX:PermSize=48m -XX:MaxPermSize=128m
-Xverify:none -da -Dplatform.home=/usr/local/BEA92/weblogic92 -
Dwls.home=/usr/local/BEA92/weblogic92/server -
Dwli.home=/usr/local/BEA92/weblogic92/integration -
Dweblogic.management.discover=true -Dwlw.iterativeDev= -Dwlw.testConsole= -
Dwlw.logErrorsToConsole= -
Dweblogic.ext.dirs=/usr/local/BEA92/patch_weblogic920/profiles/default/sysexm_
ifest_classpath -Dgov.va.med.environment.servertype=weblogic -
Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:///usr/local/BEA-
STAGE/kjm92Ldomain/log4j_managed_J2EEConfig.xml -
Dweblogic.alternateTypesDirectory=/usr/local/BEA92/user_projects/domains/kjm92Ldo
main/kaajee_security_provider_1.1.0.xxx -Dweblogic.Name=kjm92L_ManagedSvr1
```

Add/Replace the KAAJEE SSPI-related Security Policy File Reference

Add or replace the following KAAJEE SSPI-related security policy (permissions) file reference in the **Security Policy File** field (i.e., the security policy file used to start the Managed Server) on the **Server Start** tab on the Managed Server(s):

Figure 4-16. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab
(Generic example with <Alias> placeholders)

```
<BEA_HOME>/weblogic92/server/lib/weblogic.policy
```

For the following example, we substituted the <Alias> placeholder as shown below:

- <BEA_HOME> = /usr/local/BEA92

Figure 4-17. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab
(Actual example without <Alias> placeholders)

```
/usr/local/BEA92/weblogic92/server/lib/weblogic.policy
```



END: Linux Instructions

▶▶ Linux users, skip to 4.2.2.4.5.



BEGIN: Microsoft Windows Instructions

4.2.2.4.4 (Windows: Managed Servers) Modify the KAAJEE SSPI-related Classpath, Arguments, and Security Policy File

Use the WebLogic Server Console to navigate to the **Server Start** tab on the **Configuration** tab to update the Managed Server(s) KAAJEE SSPI-related classpath and arguments.



NOTE: In the examples that follow, some of the directory paths are represented by their **<Alias>**, as described in Table 4-1. You can copy and paste these examples for your own use but *must* substitute the **<Alias>** placeholder with the directory information specific to your workstation.

You *must* repeat the following procedures for *each* Managed Server.

Add/Replace the KAAJEE SSPI Directories/Files to the Managed Server Classpath

Add or replace the following KAAJEE SSPI-related classpaths in the **Class Path** field (i.e., the classpath used to start the Managed Server) on the **Server Start** tab on the Managed Server(s):

- propertiesdir (this directory points to the KaajeeDatabase.properties file)



NOTE: For more information on the KaajeeDatabase.properties file, please refer to the "Edit the KaajeeDatabase.properties File in the Props Directory" topic in this chapter.

- sspidir (this directory points to the location where you unzipped the SSPI software.)
- wlKaajeeSecurityProviders-1.1.0.xxx.jar (SSPI JAR file)
- commons-pool-1.2.jar (file)
- commons-dbcp-1.2.1.jar (file)
- commons-collections-3.1.jar (file)



NOTE: KAAJEE allows users to locate the file(s) pointed to by the propertiesdir and sspidir as follows:

- Co-located together in the same directory—Only one classpath is required.
- Located in separate directories—Two separate classpaths are required.

For these examples, the propertiesdir and sspidir classpaths are listed separately because they are located in separate directories.

Figure 4-18. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab
(Generic example with <Alias> placeholders)

```
C:\ALsPlace\bea\patch_weblogic920\profiles\default\sys_manifest_classpath\weblogi
c_patch.jar; <JAVA_HOME>\lib\tools.jar;
<BEA_HOME>\WEBLOG~1\server\lib\weblogic_sp.jar;
<BEA_HOME>\WEBLOG~1\server\lib\weblogic.jar;C:\ALsPlace\bea\WEBLOG~1\server\lib\w
ebsservices.jar;;C:\ALsPlace\bea\WEBLOG~1\common\eval\pointbase\lib\pbclient51.jar
;C:\ALsPlace\bea\WEBLOG~1\server\lib\xqrl.jar;;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx\props;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx\wlKaaJeeSecurityProvider
s-1.1.0.xxx.jar;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx\common_pool_jars\commons
-pool-1.2.jar;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx\common_pool_jars\commons
-dbc-1.2.1.jar;
<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx\common_pool_jars\commons
-collections-3.1.jar; <BEA_STAGE>;
.
.
.
.
```

Other Managed Server classpaths will follow.



NOTE: Other VistALink- and WebLogic-specific classpaths will also be displayed in this field.

For the following example, we substituted the **<Alias>** placeholders as shown below:

- **<JAVA_HOME>** = C:\ALsPlace\bea\JDK150~1
- **<BEA_HOME>** = C:\ALsPlace\bea
- **<SSPI_STAGING_FOLDER>** = c:\ALsPlace\bea\user_projects\domains\kjm92domain
- **<MANAGED_SERVER_NAME>** = kjm92_ManagedSvr1
- **<BEA_STAGE>** = c:\ALsPlace\bea-stage\kjm92domain
(Staging area for applications, JCA Connectors and configuration files)

Figure 4-19. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Server Start tab (Actual example without <Alias> placeholders)

```
C:\ALsPlace\bea\patch_weblogic920\profiles\default\sys_manifest_classpath\weblogic
c_patch.jar;C:\ALsPlace\bea\JDK150~1\lib\tools.jar;C:\ALsPlace\bea\WEBLOG~1\serve
r\lib\weblogic_sp.jar;C:\ALsPlace\bea\WEBLOG~1\server\lib\weblogic.jar;C:\ALsPlac
e\bea\WEBLOG~1\server\lib\webservices.jar;C:\ALsPlace\bea\WEBLOG~1\common\eval\p
ointbase\lib\pbclient51.jar;C:\ALsPlace\bea\WEBLOG~1\server\lib\xqrl.jar;C:\ALsP
lace\bea\user_projects\domains\kjm92domain\kaajee_security_provider_1.1.0.xxx\pro
ps;c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_provider_1.1
.0.xxx;c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_provider
_1.1.0.xxx\wlKaaJeeSecurityProviders-
1.1.0.xxx.jar;c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_p
rovider_1.1.0.xxx\common_pool_jars\commons-pool-
1.2.jar;c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_provide
r_1.1.0.xxx\common_pool_jars\commons-dbc-
1.2.1.jar;c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_provi
der_1.1.0.xxx\common_pool_jars\commons-collections-3.1.jar;c:\ALsPlace\bea-
stage\kjm92domain;
```

**Other Managed Server
classpaths will follow.**



NOTE: Other VistALink- and WebLogic-specific classpaths will also be displayed in this field.

Add/Replace the KAAJEE SSPI-related Arguments on the Managed Server(s)

Add or replace the following KAAJEE SSPI-related arguments on the Managed Server(s):

- -Xmx256m -Dweblogic.Name="**<MANAGED_SERVER_NAME>**"
- -Dgov.va.med.environment.servertype=WEBLOGIC
- -Dgov.va.med.environment.production=false
- -Dlog4j.configuration=file: **<BEA_STAGE>**/log4j_managed_J2EEConfig.xml
(NOTE: with forward slashes)
- -Dweblogic.alternateTypesDirectory=**<SSPI_STAGING_FOLDER>**/kaajee_security_provider

- -Dweblogic.ProductionModeEnabled=""

The KAAJEE SSPI-related arguments are added/replaced in the **Arguments** field (i.e., the arguments used to start the Managed Server) on the **Server Start** tab on the Managed Server(s). The arguments are added or replaced in one long string, as shown below:

Figure 4-20. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (Generic example with <Alias> placeholders)

