KERNEL AUTHENTICATION & AUTHORIZATION FOR J2EE (KAAJEE)

RELEASE NOTES & INSTALLATION GUIDE

KAAJEE Version 1.0.1.xxx &
SSPI Version 1.0.0.010

April 2009

Department of Veterans Affairs (VA)
Office of Information & Technology (OI&T)
Office of Enterprise development (OED)

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.

Date	Revision	Description	Author(s)
04/02/09	1.0	Initial KAAJEE 1.0.1.xxx documentation. **KAAJEE 1.0.1.003** **SSPI 1.0.0.010** **Kernel Patch XU*8.0*451**	KAAJEE Development Team, Oakland, CA Oakland Office of Information Field Office (OIFO): • REDACTED

Table i. Documentation revision history

Patch Revisions

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

NOTE: Kernel is the designated custodial software application for KAAJEE; however, KAAJEE comprises multiple patches and software releases from several Health***e***Vet-VistA applications.

REF: For the specific VistA M Server software patches required for the implementation of KAAJEE, please refer to Table 2-2 in this manual.

Contents

Figures and Tables

Figures

Tables

Acknowledgements

The Kernel Authentication and Authorization Java (2) Enterprise Edition (KAAJEE) development team consists of the following Office of Enterprise Development (OED) personnel (listed alphabetically within a category):
REDACTED

The KAAJEE development team would like to thank the following sites/organizations/personnel for their assistance in reviewing and/or testing KAAJEE-related software and documentation (project development teams are listed alphabetically):

- Emergency Department Integration System (EDIS)—Development Team
- Spinal Cord Dysfunction (SCD)

Orientation

This manual is intended for use in conjunction with the installation of the Kernel Authorization and Authentication for J2EE (KAAJEE) software. It outlines the details of KAAJEE-related software and gives guidelines on how the software is installed within Health***e***Vet-Veterans Health Information Systems and Technology Architecture (VistA).

The intended audience of this manual is all key stakeholders. The primary stakeholder is the Office of Enterprise Development (OED). Additional stakeholders include:

- Health***e***Vet-VistA application developers of Web-based applications in the WebLogic 8.1 (SP4 or higher) Application Server environment.
- Information Resource Management (IRM) and Information Security Officers (ISOs) at Veterans Affairs Medical Centers (VAMCs) responsible for computer management and system security.
- Enterprise Product Support (EPS).
- VAMC personnel who will be using Health***e***Vet-VistA Web-based applications running in the WebLogic 8.1 (SP4 or higher) Application Server environment.

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 Health***e***Vet-Veterans Health Information Systems and Technology Architecture (VistA) software products.

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

- Linux (i.e., Red Hat Enterprise ES 3.0)
- 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:

Symbol	Description
ℹ	**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/VIRGIN INSTALLATION:** Used to denote Upgrade or Virgin 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) are set off and indicated with this Linux "Tux" penguin icon.
wind ows	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.

Table ii. Documentation symbol/term descriptions

- 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.

  ℹ **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).

How to Obtain Technical Information Online

Exported VistA M Server-based file, routine, and global documentation can be generated through the use of Kernel, MailMan, and VA FileMan utilities.

ℹ **NOTE:** Methods of obtaining specific technical information online will be indicated where applicable under the appropriate topic.

Help at Prompts

VistA M Server-based software provides online help and commonly used system default prompts. Users are encouraged to enter question marks at any response prompt. At the end of the help display, you are immediately returned to the point from which you started. This is an easy way to learn about any aspect of VistA M Server-based software.

Obtaining Data Dictionary Listings

Technical information about VistA M Server-based files and the fields in files is stored in data dictionaries. You can use the List File Attributes option on the Data Dictionary Utilities submenu in VA FileMan to print formatted data dictionaries.

REF: For details about obtaining data dictionaries and about the formats available, please refer to the "List File Attributes" chapter in the "File Management" section of the *VA FileMan Advanced User Manual*.

Assumptions about the Reader

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

- Health***e***Vet-VistA computing environment:
  - Kernel—VistA M Server software
  - Remote Procedure Call (RPC) Broker—VistA M Server software
  - VA FileMan data structures and terminology—VistA M Server software
  - VistALink—VistA M Server and Application Server software
- Linux or Microsoft Windows environment
- Java:
  - Java Programming language
  - Java 2 Standard Edition (J2SE) Java Development Kit (JDK, a.k.a. Java Software Development Kit [SDK])
- M programming language
- WebLogic 8.1 (SP4 or higher)—Application server
- Oracle 9i—Database (e.g., Security Service Provider Interface [SSPI])
- 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 VistA Automated Access Request software; however, no attempt is made to explain how the overall Health***e***Vet-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 World Wide Web (WWW) and VA Intranet for a general orientation to Health***e***Vet-VistA. For example, go to the Department of Veterans Affairs (VA) Office of Information & Technology (OI&T) VistA Development Intranet Website:

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*, this manual
- *Kernel Authentication & Authorization for J2EE (KAAJEE) Deployment Guide*
- KAAJEE Website: 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 VistALink documentation located on the VHA Software Document Library (VDL) Website at the following Website:

http://www.va.gov/vdl/application.asp?appid=163

Health***e***Vet-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 Website:

http://www.adobe.com/

Health***e***Vet-VistA documentation can be downloaded from the Office of Information and Technology (OI&T) VHA Software Document Library (VDL) Website:

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

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

- Preferred Method REDACTED

  This method transmits the files from the first available FTP server.
- Albany OIFO REDACTED
- Hines OIFO REDACTED
- Salt Lake City OIFO REDACTED

DISCLAIMER: The appearance of any external hyperlink references in this manual does not constitute endorsement by the Department of Veterans Affairs (VA) of this Website 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 Release Notes

The Health*e*Vet-Veterans Health Information Systems and Technology Architecture (VistA) Kernel Authentication and Authorization for Java (2) Enterprise Edition (KAAJEE) Version KAAJEE 1.0.1.xxx software is now available. As of the original release of KAAJEE 1.0.0.019, this enhanced software has the following features:

- **Enhanced Login Functionality:**
  - Removed **Refresh** button from KAAJEE login page—Previously, the **Refresh** button was used to refresh the sort order of the station numbers in the Institutions dropdown box (either sorted by number or by name). Adversely, this button would also reset the Access and Verify codes to null when pressed. Also, because the **Refresh** button was the default button (focus) rather than the **Login** button, the user's Access and Verify codes would be reset to null whenever users pressed the **<Enter>** key. Thus, users had to remember to manually tab to and press the **Login** button rather than just pressing **<Enter>**. This version of KAAJEE removes the need for the **Refresh** button by sorting the Station Numbers in the background on the client via JavaScript code rather than on the server. Therefore, the **Login** button is now the default button and will *not* reset the user's Access and Verify codes when the **<Enter>** key is pressed.
  - Added JavaScript code for client-side sorting of Institutions—JavaScript will be used in conjunction with the existing sort radio buttons to change the sorting of station numbers in the Institution dropdown box. The actual sorting will now occur on the client instead of the server.
  - Provided Access code**;**Verify code capability in one line—An inconsistency was noted between the login capabilities of KAAJEE Web logins and VistA M Server legacy application logins. Users were accustomed to entering both their Access code and Verify code separated by a semicolon ("**;**") in a single line (e.g., **accesscode;verifycode**) on VistA M Server legacy applications; however, that capability did not exist with KAAJEE Web logins. This version of KAAJEE addresses this issue by now permitting both the Access code and Verify code to be entered in the **Access Code** text box separated by a semicolon on the KAAJEE login Web page.
  - Added support for parameter passing of Default Institution and Institution sorting preferences.. When the consuming application provides a web URL link to their protected Web pages, they can now include a list of KAAJEE supported parameters. In addition, each login user can create shortcuts to these links on their desktops. The login user can then edit these shortcuts to include the desired parameters and corresponding values. The use of these shortcuts can be used as a workaround to the issue of using persistent cookies when using Thin Clients and Terminal Servers. The issue of using persistent cookies on Terminal Servers is that they are often not retained as part of the roaming user profile upon logout and disconnect.
  - Made the KAAJEE Login Web page more Section 508 friendlier.
- **Added Sample Web Application—**In order to test the KAAJEE login, you need to deploy a consuming J2EE Web Application that is configured to use KAAJEE. Therefore, this version of KAAJEE provides a standalone KAAJEE Sample Web Application that can be used to test KAAJEE. In addition, this application can be used by J2EE developers as a sample to assist them in configuring their J2EE Web-based application to use KAAJEE. Therefore, this sample Web

application can be used by Web administrators, SQA, Testing Services, J2EE developers, and support personnel. The KAAJEE Sample Web Application also requires the allocation of the XUKAAJEE_SAMPLE VistA M Server security key (see description that follows).

- **Added VistA M Sever Security—**Kernel Patch XU*8.0*451 added a new XUKAAJEE_SAMPLE VistA M Server security key that is required to run the KAAJEE Sample Web Application (see description above).

  In KAAJEE, first, an initial authentication occurs against a VistA M Server (i.e., Access and Verify codes). Then, if the login user passes this phase, the XUKAAJEE_SAMPLE VistA M security key is used to create a J2EE group/principal of the same name on the J2EE Application Server if not already created. In addition, the login user will be assigned membership to this group on the J2EE Application Server during the login session. This membership is necessary as the authorization aspect of Form-based Authentication validates the role-based access by the membership of the associated group/principal.

- **Updated Software Version Support:**
  - Compiled and tested KAAJEE against Standard Data Services (SDS) 13.0.
  - Compiled and tested KAAJEE against VistALink 1.5.1.002.

- **Made Bug Fixes:**
  - Kernel Patch XU*8.0*469 fixed the issue with KAAJEE login *not* updating LAST SIGN-ON DATE/TIME field (#202) in the NEW PERSON file (#200)—User deactivation can occur if the KAAJEE login user infrequently logs into the VistA M system outside of the KAAJEE login Web page. This user deactivation will occur after 90 days after the last successful login outside of a KAAJEE login. If the KAAJEE login user never logs in outside of KAAJEE, this deactivation will occur after 30 days from the user creation date or last edit date. The Kernel option that performs this check, if tasked/scheduled, is the Automatic Deactivation of Users option [XUAUTODEACTIVATE].

    This bug was entered via a Remedy ticket, which was identified as HD0000000166523. The fix for this bug was made in subroutine SLOG of the XUS1 routine. Kernel Patch XU*8.0*451 updates the XUS1 routine. With this patch installed, the LAST SIGN-ON DATE/TIME field (#202) associated with the login user will be updated each time the login user successfully logs in through KAAJEE.

Users can check this by seeing whether this field is updated after successfully logging into VistA through KAAJEE. To check this using VA FileMan do the following:

```
>D Q^DI


VA FileMan 22.0


Select OPTION: INQUIRE TO FILE ENTRIES



OUTPUT FROM WHAT FILE: SECURITY KEY// NEW PERSON <Enter>    (229 entries)
Select NEW PERSON NAME: TESTER,KAAJEE WEBKAT <Enter>       KWT

ANOTHER ONE: <Enter>
STANDARD CAPTIONED OUTPUT? Yes// <Enter>  (Yes)
Include COMPUTED fields:  (N/Y/R/B): NO// <Enter> - No record number
(IEN), no Computed Fields
DISPLAY AUDIT TRAIL? No// <Enter>  NO

NAME: TESTER,KAAJEE WEBKAT               INITIAL: KWT
  ACCESS CODE: <Hidden>                  FILE MANAGER ACCESS CODE: #
  VERIFY CODE never expires: Yes
  DATE VERIFY CODE LAST CHANGED: MAR 30,2007
  VERIFY CODE: <Hidden>                  SEX: MALE
  PREFERRED EDITOR: SCREEN EDITOR - VA FILEMAN
  DATE ENTERED: APR 30, 2004             CREATOR: XUUSER,ONE
  SSN: 777777777
  LAST SIGN-ON DATE/TIME: JUN 28, 2007@11:53:53
  XUS Logon Attempt Count: 0             XUS Active User: No
  TERMINAL TYPE LAST USED: C-VT320
  NAME COMPONENTS: 200                   SERVICE/SECTION: IRM
  SIGNATURE BLOCK PRINTED NAME: KAAJEE WEBKAT TESTER
KEY: PATS-SIT                            GIVEN BY: XUUSER,ONE
  DATE GIVEN: APR 30, 2004
KEY: PATSSIT                             GIVEN BY: XUUSER,ONE
  DATE GIVEN: MAY 30, 2004
KEY: ALSTESTGROUP                        GIVEN BY: XUUSER,ONE
  DATE GIVEN: OCT 20, 2004
KEY: XUFATKAAT_SAMPLE                    GIVEN BY: XUUSER,ONE
  DATE GIVEN: MAR 11, 2005

Enter RETURN to continue or '^' to exit:
KEY: XUKAAJEE_SAMPLE                     GIVEN BY: XUUSER,ONE
  DATE GIVEN: JUN 14, 2007
  MULTIPLE SIGN-ON: ALLOWED              TIMED READ (# OF SECONDS): 900
  PRIMARY MENU OPTION: XMUSER
```

Figure 2.1-1. Verifying KAAJEE updates to the LAST SIGN-ON DATE/TIME field (#202) in the NEW PERSON file (#200)

If the field shows up with a current date/time, then it is being updated.

You would need to perform this check before and after installing this M-side patch.

- Fixed Response already committed error—The code that was fixed was associated with processing the persistent cookie information on the Application Server. This fix should also fix the extra M process that was created.

2 Pre-Installation Instructions

2.1 Purpose

The purpose of this guide is to provide instructions for installing the Health*e*Vet-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 signon architecture for Health*e*Vet-VistA Web-based applications. For example:

- Blind Rehab
- Patient Advocate Tracking System (PATS)
- Veterans Personal Finance System (VPFS)

These Health*e*Vet-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).

2.2 Installation Procedures—Outline

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

I. Pre-Installation Instructions.

II. VistA M Server Installation Instructions:

   A. Kernel Patches

   B. RPC Broker Patches

III. Application Server Installation Instructions:

   A. Security Service Provider Interface (SSPI)

      1. Installation

      2. Configuration

   B. Configure Standard Data Services (SDS) Tables

   C. Edit the KAAJEE Configuration File

   D. Configure log4j for all J2EE-based application log entries

NOTE: Kernel is the designated custodial software application for KAAJEE; however, KAAJEE comprises multiple patches and software releases from several Health*e*Vet-VistA applications.

REF: For the specific VistA M Server software patches required for the implementation of KAAJEE, please refer to Table 2-2 in this chapter.

2.3 Distribution Files

Confirm the following KAAJEE and related software and documentation files:

File/Item Name	Type	Description
KAAJEE_1_0_1_README.TXT	ASCII	**Readme File**. Use this file for any pre-installation instructions, last minute changes, new instructions, and additional information to supplement the manuals. Read all sections of this file prior to following the installation instructions in the *Kernel Authentication & Authorization for J2EE (KAAJEE) Installation Guide* (i.e., KAAJEE_1_0_1_INSTALLGUIDE.PDF).
KAAJEE_1_0_1_INSTALLGUIDE.PDF	Binary	**Installation Guide**. Use this manual in conjunction with the Readme text file (i.e., KAAJEE_1_0_1_README.TXT) to install the required software.
KAAJEE_1_0_1_DEPLOYGUIDE.PDF	Binary	**Deployment Guide**. This manual contains the User Guide, Developer guide, and Systems Management Guide for KAAJEE.
KAAJEE-related VistA M Server Patches (See Table 2-2)	ASCII	**KIDS Distributions (Patches)/Software Releases**. Software patches for installation on the VistA M Server: • Kernel—Options, RPCs, Routines, & Files • RPC Broker—Options, RPCs, Routines, & Files
kaajee_security_provider_1.0.0.010.zip	Binary	**Security Service Provider Interface (SSPI) Software.** The KAAJEE SSPI software download Zip file for installation on the application server.
kaajee_security_provider_1.0.0.010.zip.MD5	Binary	**Security Service Provider Interface (SSPI) Software Checksum.** The MD5 checksum value for the KAAJEE SSPI software download Zip file.

Table 2-1. Distribution files—KAAJEE server files

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

- Preferred Method REDACTED

  This method transmits the files from the first available FTP server.
- Albany OIFO REDACTED
- Hines OIFO REDACTED
- Salt Lake City OIFO REDACTED

REF: For more information on MD5 files, please refer to the "Checksums: MD5—Application Server Java-related Software" topic in this chapter.

2.4 Checksums: MD5—Application Server Java-related Software

In order to determine the validity and integrity for all software deployable artifacts (e.g., ear, war, and jar files) and/or distribution artifacts (e.g., zip files), KAAJEE includes MD5 files containing the checksums for those jar and zip file artifacts that are created by and exported with the KAAJEE software. In this way, all stakeholders (project development teams, sites, etc.) can be assured of the current version of the KAAJEE deliverables that are being tested/deployed.

REF: For more information on the MD5 checksum and algorithm, please refer to the following Website:

http://www.fourmilab.ch/md5/

As of the release of the KAAJEE SSPI 1.0.0.010 software, the following MD5 files are included as part of the KAAJEE software distribution:

- Zip Distribution Artifact—kaajee_security_provider_1.0.0.010.zip.MD5
- Jar Deployment Artifact—wlKaajeeSecurityProviders-1.0.0.010.jar.MD5

To obtain and compare MD5 checksum values do the following:

- Linux—Run openssl (distributed with Linux).
- Windows—Run the md5.exe software (available for download on the http://www.fourmilab.ch/md5/ Website).

2.4.1 Zip Distribution Artifact

The following examples show you how to return the MD5 checksums and compare the MD5 values of the KAAJEE zip distribution artifact (i.e., kaajee_security_provider_1.0.0.010.zip) with the MD5 file (i.e., kaajee_security_provider_1.0.0.010.zip.MD5).

In these examples, the MD5 checksum values are as follows:

- **e0a6091bd17b7593441bf9b13aae9423**—kaajee_security_provider_1.0.0.010.zip (distribution artifact)
- **e0a6091bd17b7593441bf9b13aae9423**—kaajee_security_provider_1.0.0.010.zip.MD5 (MD5 file)

When the checksums match via a visual comparison, users can be assured that the current version of the deliverables that are being tested/deployed are correct.

BEGIN: Linux Instructions

1. **(Linux) Open a Terminal**

   Open any X-Windows terminal server software (e.g., Virtual Network Computing [VNC]) or a secure character-based terminal emulator (e.g., Putty) to access Linux.

   Navigate to the staging folder where you loaded the KAAJEE SSPI Zip distribution file.

2. **(Linux) Run openssl**

   Run openssl, as shown below:

```
[REDACTED_1.0.0.010]$ openssl md5 kaajee_security_provider_1.0.0.010.zip
<Enter>
MD5(kaajee_security_provider_1.0.0.010.zip)=
e0a6091bd17b7593441bf9b13aae9423
[REDACTED_1.0.0.010]$ cat kaajee_security_provider_1.0.0.010.zip.MD5 <Enter>
e0a6091bd17b7593441bf9b13aae9423
```

Figure 2.4-1. Linux—Sample MD5 checksum comparison for a Zip distribution artifact

END: Linux Instructions

▶▶▌ Linux users, skip to 2.4.2.

BEGIN: Microsoft Windows Instructions

1. **(Windows) Open a Dos/Command Shell**

   Log onto Windows and open a DOS shell.

   Navigate to the staging folder where you loaded the KAAJEE SSPI Zip distribution file.

2. **(Windows) Run MD5.exe**

   Run MD5.exe, as shown below:

```
C:\TEMP\TEMP_KAAJEE_SSPIs\kaajee_security_provider_1.0.0.010\kaajee_security
_provider>md5 -l kaajee_security_provider_1.0.0.010.zip <Enter>
e0a6091bd17b7593441bf9b13aae9423  kaajee_security_provider_1.0.0.010.zip

C:\TEMP\TEMP_KAAJEE_SSPIs\kaajee_security_provider_1.0.0.010\kaajee_security
_provider>type kaajee_security_provider_1.0.0.010.zip.MD5 <Enter>
e0a6091bd17b7593441bf9b13aae9423
```

Figure 2.4-2. Windows—Sample MD5 checksum comparison for a Zip distribution artifact

END: Microsoft Windows Instructions

2.4.2 Jar Distribution Artifact

The following examples show you how to return the MD5 checksums and compare the MD5 values of the KAAJEE jar deployment artifact (i.e., wlKaajeeSecurityProviders-1.0.0.010.jar) with the MD5 file (i.e., wlKaajeeSecurityProviders-1.0.0.010.jar.MD5).

In these examples, the MD5 checksum values are as follows:

- **2de1649fee13294dc6faf8ab54f3e0f1**—wlKaajeeSecurityProviders-1.0.0.010.jar (deployment artifact)
- **2de1649fee13294dc6faf8ab54f3e0f1**—wlKaajeeSecurityProviders-1.0.0.010.jar.MD5 (MD5 file)

When the checksums match via a visual comparison, users can be assured that the current version of the deliverables that are being tested/deployed are correct.

BEGIN: Linux Instructions

1. **(Linux) Open a Terminal**

   Open any X-Windows terminal server software (e.g., Virtual Network Computing [VNC]) or a secure character-based terminal emulator (e.g., Putty) to access Linux.

   Navigate to the staging folder where you decompressed the KAAJEE SSPI software.

2. **(Linux) Run openssl**

   Run openssl, as shown below:

```
[REDACTED]$ openssl md5 wlKaajeeSecurityProviders-1.0.0.010.jar <Enter>
MD5(wlKaajeeSecurityProviders-1.0.0.010.jar)=
2de1649fee13294dc6faf8ab54f3e0f1
[REDACTED]$ cat wlKaajeeSecurityProviders-1.0.0.010.jar.MD5 <Enter>
2de1649fee13294dc6faf8ab54f3e0f1
```

Figure 2.4-3. Linux—Sample MD5 checksum comparison for a Jar deployment artifact

END: Linux Instructions

⏭ Linux users, skip to 2.5.

BEGIN: Microsoft Windows Instructions

1. **(Windows) Open a Dos/Command Shell**

   Log onto Windows and open a DOS shell.

   Navigate to the staging folder where you unzipped the KAAJEE SSPI software.