```
-Xms256m -Xmx512m -XX:CompileThreshold=8000 -XX:PermSize=48m -XX:MaxPermSize=128m
-Xverify:none -da -Dplatform.home=C:\ALsPlace\bea\WEBLOG~1 -
Dwls.home=C:\ALsPlace\bea\WEBLOG~1\server -
Dwli.home=C:\ALsPlace\bea\WEBLOG~1\integration -
Dweblogic.management.discover=true -Dwlw.iterativeDev= -Dwlw.testConsole= -
Dwlw.logErrorsToConsole= -
Dweblogic.ext.dirs=C:\ALsPlace\bea\patch_weblogic920\profiles\default\sysext_manifest_classpath
-Dgov.va.med.environment.srvtype=weblogic -
Dgov.va.med.environment.production=false -Dlog4j.configuration=file://
<BEA_STAGE>/log4j_managed_J2EEConfig.xml -
Dweblogic.alternateTypesDirectory=<SSPI_STAGING_FOLDER>\kaajee_security_provider_1.1.0.xxx
-Dweblogic.Name=<MANAGED_SERVER_NAME>
```

For the following example, we substituted the <Alias> placeholders as shown below:

- <MANAGED_SERVER_NAME> = kjm92_ManagedSvr1
- <USER_DOMAIN_HOME> = c:\ALsPlace\bea\user_projects\domains\kjm92domain
- <SSPI_STAGING_FOLDER> = c:\ALsPlace\bea\user_projects\domains\kjm92domain

Figure 4-21. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Server Start tab (Actual example without <Alias> placeholders)

```
-Xms256m -Xmx512m -XX:CompileThreshold=8000 -XX:PermSize=48m -XX:MaxPermSize=128m
-Xverify:none -da -Dplatform.home=C:\ALsPlace\bea\WEBLOG~1 -
Dwls.home=C:\ALsPlace\bea\WEBLOG~1\server -
Dwli.home=C:\ALsPlace\bea\WEBLOG~1\integration -
Dweblogic.management.discover=true -Dwlw.iterativeDev= -Dwlw.testConsole= -
Dwlw.logErrorsToConsole= -
Dweblogic.ext.dirs=C:\ALsPlace\bea\patch_weblogic920\profiles\default\sysext_manifest_classpath
-Dgov.va.med.environment.srvtype=weblogic -
Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:///c:/ALsPlace/bea-stage/kjm92domain/log4j_managed_J2EEConfig.xml -
Dweblogic.alternateTypesDirectory=c:\ALsPlace\bea\user_projects\domains\kjm92domain\kaajee_security_provider_1.1.0.xxx
-Dweblogic.Name=kjm92_ManagedSvr1
```

Add/Replace the KAAJEE SSPI-related Security Policy File Reference

Add or replace the following KAAJEE SSPI-related security policy (permissions) file reference in the **Security Policy File** field (i.e., the security policy file used to start the Managed Server) on the **Server Start** tab on the Managed Server(s):

Figure 4-22. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (*Generic example with <Alias> placeholders*)

```
<BEA_HOME>\weblogic92\server\lib\weblogic.policy
```

For the following example, we substituted the **<Alias>** placeholder as shown below:

- **<BEA_HOME>** = C:\ALsPlace\bea

Figure 4-23. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Server Start tab (*Actual example without <Alias> placeholders*)

```
C:\ALsPlace\bea\weblogic92\server\lib\weblogic.policy
```



END: Microsoft Windows Instructions

4.2.2.4.5 (Oracle Database) Create KAAJEE Schema & SSPI Tables



UPGRADES: Skip this step if the DBA has already created the KAAJEE schema and SSPI tables on the Oracle database, unless it is specifically noted that changes are required in the KAAJEE software release e-mail or Web site.

Contact the DBA to create the KAAJEE user ID, schema, and SSPI tables on the Oracle database.

Create KAAJEE User ID & Schema

In summary, the DBA will need to perform the following procedures:

- Identify and create an Oracle Tablespace to hold the KAAJEE schema.
- Create a user account KAAJEE.
- Give "connect" and "resource" and "unlimited tablespace" privileges to the user account.
- The user account should have a "default" profile.
- Set the default tablespace for the KAAJEE user to the one created earlier
- Set the default "TEMP" tablespace for the KAAJEE user.



REF: For detailed step-by-step instructions on how to create a database on Oracle, please refer to the appropriate Oracle documentation.

Create KAAJEE SSPI Tables

KAAJEE requires the following two SSPI SQL database tables:

Table 4-3. Oracle Database—KAAJEE SSPI SQL table definitions

KAAJEE SSPI Table Name	Description
PRINCIPALS	This table has users and group data and is stored in an Oracle 9i database.
GROUPMEMBERS	This table has users and group mappings and is stored in an Oracle 9i database.



NOTE: We recommend that you create the KAAJEE SSPI database tables in the same schema created in the previous step.

Run the OracleTables.sql script, which can be found in the KAAJEE SSPI distribution zip file (i.e., kaajee_security_provider_1.1.0.xxx.zip) in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/sql

This SQL script creates the required KAAJEE SSPI SQL table definitions.

Use the Oracle SQL*Plus software, or other similar software of your choice, to create/edit the SSPI SQL table definitions (Table 4-3):

Figure 4-24. Oracle Database—Sample SSPI SQL script for KAAJEE table definitions