2. **(Windows) Run MD5.exe**

   Run MD5.exe, as shown below:

```
C:\TEMP\TEMP_KAAJEE_SSPIs\kaajee_security_provider_1.0.0.010\kaajee_security
_provider>md5 -l wlKaajeeSecurityProviders-1.0.0.010.jar <Enter>
2de1649fee13294dc6faf8ab54f3e0f1  wlKaajeeSecurityProviders-1.0.0.010.jar

C:\TEMP\TEMP_KAAJEE_SSPIs\kaajee_security_provider_1.0.0.010\kaajee_security
_provider>type wlKaajeeSecurityProviders-1.0.0.010.jar.MD5 <Enter>
2de1649fee13294dc6faf8ab54f3e0f1
```

Figure 2.4-4. Windows—Sample MD5 checksum comparison for a Jar deployment artifact

END: Microsoft Windows Instructions

2.5 Dependencies—VistA M Server Patches

Kernel is the designated custodial software application of KAAJEE-related software. However, KAAJEE comprises/depends on multiple software patches released by several VistA M Server applications (listed by software name):

Software	Version	Patch Release	Subject/Description
Kernel	8.0	XU*8.0*265	3 Strikes and You Are Out—This patch was released with KAAJEE 1.0.0.019. It enhanced security by providing Internet Protocol (IP) address locking functionality (terminal servers are uniquely handled). Also provides special locking security for individual users. **NOTE:** This patch is required for Kernel Patch XU*8.0*337.
		XU*8.0*329	Used with Web-Based Kernel Authentication Tool—This patch was released with KAAJEE 1.0.0.019. It contained the following: • One "B"-type option, XUS KAAJEE WEB LOGON. This option contains references to the following RPCs in its "RPC" multiple: – XUS ALLKEYS – XUS KAAJEE GET USER INFO – XUS KAAJEE LOGOUT This option has no effect on those RPCs as such; however, having this option assigned allows KAAJEE to call these RPCs on behalf of the end-user. • One "Menu"-type option, XUCOMMAND. This option is only used to link XUS KAAJEE WEB LOGON to XUCOMMAND. As all authenticated users have access to XUCOMMAND, this linkage enables all users to have access to all RPCs listed under the XUS KAAJEE WEB LOGON "B"-type option. • Two RPCs: – XUS KAAJEE GET USER INFO – XUS KAAJEE LOGOUT • One Routine: XUSKAAJ
		XU*8.0*337	CCOW SSO/UC Support—This patch was released with KAAJEE 1.0.0.019. It updated Kernel Authentication and Authorization routines in order to enable SSO/UC and provide the VPID for KAAJEE. It also distributes the XUS ALLKEYS RPC that is required by KAAJEE.

Software	Version	Patch Release	Subject/Description
			NOTE: Kernel (i.e., Kernel Patch XU*8.0*337) is the designated custodial package of the SSO/UC-related software. This patch is dependent on Kernel Patch XU*8.0*265 because Patches XU*8.0*265 and 337 are modifying the same Kernel Authentication and Authorization routines.
		XU*8.0*361	Proxy Application User for Re-hosting Effort—This patch was released with KAAJEE 1.0.0.019. FatKAAT (not yet released) uses the Application Proxy user provided with this patch. **NOTE:** This patch is not directly required by KAAJEE; however, since VistALink requires this patch and KAAJEE requires VistALink, this patch is included here.
		XU*8.0*430	Return Value For Administrative Parent (KAAJEE)—This patch returns a caret ("**^**") for the Administrative Parent station number whenever it is *not* defined. KAAJEE converts this caret ("**^**") back to an empty string to eliminate a code problem.
		XU*8.0*451	KAAJEE Login Page—Removal of Refresh Button. This patch is currently was released with KAAJEE 1.0.1.xxx. This patch provides the following functionality or bug fixes: • Enhanced Login Functionality: – Removed Refresh button from KAAJEE login page. – Added JavaScript code for client-side sorting of Institutions. – Provided Access code ; Verify code capability in one line. – Added support for parameter passing of Default Institution and Institution sorting preferences. This addresses the issues of persistent cookies when using Thin Clients and Terminal Servers. – Made the KAAJEE Login Web page more Section 508 friendlier. • Added Sample Web Application—Provide KAAJEE Sample Web Application.

Software	Version	Patch Release	Subject/Description
			• Updated Software Version Support: – Compiled and tested KAAJEE against SDS 13.0. – Compiled and tested KAAJEE against VistALink 1.5.1.002. • Bug Fixes: – Fixed issue with KAAJEE login not updating LAST SIGN-ON DATE/TIME field (#202) in the NEW PERSON file (#200). – Fixed Response already committed error—The code that was fixed was associated with processing the persistent cookie information on the Application Server. This fix should also fix the extra M process that was created.
RPC Broker	1.1	XWB*1.1*35	NON-callback Server—This patch was released with KAAJEE 1.0.0.019. It provided local sites with the ability to control the range of ports used in connecting to joint and/or contracting facilities, useful behind firewalls. This patch contains the following: • Modified XWB LISTENER STARTER option. • Added a new XWB LISTENER STOP ALL option. • Modified RPC BROKER SITE PARAMETERS file (#8994.1). • Modified XWB LISTENER EDIT template. • New entry added to the PARAMETER DEFINITION file (#8989.51). • Modified/New routines.

Table 2-2. Dependencies—VistA M Server patches

REF: For specific VistA M Server patch details, please refer to the Patch Module on FORUM.

NOTE: This table only includes VistA M Server software patches required for KAAJEE; it does *not* list Commercial-Off-The-Shelf (COTS) software or other Health*e*Vet-VistA software/patches that are not directly related to KAAJEE.

2.6 Installer/Developer Notes—KAAJEE Software Virgin Installations and Upgrades

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

> ★ **VIRGIN INSTALLATION:** *Virgin 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.

2.7 VistA M Server Environment

NOTE: The information in this topic is directed at the Information Resource Management (IRM) personnel located at a site.

2.7.1 Server Requirements

The following minimum software tools network configuration are required on the VistA M Server running KAAJEE-based Web applications:

Minimum Software/Configuration	Description
Operating System Software	One of the following operating systems: • Digital Standard M (DSM) V6.3-031 for OpenVMS AXP or greater • InterSystems Caché **NOTE:** The VistA M Server need not be an NT system.
Fully Patched M Accounts	You should have both a development Test account and a Production account for KAAJEE software. The account(s) *must* contain the *fully* patched versions of the following software: • Kernel 8.0 • Kernel Toolkit 7.3 • RPC Broker 1.1 • VA FileMan 22.0 • VistALink 1.5 **NOTE:** Kernel is the designated custodial software application for KAAJEE. However, KAAJEE comprises multiple patches and software releases from several Health**e**Vet-VistA applications. **REF:** For the specific software/patches required for the implementation of KAAJEE, please refer to Table 2-2 in this chapter.
Network Communications Software **REF:** For more information on telecommunications support, please visit the VHA Communication Services Office (CSO) Home Page: http://vaww.va.gov/cso/	The VistA M Server needs to have TCP/IP running.

Table 2-3. VistA M Server minimum software/network tools/utilities required for KAAJEE

2.7.2 Site Configuration

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.
- Health***e***Vet-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].

2.7.2.1 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 3.0 (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 3.0 (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 Chapter 4, "Application Server Installation Instructions," 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.

2.7.2.2 Validate Institution Associations

KAAJEE uses the Standard Data Services (SDS) tables 3.0 (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 Website:

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

2.8 Application Server Environment Requirements

NOTE: The information in this topic is directed at the Enterprise Management Center (EMC) personnel responsible for maintaining the application servers.

The following minimum software tools are required for the application server running KAAJEE-based Web applications:

Minimum Software/Configuration	Description
Operating System Software	One of the following operating systems: - Linux (i.e., Red Hat Enterprise ES 3.0) - Microsoft Windows XP or 2000
Application Sever Software	In order to develop, test, and run Web-based applications that incorporate KAAJEE functionality, the developer, IRM, or Application Server System Manager (e.g., EMC) *must* install or have running an application server. Currently, KAAJEE supports the WebLogic V 8.1 (SP4 or higher) application server. **NOTE:** This manual assumes that the WebLogic Application Server platform is already installed and running. **REF:** For more information on the WebLogic Application Server, please visit the BEA Home Page: http://www.bea.com/
SSPI Software	In order to develop, test, and run Web-based applications that incorporate KAAJEE functionality, the developer, IRM, or Application Server System Manager (e.g., EMC) *must* install and configure the Security Service Provider Interface (SSPI) 1.0.0.010. **REF:** Installation and configuration instructions are included in the Chapter 4, "Application Server Installation Instructions," in this manual.

Minimum Software/Configuration	Description
VistALink Software	In order to develop, test, and run Web-based applications that incorporate KAAJEE functionality, the developer, IRM, or Application Server System Manager (e.g., EMC) *must* install the VistALink software on the application server. KAAJEE uses the J2EE-enabled version of VistALink and requires the following: • VistALink 1.5 Software (fully patched)—The software is installed in the same instance(s) in which the developer's Web application is installed. • Connectors should be configured for all of the VistA M Servers to which the Web-based application needs to connect via VistALink. **NOTE:** This manual assumes that the VistALink software is already installed and running on the application server.
Network Communications Software **REF:** For more information on telecommunications support, please visit the VHA Communication Services Office (CSO) Home Page: http://vaww.va.gov/cso/	The application server needs to have TCP/IP running.

Table 2-4. Application server minimum software/network tools/utilities required for KAAJEE

3 VistA M Server Installation Instructions

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

REF: For VistA M Server platform requirements, please refer to the "Server Requirements" topic in the "VistA M Server Environment" topic in Chapter 2, "Pre-Installation Instructions," in this manual.

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

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

File Name	Type	Description
KAAJEE_1_0_1_README.TXT	ASCII	**Readme Text File.** This file provides any pre-installation instructions, last minute changes, new instructions, and additional information to supplement the manuals. Read all sections of this file *prior* to following the installation instructions in the *Kernel Authentication & Authorization for J2EE (KAAJEE) Installation Guide* (i.e., KAAJEE_1_0_1_INSTALLGUIDE.PDF).
KAAJEE_1_0_1_INSTALLGUIDE.PDF	Binary	**Installation Guide.** Use in conjunction with the Readme text file (i.e., KAAJEE_1_0_1_README.TXT).
XU*8.0*265	ASCII	**Kernel Patch XU*8.0*265.** KIDS build for Kernel Patch XU*8.0*265 (released with KAAJEE 1.0.0.019 on 12/12/05, required for Patch XU*8.0*337, see Table 2-2 for patch details). Follow normal procedures to obtain and install this released patch (see FORUM).
XU*8.0*329	ASCII	**Kernel Patch XU*8.0*329.** KIDS build for Kernel Patch XU*8.0*329 (released with KAAJEE 1.0.0.019 on 06/13/06, see for patch details). Follow normal procedures to obtain and install this patch (see FORUM).
XU*8.0*337	ASCII	**Kernel Patch XU*8.0*337.** KIDS build for Kernel Patch XU*8.0*337 (released with KAAJEE 1.0.0.019 on 12/22/05, see Table 2-2 for patch details). Follow normal procedures to obtain and install this released patch (see FORUM).
XU*8.0*361	ASCII	**Kernel Patch XU*8.0*361.** KIDS build for Kernel Patch XU*8.0*361 (released with KAAJEE 1.0.0.019 on 01/31/06, see Table 2-2 for patch details). Follow normal procedures to obtain and install this patch (see FORUM).
XU*8.0*430	ASCII	**Kernel Patch XU*8.0*430.** KIDS build for Kernel Patch XU*8.0*430 (released with KAAJEE 1.0.0.019 on 10/26/06,

File Name	Type	Description
		see Table 2-2 for patch details). Follow normal procedures to obtain and install this patch (see FORUM).
XU*8.0*451	ASCII	**Kernel Patch XU*8.0*451.** KIDS build for Kernel Patch XU*8.0*451 (released with KAAJEE 1.0.1.xxx, see Table 2-2 for patch details). Follow normal procedures to obtain and install this released patch (see FORUM).
XWB*1.1*35	ASCII	**RPC Broker Patch XWB*1.1*35.** KIDS build for RPC Broker Patch XWB*1.1*35 (released with KAAJEE 1.0.0.019 on 01/20/05, see Table 2-2 for patch details). Follow normal procedures to obtain and install this released patch (see FORUM).

Table 3-1. Distribution files—KAAJEE-related VistA M Server files

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

- REDACTED

3.2 Retrieve VistA M Server Patches *(required)*

At the time of publication of this manual, several VistA M Server-side patches are required for KAAJEE installation (see Table 3-1). You should have these patches readily available so that you can apply them later in the installation process. Obtain all released KAAJEE-related VistA M Server-side patches from the Patch module on FORUM or through normal procedures.

NOTE: Kernel is the designated custodial software application for KAAJEE; however, KAAJEE comprises multiple patches and software releases from several Health***e***Vet-VistA applications.

REF: For the specific VistA M Server software patches required for the implementation of KAAJEE, please refer to Table 2-2 in Chapter 2, "Pre-Installation Instructions," in this manual.

3.3 Do Not Run any KAAJEE-based Software during the Installation *(required)*

No Health***e***Vet-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.

3.4.1 Verify Host File Server (HFS) Device in the DEVICE File (#3.5)

Verify that you have a Host File Server (HFS) device in the DEVICE file (#3.5) named **"HFS"**. If you have performed KIDS installations on the VistA M Server before, you probably already have an appropriate HFS device set up. If you don't have an entry for this device, you *must* create one.

REF: For information on how to create an HFS device, please refer to the "Host Files" chapter in the *Kernel Systems Management Guide*.

3.4.2 Verify Null Device in the DEVICE File (#3.5)

Verify that you have a Null device in the DEVICE file (#3.5) named "NULL" (or whose mnemonic is named "NULL").

You can have other devices with similar names, but one device is needed whose name or mnemonic is "NULL." The subtype should be a "P-" subtype (e.g., P-OTHER), the margin should be a minimum of 80, and the page length should be a minimum of 60. Sample setups:

Caché or DSM for OpenVMS Null Device Setup Example

```
NAME: NULL                          $I: _NLA0:
  ASK DEVICE: NO                    ASK PARAMETERS: NO
  SIGN-ON/SYSTEM DEVICE: NO         LOCATION OF TERMINAL: Bit Bucket
  SUBTYPE: P-OTHER                  TYPE: TERMINAL
```

Caché/NT Null Device Setup Example

```
NAME: NULL                          $I: //./nul
  ASK DEVICE: NO                    ASK PARAMETERS: NO
  SIGN-ON/SYSTEM DEVICE: NO         LOCATION OF TERMINAL: BIT BUCKIT
  SUBTYPE: P-OTHER                  TYPE: TERMINAL
```

P-OTHER Terminal Type Setup Example

```
NAME: P-OTHER                       RIGHT MARGIN: 132
  FORM FEED: #                      PAGE LENGTH: 64
  BACK SPACE: $C(8)                 DESCRIPTION: General prntr (132)
```

3.5 Install KAAJEE-related VistA M Server Patches *(required)*

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.

The KAAJEE-related VistA M Server patches are listed in Table 2-2. All VistA M Server patches are distributed in Kernel 8.0 KIDS format. Follow the normal procedures to obtain released patches.

REF: For more information on these patches, please refer to Table 2-2 in this manual or the Patch Module on FORUM.

3.5.1 Load/Install the KIDS Builds

Using KIDS, load and install the KAAJEE-related VistA M Server patches on all VistA M systems to which any Web-based application will be connecting (i.e., VistA M Server Test and Production accounts).

Follow the instructions under the "Installation Instructions" section in the patch description in order to install each patch.

REF: For more information on KIDS, please refer to the KIDS section in the *Kernel Systems Management Guide* located on the VDL at the following Website:

http://www.va.gov/vdl/application.asp?appID=10

3.5.2 Verify the XUS KAAJEE WEB LOGON Menu Option is Linked to XUCOMMAND

After installing Kernel Patch XU*8.0*329, verify that Kernel Patch XU*8.0*329 automatically linked the XUS KAAJEE WEB LOGON menu option to the XUCOMMAND menu. As all authenticated users have access to XUCOMMAND, this linkage enables all users to have access to all RPCs listed under the XUS KAAJEE WEB LOGON "B"-type option.

NOTE: Use the VA FileMan Inquire to File Entries option to verify that XUCOMMAND shows XUS KAAJEE WEB LOGON as a list item.

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

4 Application Server Installation Instructions

The installation instructions in this section are directed at the Enterprise Management Center (EMC) personnel 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 and Microsoft Windows platforms.

REF: For application server platform requirements, please refer to the "Application Server Environment Requirements" topic in Chapter 2, "Pre-Installation Instructions," in this manual.

4.1 Confirm/Obtain Application Server Distribution Files *(recommended)*

The following files are needed to install the Kernel Authentication and Authorization Java (2) Enterprise Edition (KAAJEE) application server software:

File Name	Type	Description
KAAJEE_1_0_1_README.TXT	ASCII	**Readme Text File.** This file provides any pre-installation instructions, last minute changes, new instructions, and additional information to supplement the manuals. Read all sections of this file prior to following the installation instructions in the *Kernel Authentication & Authorization for J2EE (KAAJEE) Installation Guide* (i.e., KAAJEE_1_0_1_INSTALLGUIDE.PDF).
KAAJEE_1_0_1_INSTALLGUIDE.PDF	Binary	**Installation Guide.** Use in conjunction with the Readme text file (i.e., KAAJEE_1_0_1_README.TXT).
kaajee_security_provider_1.0.0.010.zip	Binary	**Security Provider Interface (SSPI) Software.** The KAAJEE SSPI software download Zip file for installation on the application server.
kaajee_security_provider_1.0.0.010.zip.MD5	Binary	**Security Service Provider Interface (SSPI) Software Checksum.** The MD5 checksum value for the KAAJEE SSPI software download Zip file.

Table 4-1. Distribution files—KAAJEE application server files

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

- REDACTED

REF: For more information on MD5 files, please refer to the "Checksums: MD5—Application Server Java-related Software" topic in Chapter 2, "Pre-Installation Instructions," 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:

Directory <Alias> Placeholder	Description (and Document Default Directories)
<BEA_HOME>	The directory where you installed the WebLogic Server 8.1 (SP4 or higher) 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: **Linux:** /u01/app/bea **Windows:** C:\bea
<JAVA_HOME>	The directory where you installed the Java developer software. For the examples in this document, the default home directory is: **Linux:** /usr/java/j2sdk1.4.2_05 **Windows:** C:\java\j2sdk1.4.2_05
<DOMAIN_NAME>	The name of your WebLogic domain. For the examples in this document (Linux/Windows), the directory is: **Linux/Windows:** kaajeewebdomain.
<USER_DOMAIN_HOME>	The directory where your user domain is located. For the examples in this document, the directory is: **Linux:** /u01/app/bea/user_projects/domains/kaajeewebdomain **Windows:** C:\bea\user_projects\domains\kaajeewebdomain
<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: **Linux/Windows:** devKAAJEE1.
<HEV CONFIGURATION FOLDER>	This is the folder placed on the classpath of WebLogic Application Servers, containing the configuration files for all Health**e**Vet-VistA J2EE applications.

Table 4-2. Application server directory <Alias> placeholders (for documentation purposes)

4.2 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:

4.2.1 (Linux: Admin Server) Open a Terminal

Open any X-Windows terminal server software (e.g., Virtual Network Computing [VNC]) or a secure character-based terminal emulator (e.g., Putty) to access the Linux Admin Server.

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

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

Navigate to the following directory:

<BEA_HOME>/weblogic81/common/bin

4.2.3 (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:

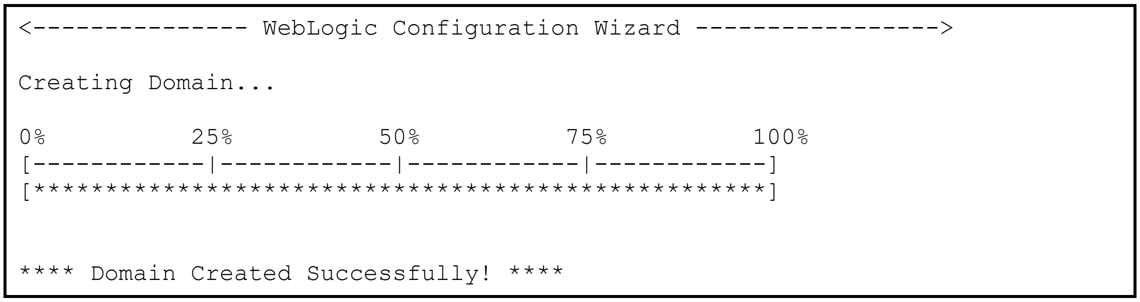
   ./config.sh

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

   If you are using an X-Windows terminal server (e.g., VNC), follow the instructions as shown in Step #4.2.5, "(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 8.1.2.0," 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:



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

Creating Domain...

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


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

Figure 4.2-1. Linux Admin Server—Successful domain creation message

END: Linux Instructions

Linux users, skip to 4.3.