```
drop table Principals;
drop table GroupMembers;

create table Principals ( name varchar2(32) not null, isuser varchar2(10) not
null, password varchar2(32), CONSTRAINT Principals_pk PRIMARY KEY (name,isuser));

create table GroupMembers ( principal varchar2(32) not null, mygroup varchar2(32)
not null, CONSTRAINT GroupMembers_pk PRIMARY KEY (principal, mygroup));
```

Validate/Verify the Creation of the KAAJEE Database Schema & Tables

To validate/verify the creation of the KAAJEE database user ID, schema, and tables, log in as user KAAJEE.



Oracle database users, skip to 4.2.2.4.7.

4.2.2.4.6 (Caché Database) Create KAAJEE Schema & SSPI Tables



UPGRADES: Skip this step if the DBA has already created the KAAJEE schema and SSPI tables on the Caché database, unless it is specifically noted that changes are required in the KAAJEE software release e-mail or Web site.

Contact the DBA to create the KAAJEE user ID, schema, and SSPI tables on the Caché database.

Create KAAJEE User ID & Schema



REF: For detailed step-by-step instructions on how to create a database on Caché, please refer to the appropriate Caché documentation.



REF: For more information about Caché schemas, please refer to the "Caché Tables and Schemas" section located at the following Web address:

<http://vhaispwww5/migration/foundations/docs/cachedocs/archival/SQL/Caché%20Tables%20and%20Schemas.doc>

Create KAAJEE SSPI Tables

KAAJEE requires the following two SSPI SQL database tables:

Table 4-4. Caché Database—KAAJEE SSPI SQL table definitions

KAAJEE SSPI Table Name	Description
PRINCIPALS	This table has users and group data and is stored in a Caché database.
GROUPMEMBERS	This table has users and group mappings and is stored in a Caché database.



NOTE: We recommend that you create the KAAJEE SSPI database tables in the same schema created in the previous step.

Run the CacheTables.sql script, which can be found in the KAAJEE SSPI distribution zip file (i.e., kaajee_security_provider_1.1.0.xxx.zip) in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/sql

This SQL script creates the required KAAJEE SSPI SQL table definitions.

Use the Caché Terminal with the SQL DDL import, or other similar software of your choice, to import the SQL script and run it to create/edit the SSPI SQL table definitions (Table 4-4):

Figure 4-25. Caché Database—Sample SSPI SQL script for KAAJEE table definitions

```
drop table Principals;
drop table GroupMembers;

create table Principals ( name varchar(32) not null, isuser varchar(10) not null,
password varchar(32), CONSTRAINT Principals_pk PRIMARY KEY (name,isuser));

create table GroupMembers ( principal varchar(32) not null, mygroup varchar(32)
not null, CONSTRAINT GroupMembers_pk PRIMARY KEY (principal, mygroup));
```



REF: For more information about running scripts in Caché, please refer to the "Running SQL Scripts 1-4-05 JSA3.doc" document under the "Caché SQL" section located at the following Web address:

<http://vhaispwww5/migration/foundations/cache/CacheResearchTopics.htm>

Validate/Verify the Creation of the KAAJEE Database Schema & Tables

To validate/verify the creation of the KAAJEE database user ID, schema, and tables, log in as user KAAJEE.

4.2.2.4.7 Edit the KaajeeDatabase.properties File in the Props Directory

Edit the KaajeeDatabase.properties file that is distributed with the KAAJEE SSPI software (i.e., kaajee_security_provider_1.1.0.xxx.zip). The KaajeeDatabase.properties file is located in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/props

Figure 4-26. Sample KaajeeDatabase.properties file as delivered with KAAJEE

```
DriverName=oracle.jdbc.driver.OracleDriver
db_URL=jdbc:oracle:thin:@MyDatabaseHost:port:MyDB
dbUserID=scott
Password=tiger
schema=schemaName
```

Where (sample values distributed with KAAJEE SSPIs reference Oracle):

- `DriverName = oracle.jdbc.driver.OracleDriver`
- `db_URL = jdbc:oracle:thin:@MyDatabaseHost:port:MyDB`
 - Host (e.g., Oracle)
 - Port (e.g., 1521)
 - Database Name
- `dbUserID = scott`
- `Password = tiger`
- `schema = schemaName`

You should replace the values provided in this file with the appropriate values that point to your database server and database that holds the KAAJEE tables (see Step #4.2.2.4.5 [(Oracle Database) Create KAAJEE Schema & SSPI Tables"] or #4.2.2.4.6 [(Caché Database) Create KAAJEE Schema & SSPI Tables"] and Table 4-3 or Table 4-4 in this manual).



NOTE: KAAJEE requires that you use an "application-level" database user to access the KAAJEE tables in the database. Preferably, this application-level user is the same as the one you use for your own application's database operations.



REF: For more information on the KAAJEE schema, please refer to Step #4.2.2.4.5 or #4.2.2.4.6 in this manual.

Sample Oracle and Caché Database Drivers and URLs are shown below:

Figure 4-27. Oracle Database—Sample Driver and URL

```
DriverName=oracle.jdbc.driver.OracleDriver
db_URL=jdbc:oracle:thin:@host:port:MyDatabaseName
```

Figure 4-28. Caché Database—Sample Driver and URL