BEGIN: Microsoft Windows Instructions

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

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

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

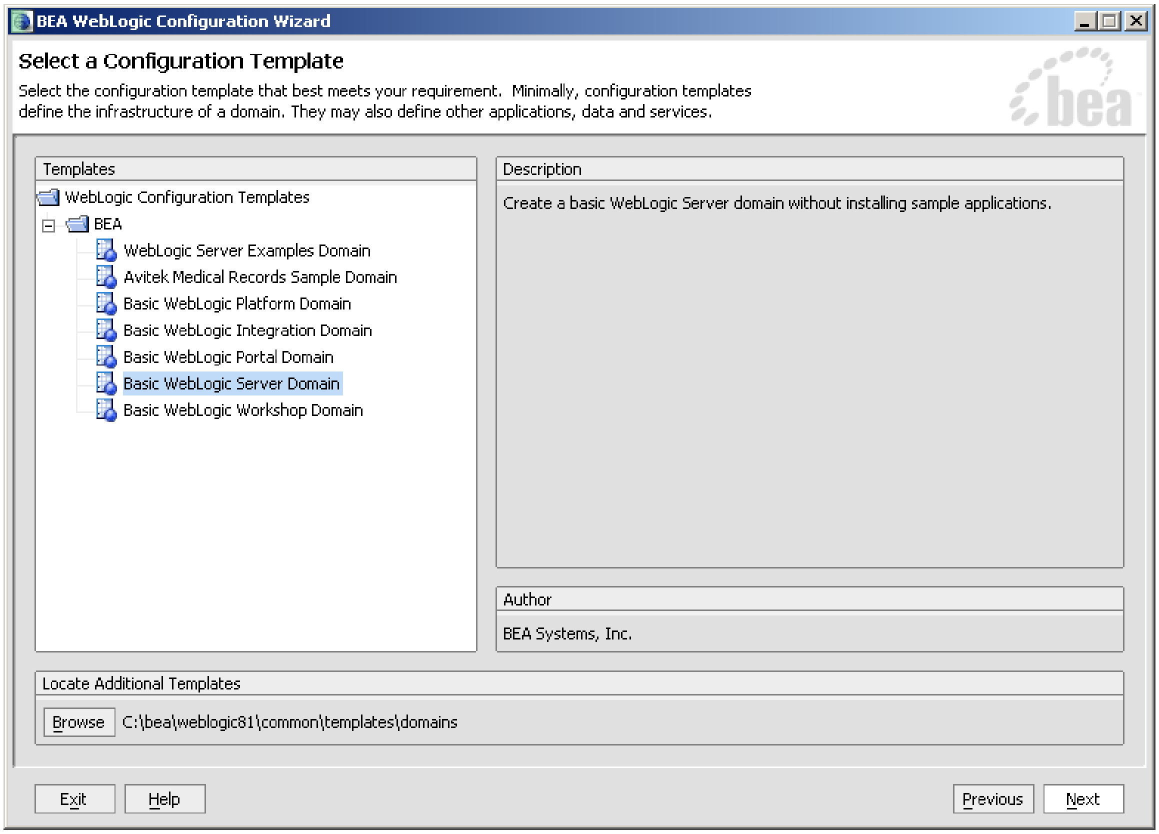
Start > Programs > WebLogic Platform 8.1 > Configuration Wizard

4.2.5 (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. Highlight "Basic WebLogic Server Domain" in the list of Templates, as shown below:



BEA WebLogic Configuration Wizard

Select a Configuration Template

Select the configuration template that best meets your requirement. Minimally, configuration templates define the infrastructure of a domain. They may also define other applications, data and services.

Templates

- WebLogic Configuration Templates
  - BEA
    - WebLogic Server Examples Domain
    - Avitek Medical Records Sample Domain
    - Basic WebLogic Platform Domain
    - Basic WebLogic Integration Domain
    - Basic WebLogic Portal Domain
    - Basic WebLogic Server Domain
    - Basic WebLogic Workshop Domain

Description

Create a basic WebLogic Server domain without installing sample applications.

Author

BEA Systems, Inc.

Locate Additional Templates

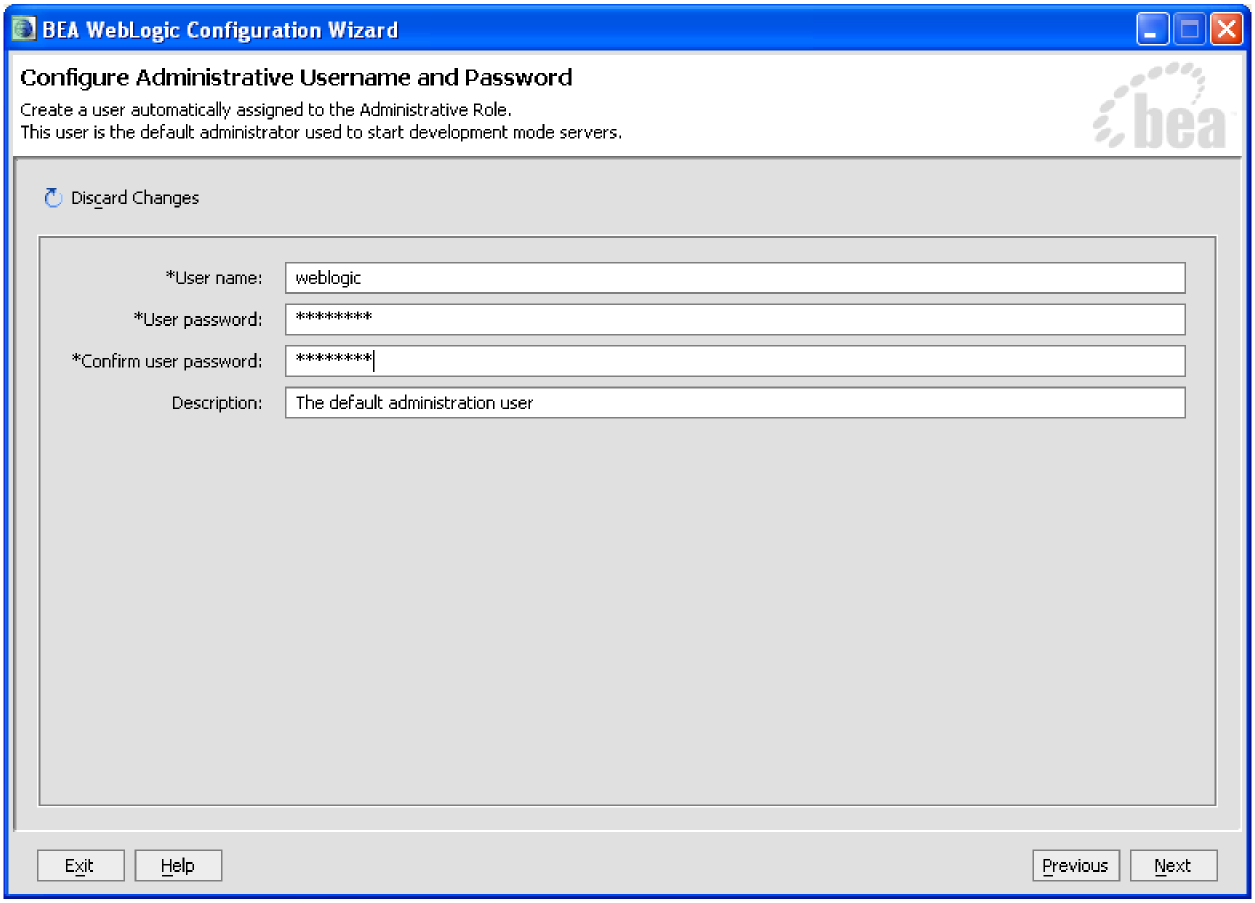
C:\bea\weblogic81\common\templates\domains

Figure 4.2-2. Windows Admin Server—WebLogic Configuration Wizard: Select a Configuration Template screen

Click **Next**.

3. Choose "Express" (default) on the next screen and click **Next**.

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



Configure Administrative Username and Password

Create a user automatically assigned to the Administrative Role.
This user is the default administrator used to start development mode servers.

*User name: weblogic

*User password: ********

*Confirm user password: ********

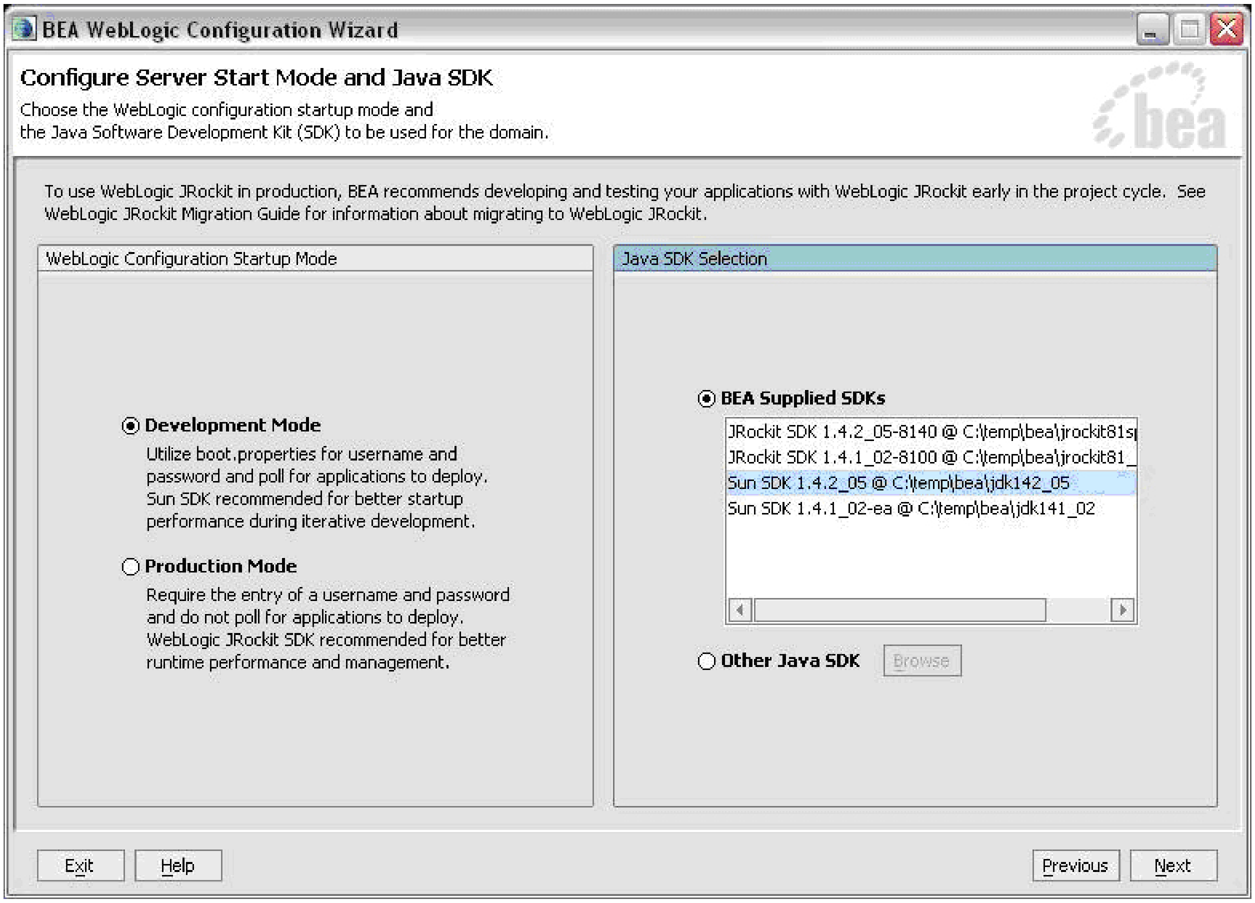
Description: The default administration user

Figure 4.2-3. Windows Admin Server—WebLogic Configuration Wizard: Configure Administrative Username and Password screen

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

Click **Next**.

5. Choose the Java SDK, as shown below:



BEA WebLogic Configuration Wizard

Configure Server Start Mode and Java SDK

Choose the WebLogic configuration startup mode and
the Java Software Development Kit (SDK) to be used for the domain.

To use WebLogic JRockit in production, BEA recommends developing and testing your applications with WebLogic JRockit early in the project cycle. See WebLogic JRockit Migration Guide for information about migrating to WebLogic JRockit.

WebLogic Configuration Startup Mode

● Development Mode

Utilize boot.properties for username and password and poll for applications to deploy. Sun SDK recommended for better startup performance during iterative development.

○ Production Mode

Require the entry of a username and password and do not poll for applications to deploy. WebLogic JRockit SDK recommended for better runtime performance and management.

Java SDK Selection

● BEA Supplied SDKs

JRockit SDK 1.4.2_05-8140 @ C:\temp\bea\jrockit81s|
JRockit SDK 1.4.1_02-8100 @ C:\temp\bea\jrockit81_
Sun SDK 1.4.2_05 @ C:\temp\bea\jdk142_05
Sun SDK 1.4.1_02-ea @ C:\temp\bea\jdk141_02

○ Other Java SDK

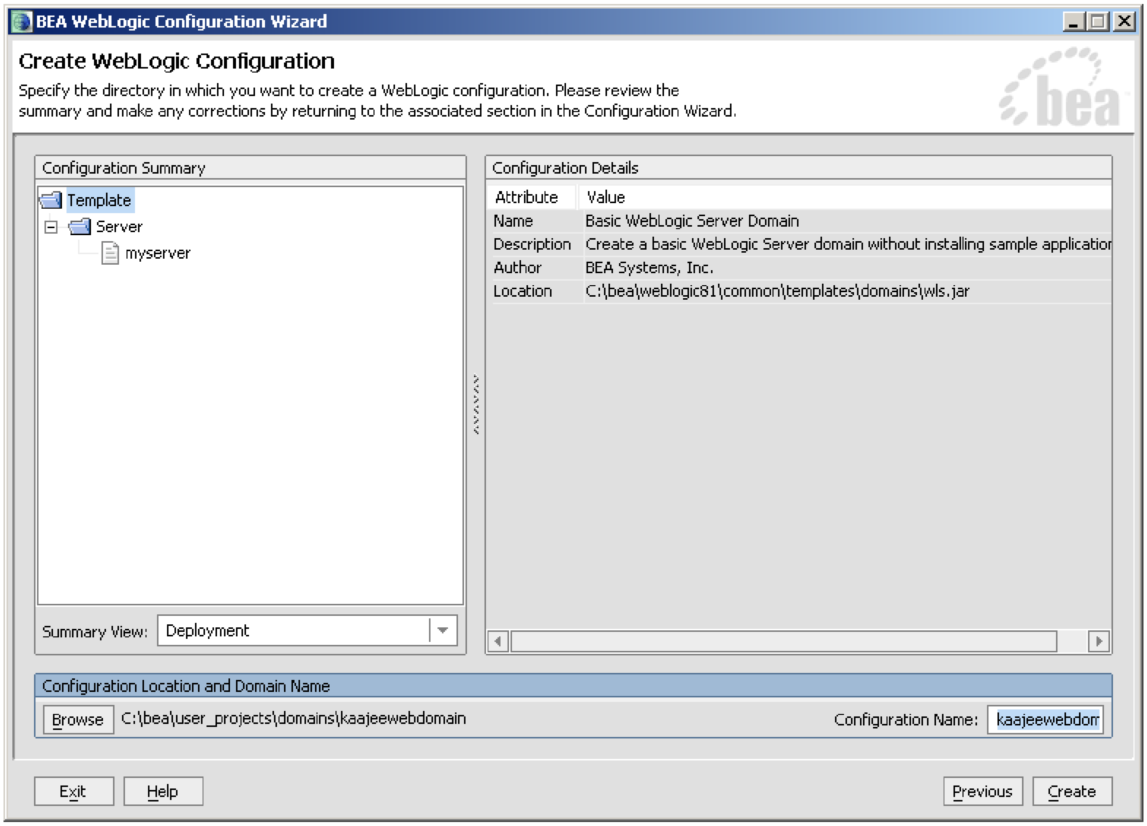
Figure 4.2-4. Windows Admin Server—WebLogic Configuration Wizard: Configure Server Start Mode and Java SDK screen

In this example, the user chose "**Sun SDK 1.4.2_05**" Java SDK from the list of BEA Supplied SDKs list.

Click **Next**.

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

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



BEA WebLogic Configuration Wizard

Create WebLogic Configuration

Specify the directory in which you want to create a WebLogic configuration. Please review the summary and make any corrections by returning to the associated section in the Configuration Wizard.

Configuration Summary

- Template
- Server
  - myserver

Summary View: Deployment

Configuration Details

Attribute	Value
Name	Basic WebLogic Server Domain
Description	Create a basic WebLogic Server domain without installing sample applicatior
Author	BEA Systems, Inc.
Location	C:\bea\weblogic81\common\templates\domains\wls.jar

Configuration Location and Domain Name

C:\bea\user_projects\domains\kaajeewebdomain

Configuration Name: kaajeewebdom

Figure 4.2-5. Windows Admin Server—WebLogic Configuration Wizard: Create WebLogic configuration

Here you should enter a domain name (e.g., kaajeewebdomain) in the text box after the "Configuration Name:' prompt.

Click **Create**.

7. Check the "Start Server and Process" check box and click **Done**. This opens a DOS box on the workstation.
8. Open an Internet Browser (e.g., Microsoft Internet Explorer) 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 in Step #4.

END: Microsoft Windows Instructions

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

The developer, IRM, or Application Server System Manager (e.g., EMC) *must* install and configure the Security Service Provider Interface (SSPI) software on the WebLogic 8.1 (SP4 or higher) Application Server in order to develop, test, and run Web-based applications that are KAAJEE-enabled.

4.3.1 Undeploy SSPI Software

VIRGIN INSTALLATIONS: Skip this step and proceed to Step #4.3.2, "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.

4.3.1.1 Delete kaajeeManageableAuthenticator

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

Security > Realms > Providers > Realms > myrealm > Providers > Authentication

Delete kaajeeManageableAuthenticator. Confirm the delete when prompted. Click **Continue** when ready.

4.3.1.2 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 **Apply**.

4.3.1.3 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

4.3.1.4 (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>**/

For example:

/u01/app/bea/user_project/domains/kaajeewebdomain/

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

-Dweblogic.alternateTypesDirectory=${sspidir}

Save and close the file.

END: Linux Instructions

⏭ Linux users, skip to 4.3.1.6.

BEGIN: Microsoft Windows Instructions

4.3.1.5 (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>**\

For example:

C:\bea\user_project\domains\kaajeewebdomain\

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

-Dweblogic.alternateTypesDirectory=%sspidir%

Save and close the file.

END: Microsoft Windows Instructions

4.3.1.6 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.

4.3.1.7 Verify Removal of the kaajeeManageableAuthenticator

Users should navigate to the following directory:

Security > Realms > Providers > Realms > myrealm > Providers > Authentication

Verify that the kaajeeManageableAuthenticator is no longer listed.

4.3.1.8 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.

4.3.1.9 Move and Back Up the wlKaajeeSecurityProviders-1.0.0.010.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-1.0.0.010.jar file to a backup directory.

4.3.1.10 KAAJEE SSPI Successfully Undeployed

At this point, users are now ready to deploy the latest version of the KAAJEE SSPIs, proceed to Step #4.3.2, "Deploy SSPI Software."

4.3.2 Deploy SSPI Software

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

4.3.2.1 Download/Obtain SSPI Software

Download the kaajee_security_provider_1.0.0.010.zip software from the EPS anonymous directories.

4.3.2.2 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>

4.3.2.3 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_1.0.0.010.zip file in the **<SSPI_STAGING_FOLDER>** directory, you will see the following contents/folder structure:

Folder/Structure	Description
..\kaajee_security_provider	This folder is the KAAJEE SSPI <root> level. This folder contains the following files: • build.xml—KAAJEE SSPI Ant build script. • readme.txt—KAAJEE SSPI documentation (manual), which includes an introduction, change history, any special installation instructions, and any known issues/limitations. • wlKaajeeSecurityProviders-1.0.0.010.jar—The KAAJEE SSPI software deployment jar file. • wlKaajeeSecurityProviders-1.0.0.010.jar.MD5—The MD5 checksum value for the KAAJEE SSPI software deployment jar file.
..\common_pools_jars	This folder contains the following files: • commons-collections-3.1.jar • commons-dbcp-1.2.1.jar • commons-pool-1.2.jar
..\props	This properties folder contains the following files: • KaajeeDatabase.properties • KaajeeManageableAuthenticator.xml

Folder/Structure	Description
..\sql	This folder contains the SQL scripts for the following databases: • CacheTables.sql • OracleTables.sql
..\src	This folder contains the KAAJEE SSPI Java source code (i.e., the application server software).

Table 4-3. kaajee-1.0.1.xxx—KAAJEE folder structure

REF: For more information on MD5 files, please refer to the "Checksums: MD5—Application Server Java-related Software" topic in Chapter 2, "Pre-Installation Instructions," in this manual.

BEGIN: Linux Instructions

Use the "jar" command to decompress the kaajee_security_provider_1.0.0.010.zip distribution file in the **<SSPI_STAGING_FOLDER>** staging directory:

```
<SSPI_STAGING_FOLDER> jar -xvf kaajee_security_provider_1.0.0.010.zip
```

END: Linux Instructions

Linux users, skip to 4.3.2.4.

BEGIN: Microsoft Windows Instructions

Unzip the kaajee_security_provider_1.0.0.010.zip distribution file in the **<SSPI_STAGING_FOLDER>** staging directory.

END: Microsoft Windows Instructions

4.3.2.4 Configure the SSPI Software on the Application Server

Configure the SSPI software on the WebLogic 8.1 (SP4 or higher) Application Server, 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

4.3.2.4.1 (Linux: Admin Server) Modify the startWebLogic.sh File

For Linux, the startWebLogic.sh 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>**/

For example:

/u01/app/bea/user_project/domains/kaajeewebdomain/

NOTE: In the examples that follow, some of the directory paths are represented by their <**Alias**>, as described in Table 4-2. 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.

4.3.2.4.1.1 Add Apache Connection Pool Jar Files to the SSPI Classpath

The following Apache connection pool jar files *must* be added to the SSPI classpath:

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

These files are located in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars/

Specifically, add the following lines after the line JAVA_VENDOR="Sun" line in the startWebLogic.sh file:

```
#REM Custom SSPI classpath:
ApacheCoonPool="<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars"
commonpool="${ApacheCoonPool}/commons-pool-1.2.jar"
commondbcp="${ApacheCoonPool}/commons-dbcp-1.2.1.jar"
commoncollection="${ApacheCoonPool}/commons-collections-3.1.jar"
propertiesdir="<SSPI_STAGING_FOLDER>/kaajee_security_provider/props"
sspidir="<SSPI_STAGING_FOLDER>/kaajee_security_provider"
```

Figure 4.3-1. Linux Admin Server—KAAJEE SSPI classpath additions to the startWebLogic.sh file (*Generic* example *with* <Alias> placeholders)

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

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

```
#REM Custom SSPI classpath:
ApacheCoonPool="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_
provider/common_pool_jars"
commonpool="${ApacheCoonPool}/commons-pool-1.2.jar"
commondbcp="${ApacheCoonPool}/commons-dbcp-1.2.1.jar"
commoncollection="${ApacheCoonPool}/commons-collections-3.1.jar"
propertiesdir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_p
rovider/props"
sspidir="/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provide
r"
```

Figure 4.3-2. Linux Admin Server—KAAJEE SSPI classpath additions to the startWebLogic.sh file (*Actual* example *without* <Alias> placeholder)

4.3.2.4.1.2 Add Variables to the SSPI Classpath

Add the following variables to the SSPI classpath:

- 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.)
- 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.