```
DriverName=com.intersys.jdbc.CacheDriver
db_URL=jdbc:Cache://MyDomainName:port/MyNamespace
```

The database connection pooling is implemented using JDBC. KAAJEE implements connection pooling in the SSPI via the Apache Jar file available at the following Web address:

<http://jakarta.apache.org/commons/dbcp/>

This allows the developer to make the connections to the database through the Database Connection Pool to give the best performance possible.

4.2.2.4.8 Restart the WebLogic Application Server Domain

Stop all servers in the domain. Restart the Admin server.

Wait for the Server to Come Up Before Proceeding

Restarting the admin server ensures that the domain server refreshes its configuration values, etc. and that the new configuration changes take effect.

4.2.2.4.9 Configure the Custom Security Authentication Providers in the WebLogic Application Server

Configure the Custom Security Authentication Providers in the WebLogic Application Server using the WebLogic Console. You can configure the WebLogic Application Server realms by using the WebLogic console mode, as shown in the steps that follow:

Log onto the WebLogic Server Administration Console

Log onto the WebLogic Server Administration Console using the Boot User Name and User Password.

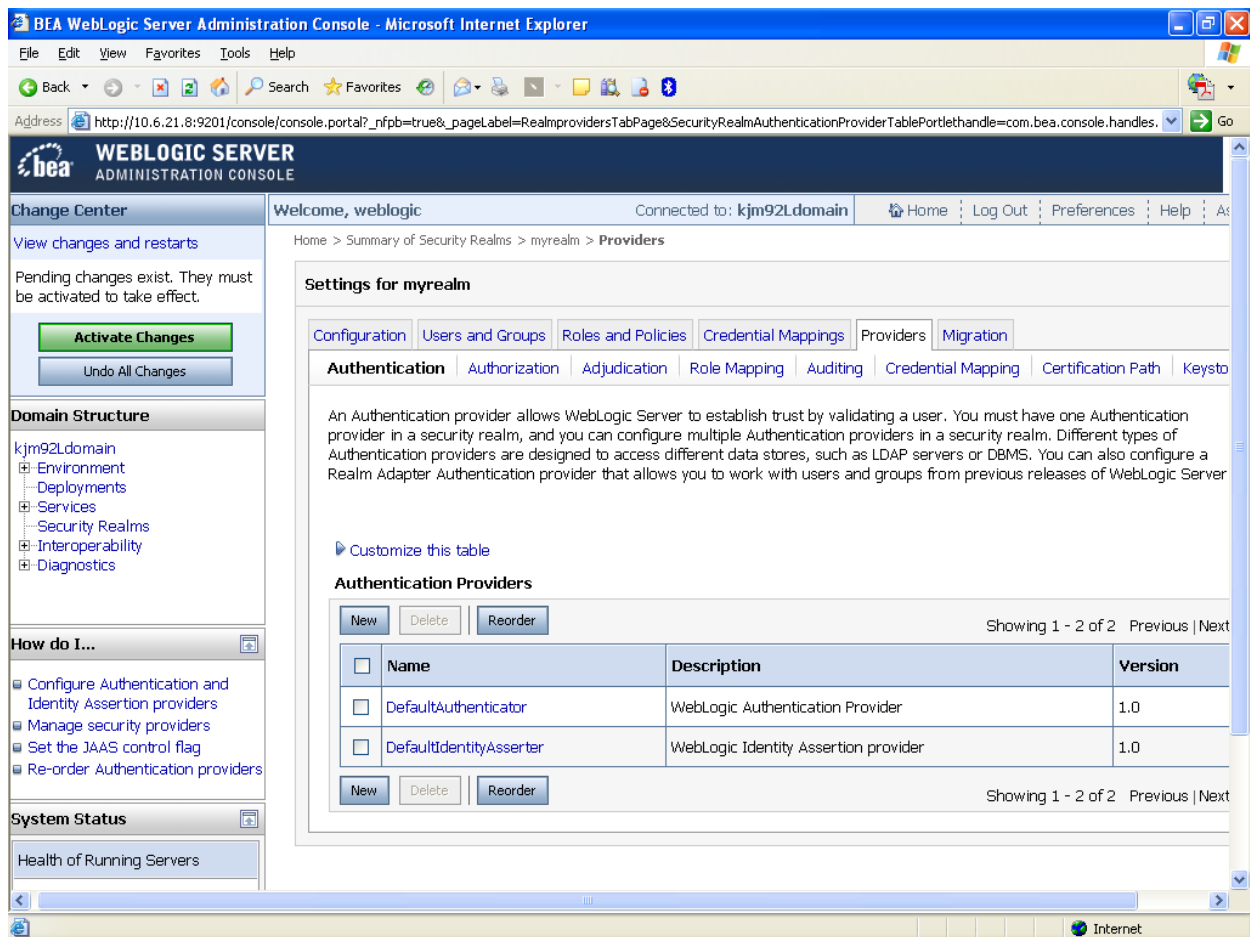
Navigate to the Authentication Directory

Select Security Realms under Domain Structure. Navigate to the Providers tab, as shown below:

Home > Summary of Security Realms > myrealm > Providers

Click on **New** in the Authentication tab, Figure 4-29:

Figure 4-29. WebLogic 9.2 and higher Server Administration Console—Select New to create new Authentication Provider



Create a New Authentication Provider

From the Providers directory, as shown below:

Home > Summary of Security Realms > myrealm > Providers

Enter **KaajeeManageableAuthenticator** for Name and select the same name in the Type pull-down menu, Figure 4-30.

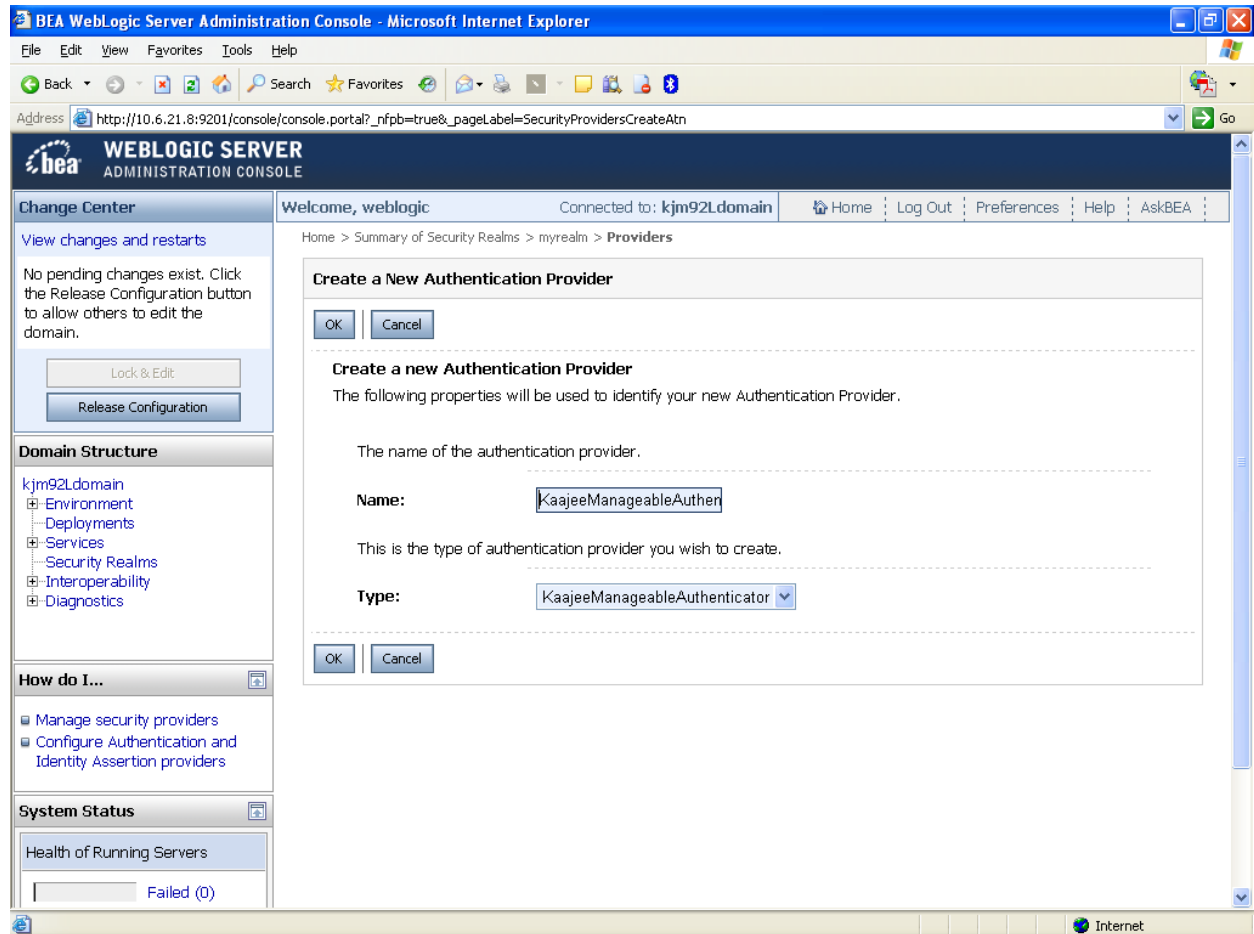


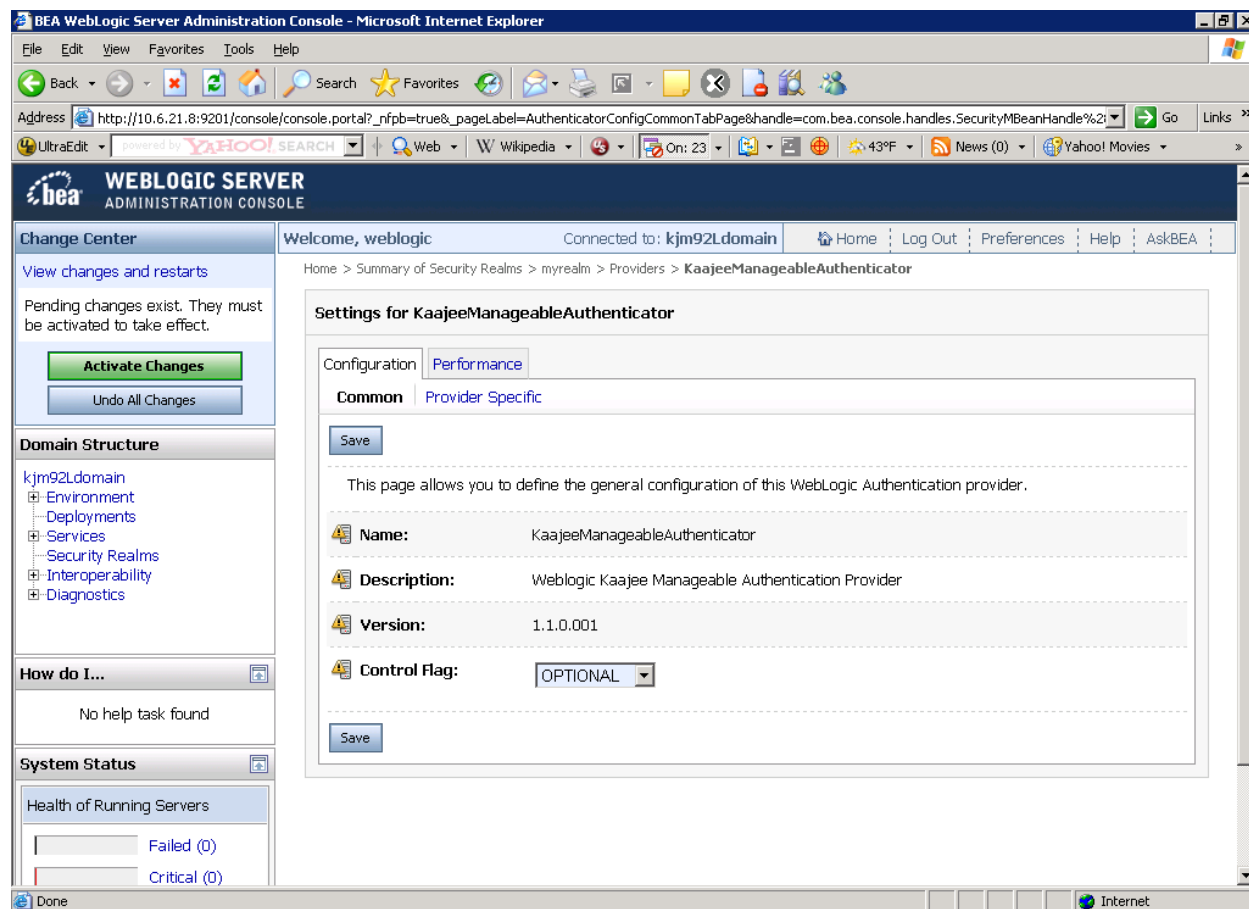
Figure 4-30. WebLogic 9.2 and higher Server Administration Console—Create a New Authentication Provider

Settings for KaaJeeManageableAuthenticator

Home > Summary of Security Realms > myrealm > Providers > KaaJeeManageableAuthenticator

Check to ensure that the Control Flag is set to the default value of OPTIONAL, Figure 4-31.

Figure 4-31. WebLogic 9.2 and higher Server Administration Console—Optional Control Flag setting for KaaJeeManageableAuthenticator



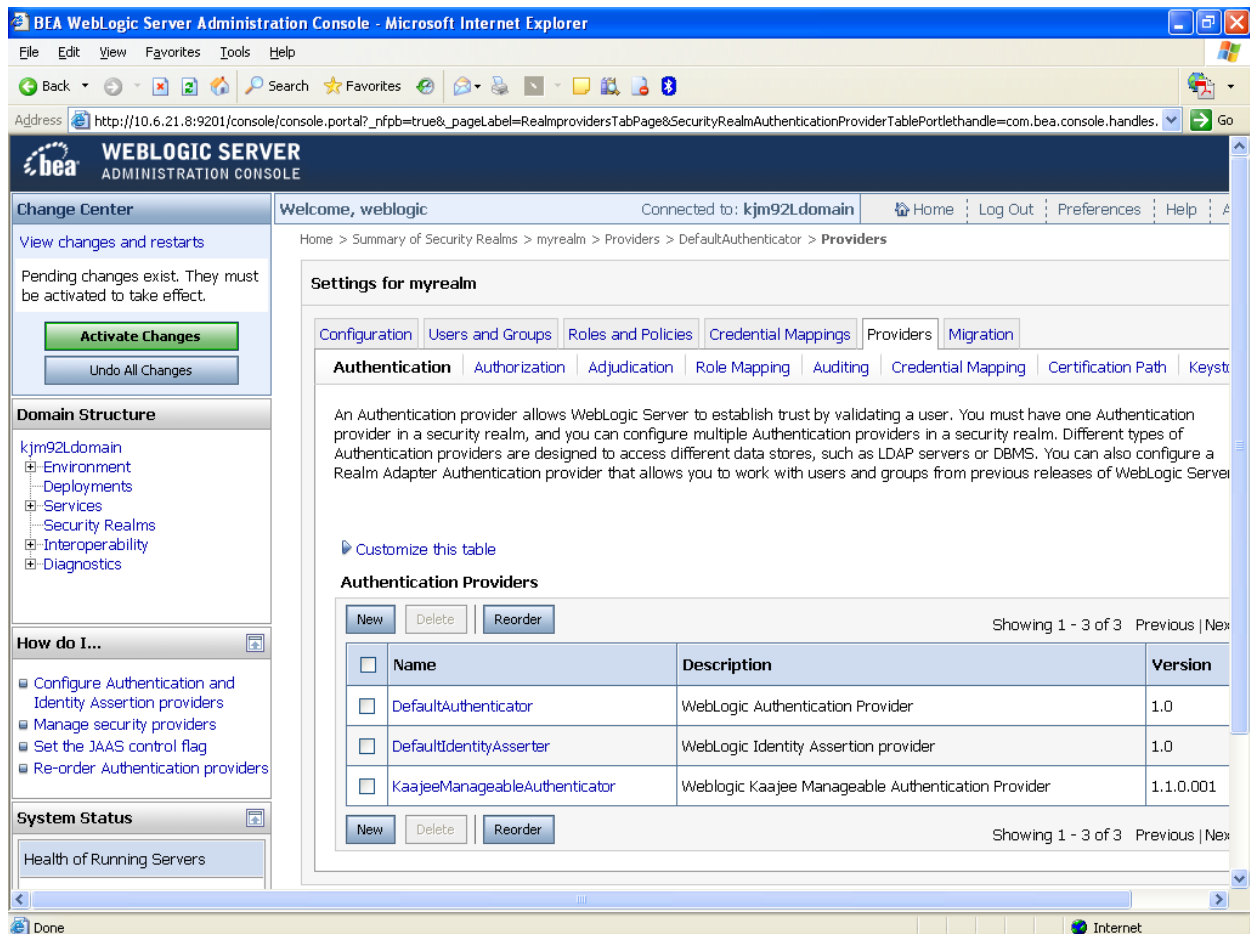
Select and Edit the Default Authenticator

This takes you back to the Authentication page on the Providers tab, as shown below:

Home > Summary of Security Realms > myrealm > Providers > KaajeeManageableAuthenticator > Providers

Select and edit the DefaultAuthenticator Authentication Provider, Figure 4-32.

Figure 4-32. WebLogic 9.2 and higher Server Administration Console—Select and edit the default Authenticator

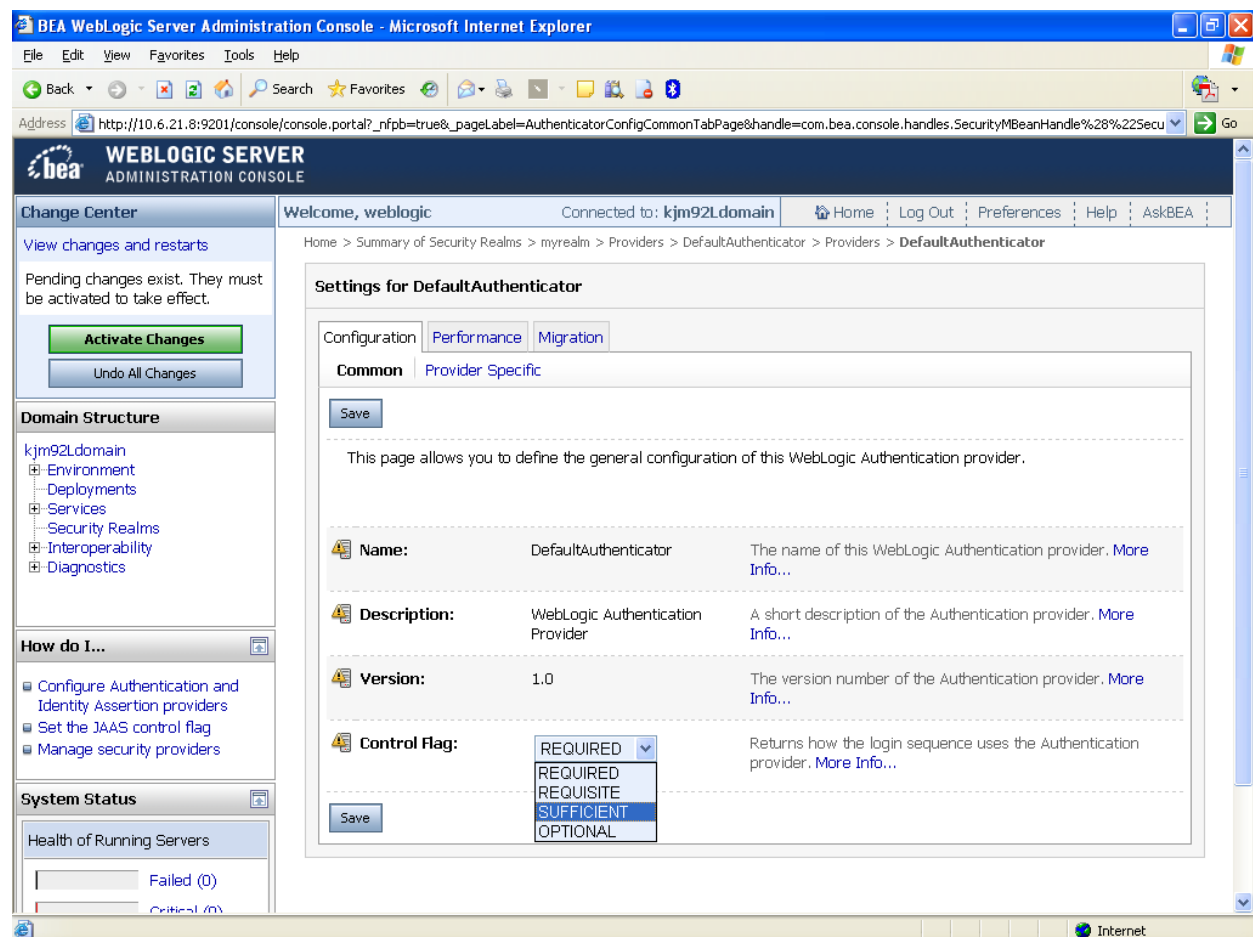


Change the Control Flag from REQUIRED to SUFFICIENT

Home > Summary of Security Realms > myrealm > Providers > DefaultAuthenticator

Use the dropdown box next to the Control Flag field to change the setting to **SUFFICIENT**, Figure 4-33.

Figure 4-33. WebLogic 9.2 and higher Server Administration Console—Change the Control Flag from REQUIRED to SUFFICIENT



Activate Changes

Activate the changes to the domain configuration by pressing Activate Changes. You should get a confirmation change that the activation was successful.

Note: If you receive the following error messages when you try to activate the new SSPI provider:

An error occurred during activation of changes, please see the log for details.
[Management:141191]The prepare phase of the configuration update failed with an exception:

[Management:141245]Schema Validation Error in config/config.xml see log for details.
Schema validation can be disabled by starting the server with the command line option: -
Dweblogic.configuration.schemaValidationEnabled=false

Add the following argument to the admin server's setDomainEnv script, and to the managed servers' "Arguments" server startup setting:

```
-Dweblogic.configuration.schemaValidationEnabled=false
```

(Do this in the same locations you are already setting the -Dweblogic.alternateTypesDirectory argument.)

Stop the WebLogic Application Server

Stop the WebLogic Application Server using the WebLogic console software (i.e., WebLogic Server 9.2 and higher Console Login).

Reboot/Restart the WebLogic Application Server

Reboot/Restart the WebLogic Application Server so all changes to the database, tables, and etc. take effect.

Verify all Changes Have Taken Place

Use the WebLogic console software (i.e., WebLogic Server 9.2 and higher Console Login) to navigate to the following locations:

- Home > Summary of Security Realms > myrealm > Users and Groups (Users tab)
- Home > Summary of Security Realms > myrealm > Users and Groups (Groups tab)



NOTE: If this is a first-time install, you will not see users populated in the Oracle tables or in the WebLogic console.

4.3 Configure SDS 13.0 (or higher) JDBC Connections with the WebLogic Server (*required*)



UPGRADES: Skip this step if you have already configured the SDS tables, unless it is specifically noted that changes are required in the KAAJEE software release e-mail or Web site.

To configure the Standard Data Services (SDS) tables for a J2EE DataSource, please refer to the "Configuring for a J2EE DataSource" topic in the *SDS API Installation Guide*.



REF: The *SDS API Installation Guide* is included in the SDS software distribution ZIP files, which are available for download at the following Web address:

http://vaww.sts.infoshare.va.gov/STS_SDS/Project%20Artifacts/Forms/AllItems.aspx

4.4 Ensure the Existence of, or Create, a KAAJEE User with Administrative Privileges (*required*)

For KAAJEE to execute correctly, the files web.xml and weblogic.xml has content that declares that KAAJEE will run with the needed privileges.

Check that your WebLogic server already has a user named "KAAJEE" and is part of the **Administrators** group, or it is part of the **Admin** global security role. If there is such a user, your installation of the KAAJEE enable web application will execute properly.

WebLogic Security Realm:

If you need to create a new user in WebLogic, ensure that

1. It is named **KAAJEE**
2. It is assigned to the **Administrators** group

Active Directory Authentication Provider:

If your WebLogic domain has integrated an Active Directory authentication provider, and you will be creating the user in Active Directory, ensure that

1. It is named **KAAJEE**
2. The user is part of a group that can be mapped in the WebLogic security realm to the Global Security Role named **Admin**.

The following shows the contents of the web.xml and weblogic.xml files as it pertains to the **KAAJEE** user.



REF: See the KAAJEE Deployment Guide for the contents of the web.xml and weblogic.xml files and additional details.

web.xml:

This file has a <run-as> tag, which causes it to run with the necessary administrative privileges. In addition, a corresponding security-role tag is defined. See the sample in Figure 4-34.

Figure 4-34. Sample excerpt from a web.xml file—Using the run-as and security-role tags