Specifically, modify the line shown below that follows the argument to set the SERVER_NAME in the startWebLogic.sh file:

From:

```
CLASSPATH="${WEBLOGIC_CLASSPATH}:${VLJ_CP}:${POINTBASE_CLASSPATH}:${JAVA_HOME}/jre/
lib/rt.jar:${WL_HOME}/server/lib/webservices.jar:${CLASSPATH}"
```

Figure 4.3-3. Linux Admin Server—KAAJEE SSPI variable additions to the startWebLogic.sh file (*Before* additions)

To (additions shown in bold typeface):

```
CLASSPATH="${WEBLOGIC_CLASSPATH}:${propertiesdir}:${sspidir}:${commonpool}:${common
dbcp}:${commoncollection}:${VLJ_CP}:${POINTBASE_CLASSPATH}:${JAVA_HOME}/jre/lib/rt.
jar:${WL_HOME}/server/lib/webservices.jar:${CLASSPATH}"
```

Figure 4.3-4. Linux Admin Server—KAAJEE SSPI variable additions to the startWebLogic.sh file (*After* additions)

4.3.2.4.1.3 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>**/weblogic81/server/lib/mbeantypes). Classpaths are used by the Health***e***Vet-VistA applications.

Specifically, modify the lines shown below that follow the arguments to use when starting this server in the startWebLogic.sh file:

From:

```
${JAVA_HOME}/bin/java ${JAVA_VM} ${MEM_ARGS} ${JAVA_OPTIONS} -
Dweblogic.Name=${SERVER_NAME} -Dweblogic.ProductionModeEnabled=${PRODUCTION_MODE} -
Djava.security.policy="${WL_HOME}/server/lib/weblogic.policy" weblogic.Server
```

Figure 4.3-5. Linux Admin Server—KAAJEE SSPI argument additions to the startWebLogic.sh file (*Before* additions)

To (additions shown in bold typeface):

```
${JAVA_HOME}/bin/java ${JAVA_VM} ${MEM_ARGS} ${JAVA_OPTIONS} -
Dweblogic.alternateTypesDirectory=${sspidir} -Dweblogic.Name=${SERVER_NAME} -
Dweblogic.ProductionModeEnabled=${PRODUCTION_MODE} -
Djava.security.policy="${WL_HOME}/server/lib/weblogic.policy" weblogic.Server
```

Figure 4.3-6. Linux Admin Server—KAAJEE SSPI argument additions to the startWebLogic.sh file (*After* additions)

END: Linux Instructions

⏭ Linux users, skip to 4.3.2.4.3.

BEGIN: Microsoft Windows Instructions

4.3.2.4.2 (Windows: Admin Server) Modify the startWebLogic.cmd File

For Windows, the startWebLogic.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>**\

For example:

C:\bea\user_project\domains\kaajeewebdomain\

NOTE: In the examples that follow, some of the directory paths are represented by their **<Alias**>, as described in Table 4-2. 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.

4.3.2.4.2.1 Add Apache Connection Pool Jar Files to the SSPI Classpath

The following Apache connection pool jar files *must* be added to the SSPI classpath:

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

These files are located in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars/

Specifically, add the following lines after "set JAVA_VENDOR=Sun" in the startWebLogic.cmd file:

```
set ApacheCoonPool=<SSPI_STAGING_FOLDER>\kaajee_security_provider\common_pools_jar
set commonpool=%ApacheCoonPool%\commons-pool-1.2.jar
set commondbcp=%ApacheCoonPool%\commons-dbcp-1.2.1.jar
set commoncollection=%ApacheCoonPool%\commons-collections-3.1.jar
set propertiesdir=<SSPI_STAGING_FOLDER>\kaajee_security_provider\props
set sspidir=<SSPI_STAGING_FOLDER>\kaajee_security_provider
```

Figure 4.3-7. Windows Admin Server—KAAJEE SSPI classpath additions to the startWebLogic.cmd file (*Generic* example *with* <Alias> placeholders)

4.3.2.4.2.2 Add Variables to the SSPI Classpath

Add the following variables to the SSPI classpath:

- 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.)
- 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.

Specifically, modify the lines shown below that follow the "set CLASSPATH=%WEBLOGIC_CLASSPATH%;" line in the startWebLogic.cmd file.

From:

```
set
CLASSPATH=%WEBLOGIC_CLASSPATH%;%VLJ_CP%;%POINTBASE_CLASSPATH%;%JAVA_HOME%\jre\lib\r
t.jar;%WL_HOME%\server\lib\webservices.jar;%CLASSPATH%
```

Figure 4.3-8. Windows Admin Server—KAAJEE SSPI variable additions to the startWebLogic.cmd file (*Before* additions)

To (additions shown in bold typeface):

```
set
CLASSPATH=%WEBLOGIC_CLASSPATH%;%propertiesdir%;%sspidir%;%commonpool%;%commondbcp%;
%commoncollection%;%VLJ_CP%;%POINTBASE_CLASSPATH%;%JAVA_HOME%\jre\lib\rt.jar;%WL_HO
ME%\server\lib\webservices.jar;%CLASSPATH%
```

Figure 4.3-9. Windows Admin Server—KAAJEE SSPI variable additions to the startWebLogic.cmd file (*After* additions)

4.3.2.4.2.3 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>**\weblogic81\server\lib\mbeantypes). Classpaths are used by the Health***e***Vet-VistA applications.

Specifically, modify the lines shown below that follow the arguments to use when starting this server in the startWebLogic.cmd file.

From:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS% -
Dweblogic.Name=%SERVER_NAME% -Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE% -
Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy" weblogic.Server
```

Figure 4.3-10. Windows Admin Server—KAAJEE SSPI argument additions to the startWebLogic.cmd file (*Before* additions)

To (additions shown in bold typeface):

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS% -
Dweblogic.alternateTypesDirectory=%sspidir% -Dweblogic.Name=%SERVER_NAME% -
Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE% -
Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy" weblogic.Server
```

Figure 4.3-11. Windows Admin Server—KAAJEE SSPI argument additions to the startWebLogic.cmd file (*After* additions)

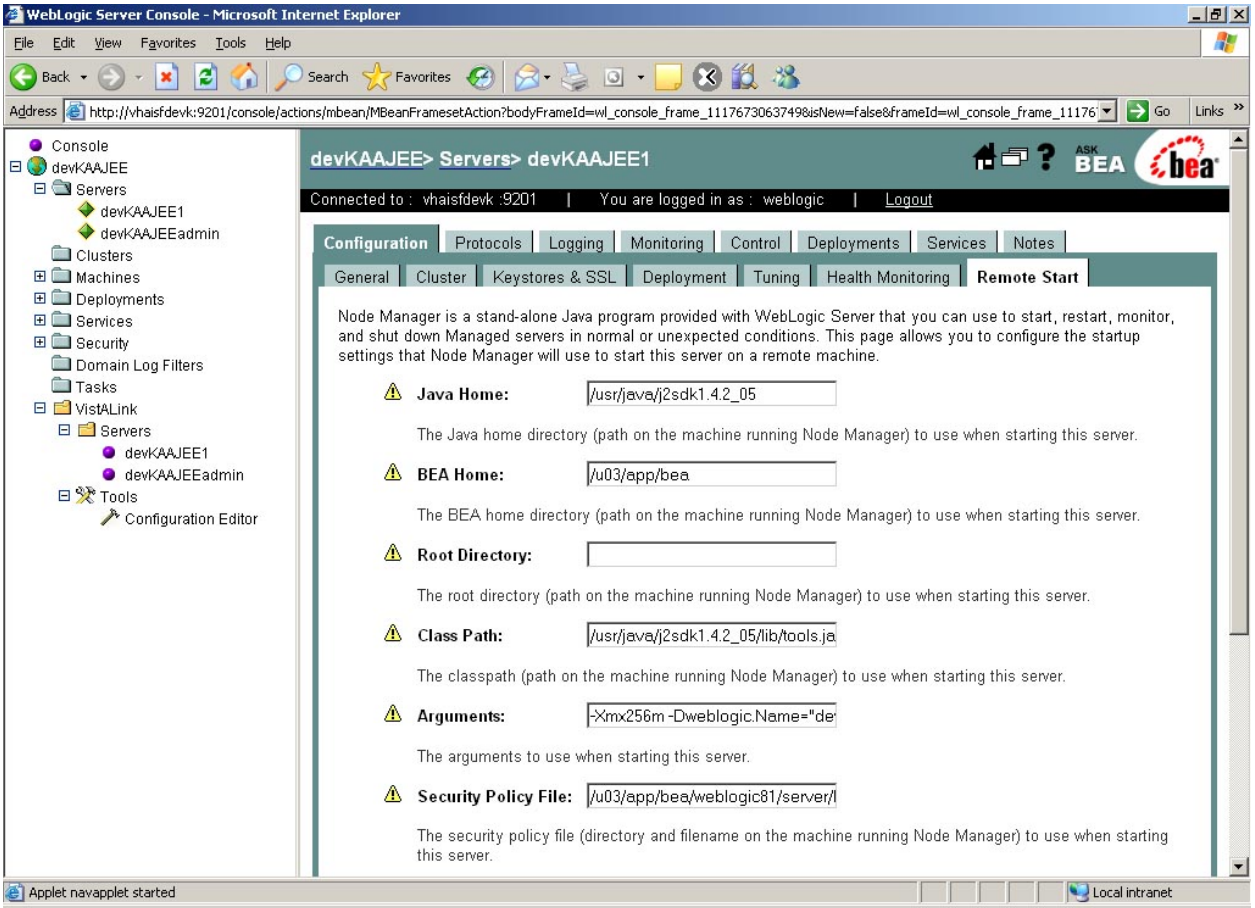
END: Microsoft Windows Instructions

⏭ Windows users, skip to 4.3.2.4.4.

BEGIN: Linux Instructions

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

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


WebLogic Server Console - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://vhaisfdevk:9201/console/actions/mbean/MBeanFramesetAction?bodyFrameId=wl_console_frame_1117673063749&isNew=false&frameId=wl_console_frame_11176
Go Links
Console
devKAAJEE
Servers
devKAAJEE1
devKAAJEEadmin
Clusters
Machines
Deployments
Services
Security
Domain Log Filters
Tasks
VistALink
Servers
devKAAJEE1
devKAAJEEadmin
Tools
Configuration Editor
ASK BEA
bea
Configuration Protocols Logging Monitoring Control Deployments Services Notes
General Cluster Keystores & SSL Deployment Tuning Health Monitoring Remote Start
Applet navapplet started
Local intranet

devKAAJEE> Servers> devKAAJEE1

Connected to : vhaisfdevk :9201 | You are logged in as : weblogic | Logout

Node Manager is a stand-alone Java program provided with WebLogic Server that you can use to start, restart, monitor, and shut down Managed servers in normal or unexpected conditions. This page allows you to configure the startup settings that Node Manager will use to start this server on a remote machine.