```
<servlet>
  <servlet-name>LoginController</servlet-name>
  <servlet-class>
    gov.va.med.authentication.kernel.servlet.LoginController
  </servlet-class>
  <run-as>
    <role-name>adminuserrole</role-name>
  </run-as>
</servlet>

<security-role>
  <role-name>adminuserrole</role-name>
</security-role>
```

weblogic.xml:

This file has a <run-as> tag, which causes it to run as an administrative user whose username is “KAAJEE.” In addition, a corresponding security-role tag is defined. See the sample in Figure 4-37.

Figure 4-35. Sample excerpt from a weblogic.xml file—Using the run-as-role-assignment tag

```
<run-as-role-assignment>
  <role-name>adminuserrole</role-name>
  <run-as-principal-name>KAAJEE</run-as-principal-name>
</run-as-role-assignment>
```



Important! The “KAAJEE” user or alternate must exist in the WebLogic Application server and have system administration privileges.

4.5 Edit the KAAJEE Configuration File (*required*)

4.5.1 Locate the kaajeeConfig.xml File (*required*)

The EMC or Application Server Administrator must first locate the kaajeeConfig.xml file in the Web application ear or standalone war file, as follows:

Exploded Ear Files

Navigate to the WEB-INF directory in the application's exploded ear/war file—Locate the KAAJEE configuration file (i.e. kaajeeConfig.xml)

Ear Files

1. Unzip the application's ear file—Explode the artifact.
2. For any war file that implements KAAJEE authentication inside the ear file, unzip the war file.
3. Navigate to the WEB-INF directory—Locate the KAAJEE configuration file (i.e. kaajeeConfig.xml)

Standalone War Files

1. Unzip the application's war file that implements KAAJEE authentication.
2. Navigate to the WEB-INF directory—Locate the KAAJEE configuration file (i.e. kaajeeConfig.xml)

The following is a sample excerpt of the kaajeeConfig.xml file as distributed with KAAJEE 1.1.0.007:

Figure 4.5-1. Sample Station Number excerpt of the kaajeeConfig.xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<kaajee-config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="kaajeeConfig.xsd">

  <!-- host application name, used for login page display and logging -->
  <host-application-name>KAAJEE Sample</host-application-name>

  <!-- put each station number for KAAJEE login here -->
  <login-station-numbers>
    <station-number>###</station-number>
    <station-number>###9XX</station-number>
    <station-number>###9XX</station-number>
    <station-number>###XX</station-number>
    <station-number>###XX</station-number>
    <station-number>###</station-number>
    <station-number>###9XX</station-number>
    <station-number>###9XX</station-number>
    <station-number>###XX</station-number>
    <station-number>###XX</station-number>
  </login-station-numbers>
  .
  .
  .
</kaajee-config>
```

Users *must* initially configure this file to change these (placeholder) Station Numbers, as distributed with KAAJEE.

4.5.2 Edit the Station Number List in the kaajeeConfig.xml File (*required*)

Use a text editor (e.g., Microsoft Notepad) or other xml editing software to open and edit the kaajeeConfig.xml file. The <station-number> tags control the Station Number list displayed to the end-user in KAAJEE's login Web page Institution drop-down list. In Figure 4.5-1, we represent the application-specific Station Numbers as placeholders displayed in bold typeface beginning with "###".

In the kaajeeConfig.xml file, you *must* replace these placeholder Station Number values with the appropriate valid values for the user to log into for your Web-based application. You can specify both division-level and facility-level Station Numbers, as appropriate for your application. To be valid, the values entered *must* be recognized by Standard Data Services (SDS).



NOTE: For every login Station Number you enter here, KAAJEE uses this as the Station Number parameter it passes to VistALink's Institution Mapping to retrieve a JNDI connector name for VistALink; therefore, every login station number should have a mapping configured in VistALink's Institution Mapping.



REF: For more information on the kaajeeConfig.xml file, please refer to Chapter 6, "KAAJEE Configuration File," in the *KAAJEE Deployment Guide*.

4.5.3 Redeploy and Test the Web Application with the Updated kaajeeConfig.xml File (*required*)

Use WebLogic to redeploy the Web application ear or standalone war file with the updated kaajeeConfig.xml file on all appropriate application servers. Test the redeployed application.

Exploded Ear Files

Leave application as an exploded ear file.

Packaged Ear Files

1. Zip any unzipped war files that implements KAAJEE authentication into a war, replacing the old war file.
2. Zip up the application ear file.

Standalone War Files

Zip any unzipped war files into a war, replacing the old war file.

4.6 (Linux/Windows) Configure log4j for All J2EE-based Application Log Entries (*required*)



UPGRADES: Skip this step if you have already configured log4j *and* added the KAAJEE-specific logger information to the active log4j configuration file on the application server, unless it is specifically noted that changes are required in the KAAJEE software release e-mail or Web site.

In order to provide a unified logger and consolidate all log/error entries into one file, all J2EE-based application-specific loggers *must* be added to the same log4j configuration file, which should be the active log4j configuration file for the server. After locating the active log4j configuration file used on the server you are configuring (e.g., mylog4j.xml file), add in the KAAJEE (and FatKAAT) loggers to that file.

To locate the active log4j configuration file, look for the "-Dlog4j.configuration=" argument in the startup script file (i.e., setDomainEnv.sh/.cmd). The "-Dlog4j.configuration=" should be set to the absolute location of the configuration file (e.g., c:/mydirectory/mylog4j.xml). If no such argument is present, look for a file named "log4j.xml" in a folder on the server classpath.

You *must* configure log4j for the first time, if all three of the following conditions exist:

- The "-Dlog4j.configuration=" argument does *not* exist in the WebLogic JVM startup script files.
- The "log4j.xml" file does *not* exist in the classpath.
- There is no pre-existing log4j configuration file in the folder placed on the classpath of the WebLogic Application Server containing the configuration files for all HealthVet-VistA J2EE applications (e.g., <HEV CONFIGURATION FOLDER>).

For first time log4j configuration procedures, please refer to the "log4j Configuration File" topic in the *VistALink Installation Guide (1.6)*. Also, sample log4j configuration files are included with the VistALink 1.6 software distribution.



REF: For more information on VistALink, please refer to the Application Modernization Foundations Web site located at the following Web address:

<http://vaww.vista.med.va.gov/vistalink>

Once the log4j file is initially configured, you need to configure the file specifically for KAAJEE log entries as outlined below.



REF: For more information on log4j guidelines, please refer to the Application Structure & Integration Services (ASIS) *Log4j Guidelines for HealtheVet-VistA Applications* document available at the following Web address:

<http://vista.med.va.gov/vistaarch/healthvet/Documents/Log4j%20Guidance%201.0.doc>

4.6.1 Configure Application for log4j

Follow the Log4J instructions (<http://jakarta.apache.org/log4j/docs/>) to configure your application for Log4J.

4.6.2 Edit the File Name and Location for All Log Entries

Edit the "verboseDailyRollingFileAppender" <appender name> tag in the active log4j configuration file (e.g., mylog4j.xml file). The "File" <param name> tag should point to the common file name and location where all J2EE-based application daily log entries for that domain will be recorded, as shown below:

Figure 4-2. Sample excerpt of the mylog4j.xml file—Editing common log file name and location (Windows)

```
<appender name="verboseDailyRollingFileAppender"
class="org.apache.log4j.DailyRollingFileAppender">
  <param name="File"
value="C:/AllAppData/bea/user_projects/domains/AllAppDomain/log/AllApp.log"/>
  <param name="DatePattern" value="'.'yyyy-MM-dd"/>
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%-4r %d{ISO8601} [%t] %-5p
%C:%M:%L - %m%n"/>
  </layout>
</appender>
```

In this example (Figure 4-37), the following common log file name and location is indicated:

`C:/AllAppData/bea/user_projects/domains/AllAppDomain/log/AllApp.log`

The application server administrator should point to the same log file established for that domain on the application server where *all* J2EE-based applications are logging their entries.

4.6.3 Add KAAJEE-specific Logger Tags

Add the following four KAAJEE-specific logger tags to the active log4j configuration file (e.g., mylog4j.xml file) on the application server:

- gov.va.med.authentication.kernel
- gov.va.med.authentication.kernel.cactus



NOTE: Figure 4-3 shows the detailed logger tag information that *must* be added to the active log4j configuration file (e.g., mylog4j.xml file) for KAAJEE.

Generally, the log level should be set as follows:

- Integrating KAAJEE—Set log level to DEBUG.
- Normal Operation Mode—Set log level to ERROR.

The following figure shows the detailed logger tag information that *must* be added to the active log4j configuration file (e.g., mylog4j.xml file) for KAAJEE:

Figure 4-3. Sample excerpt of the mylog4j.xml file—Adding KAAJEE logger information

```
.
.
.

<logger name="gov.va.med.authentication.kernel" additivity="false" >
    <level value="debug" />
    <appender-ref ref="verboseDailyRollingFileAppender"/>
</logger>

<logger name="gov.va.med.authentication.kernel.cactus" additivity="false" >
    <level value="debug" />
    <appender-ref ref="verboseDailyRollingFileAppender"/>
</logger>
.
.
.
```



NOTE: The log level value in this sample log4j.xml configuration file is currently set to "debug" mode for KAAJEE-related logger entries. To set those logger entries to normal operations you would change "debug" to "error."



Congratulations! You have now completed the installation and configuration of KAAJEE-related software on the WebLogic Application Server.

Upon completing the installation of KAAJEE-related software on the VistA M Server and WebLogic Application Server, you are now ready to develop/run HealthVet-VistA Web-based applications that use KAAJEE.

Appendix A: Installation Back-Out or Roll-Back Procedure

KAAJEE 1.1 comprises both a Java and an M component, similar to KAAJEE 1.0.

VistA M Server

The M component in KAAJE 1.1 (Patch XU*8*504) includes a new routine but doesn't change any of the existing KAAJEE 1.0 routines. You can use the Kernel Installation and Distribution System (KIDS) option Backup a Transport Global [XPD BACKUP] to remove this routine.

If the installation fails, the recommended actions are as follows:

1. Review the install logs.
2. Determine and address the cause of install failure.
3. Re-run the installation.

Optionally delete the following KAAJEE 1.1 components:

- Routine:
 - XUSKAAJ1
- Options:
 - XUS KAAJEE PROXY LOGON
 - XUS KAAJEE WEB LOGON
- Remote Procedure Calls (RPCs):
 - XUS KAAJEE GET CCOW TOKEN
 - XUS KAAJEE GET USER VIA PROXY
- Security Key:
 - XUKAAJEE_SAMPLE
- Application Proxy:
 - KAAJEE,PROXY

J2EE Application Server

The Java component in KAAJE 1.1 is not a stand-alone Web application. It is an embedded set of Java components and a Java library to be embedded within a consuming Web application. Therefore, there is no back-out/roll-back procedure specific to KAAJEE 1.1. Each consuming application would have to devise their own back-out/roll-back procedure.