Java Home: /usr/java/j2sdk1.4.2_05

The Java home directory (path on the machine running Node Manager) to use when starting this server.

BEA Home: /u03/app/bea

The BEA home directory (path on the machine running Node Manager) to use when starting this server.

Root Directory:

The root directory (path on the machine running Node Manager) to use when starting this server.

Class Path: /usr/java/j2sdk1.4.2_05/lib/tools.ja

The classpath (path on the machine running Node Manager) to use when starting this server.

Arguments: -Xmx256m -Dweblogic.Name="de

The arguments to use when starting this server.

Security Policy File: /u03/app/bea/weblogic81/server/l

The security policy file (directory and filename on the machine running Node Manager) to use when starting this server.

Figure 4.3-12. Linux Managed Server—WebLogic Server Console Screen: Remote Start Tab

NOTE: In the examples that follow, some of the directory paths are represented by their <**Alias**>, as described in Table 4-2. 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.

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

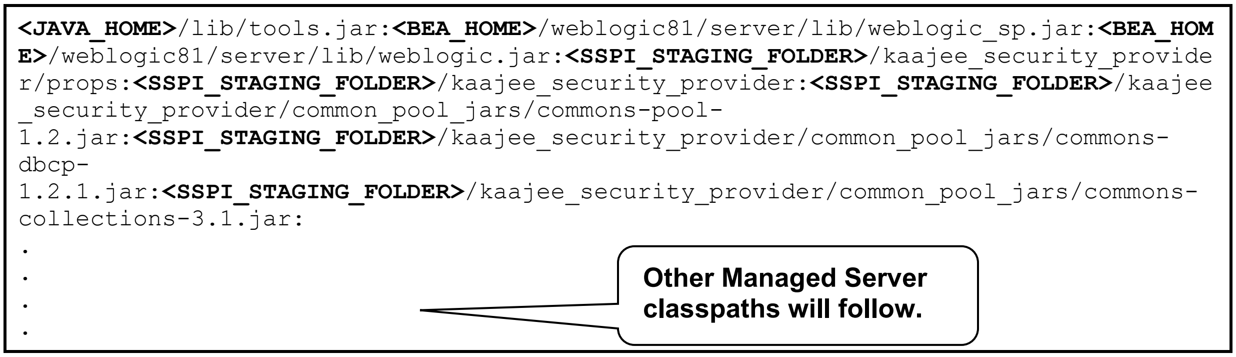
Add or replace the following KAAJEE SSPI-related classpaths in the **Class Path** field (i.e., the classpath used to start the Managed Server) on the **Remote Start** tab on the Managed Server(s):

- 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.)
- 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.



```
<JAVA_HOME>/lib/tools.jar:<BEA_HOME>/weblogic81/server/lib/weblogic_sp.jar:<BEA_HOM
E>/weblogic81/server/lib/weblogic.jar:<SSPI_STAGING_FOLDER>/kaajee_security_provide
r/props:<SSPI_STAGING_FOLDER>/kaajee_security_provider:<SSPI_STAGING_FOLDER>/kaajee
_security_provider/common_pool_jars/commons-pool-
1.2.jar:<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars/commons-
dbcp-
1.2.1.jar:<SSPI_STAGING_FOLDER>/kaajee_security_provider/common_pool_jars/commons-
collections-3.1.jar:
.
.
.
.
```

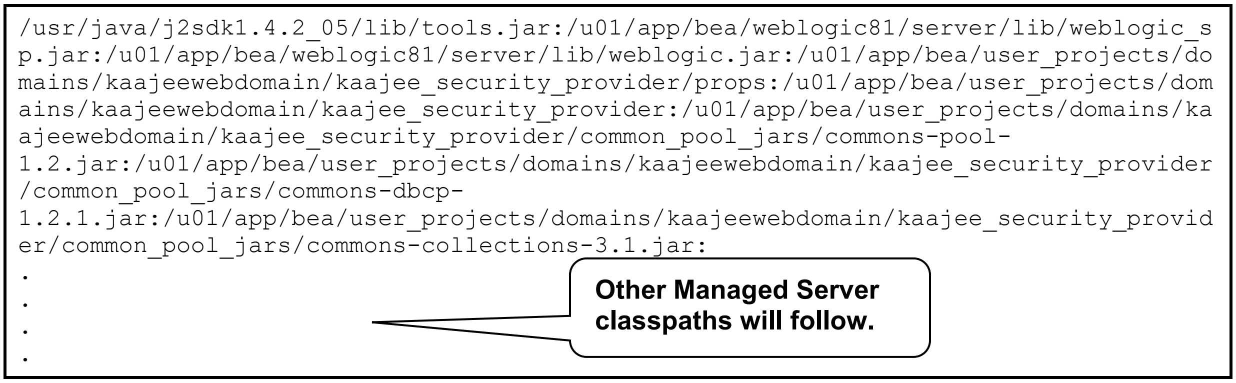
Other Managed Server classpaths will follow.

Figure 4.3-13. Linux Managed Server—KAAJEE SSPI classpath additions on the Remote Start tab (*Generic* example *with* <Alias> placeholders)

> **NOTE:** Other VistALink- and WebLogic-specific classpaths (e.g., vljConnector-1.5.1.**xxx**.jar and vljFoundationsLib-1.5.1.**xxx**.jar) will also be displayed in this field.

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

- **<JAVA_HOME>** = /usr/java/j2sdk1.4.2_05
- **<BEA_HOME>** = /u01/app/bea
- **<SSPI_STAGING_FOLDER>** = /u01/app/bea/user_projects/domains/kaajeewebdomain
- **<MANAGED_SERVER_NAME>** = devKAAJEE1



```
/usr/java/j2sdk1.4.2_05/lib/tools.jar:/u01/app/bea/weblogic81/server/lib/weblogic_s
p.jar:/u01/app/bea/weblogic81/server/lib/weblogic.jar:/u01/app/bea/user_projects/do
mains/kaajeewebdomain/kaajee_security_provider/props:/u01/app/bea/user_projects/dom
ains/kaajeewebdomain/kaajee_security_provider:/u01/app/bea/user_projects/domains/ka
ajeewebdomain/kaajee_security_provider/common_pool_jars/commons-pool-
1.2.jar:/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provider
/common_pool_jars/commons-dbcp-
1.2.1.jar:/u01/app/bea/user_projects/domains/kaajeewebdomain/kaajee_security_provid
er/common_pool_jars/commons-collections-3.1.jar:
.
.
.
.
```

Other Managed Server classpaths will follow.

Figure 4.3-14. Linux Managed Server—KAAJEE SSPI classpath additions/replacements on the Remote Start tab
(*Actual* example *without* <Alias> placeholders)

NOTE: Other VistALink- and WebLogic-specific classpaths (e.g., vljConnector-1.5.1.**xxx**.jar and vljFoundationsLib-1.5.1.**xxx**.jar) will also be displayed in this field.

4.3.2.4.3.2 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:**<USER_DOMAIN_HOME>**/managed_mylog4j.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 **Remote Start** tab on the Managed Server(s). The arguments are added or replaced in one long string, as shown below:

```
-Xmx256m -Dweblogic.Name="<MANAGED_SERVER_NAME>" -
Dgov.va.med.environment.servertype=WEBLOGIC -
Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:<USER_DOMAIN_HOME>/managed_mylog4j.xml -
Dweblogic.alternateTypesDirectory=<SSPI_STAGING_FOLDER>/kaajee_security_provider -
Dweblogic.ProductionModeEnabled=""
```

Figure 4.3-15. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Remote Start tab
(*Generic* example *with* <Alias> placeholders)

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

- **<MANAGED_SERVER_NAME>** = devKAAJEE1
- **<USER_DOMAIN_HOME>** = /u01/app/bea/user_projects/domains/kaajeewebdomain
- **<SSPI_STAGING_FOLDER>** = /u01/app/bea/user_projects/domains/kaajeewebdomain

```
-Xmx256m -Dweblogic.Name="devKAAJEE1" -Dgov.va.med.environment.servertype=WEBLOGIC
-Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:/u01/app/bea/user_projects/domains/kaajeewebdomain/manage
d_mylog4j.xml -
Dweblogic.alternateTypesDirectory=/u01/app/bea/user_projects/domains/kaajeewebdomai
n/kaajee_security_provider -Dweblogic.ProductionModeEnabled=""
```

Figure 4.3-16. Linux Managed Server—KAAJEE SSPI argument additions/replacements on the Remote Start tab
(*Actual* example *without* <Alias> placeholders)

4.3.2.4.3.3 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 **Remote Start** tab on the Managed Server(s):

```
<BEA_HOME>/weblogic81/server/lib/weblogic.policy
```

Figure 4.3-17. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Remote Start tab
(*Generic* example *with* <Alias> placeholders)

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

- **`<BEA_HOME>`** `= /u01/app/bea`

```
/u01/app/bea/weblogic81/server/lib/weblogic.policy
```

Figure 4.3-18. Linux Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Remote Start tab
(*Actual* example *without* <Alias> placeholders)

END: Linux Instructions

⏭ Linux users, skip to 4.3.2.4.5.

BEGIN: Microsoft Windows Instructions

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

Use the WebLogic Server Console to navigate to the **Remote 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 4-2. 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.

4.3.2.4.4.1 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 **Remote 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.)
- 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.

```
<JAVA_HOME>\lib\tools.jar;<BEA_HOME>\weblogic81\server\lib\weblogic_sp.jar;<BEA_HOM
E>\weblogic81\server\lib\weblogic.jar;<SSPI_STAGING_FOLDER>\kaajee_security_provide
r\props;<SSPI_STAGING_FOLDER>\kaajee_security_provider;<SSPI_STAGING_FOLDER>\kaajee
_security_provider\common_pool_jars\commons-pool-
1.2.jar;<SSPI_STAGING_FOLDER>\kaajee_security_provider\common_pool_jars\commons-
dbcp-
1.2.1.jar;<SSPI_STAGING_FOLDER>\kaajee_security_provider\common_pool_jars\commons-
collections-3.1.jar;
.
.
.
.
```

Other Managed Server classpaths will follow.

Figure 4.3-19. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Remote Start tab (*Generic* example *with* <Alias> placeholders)

NOTE: Other VistALink- and WebLogic-specific classpaths (e.g., vljConnector-1.5.1.**xxx**.jar and vljFoundationsLib-1.5.1.**xxx**.jar) will also be displayed in this field.

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

- **<JAVA_HOME>** = C:\java\j2sdk1.4.2_05
- **<BEA_HOME>** = C:\bea
- **<SSPI_STAGING_FOLDER>** = C:\bea\user_projects\domains\kaajeewebdomain
- **<MANAGED_SERVER_NAME>** = devKAAJEE1

```
C:\java\j2sdk1.4.2_05\lib\tools.jar;C:\bea\weblogic81\server\lib\weblogic_sp.jar;C:
\bea\weblogic81\server\lib\weblogic.jar;C:\bea\user_projects\domains\kaajeewebdomai
n\kaajee_security_provider\props;C:\bea\user_projects\domains\kaajeewebdomain\kaaje
e_security_provider;C:\bea\user_projects\domains\kaajeewebdomain\kaajee_security_pr
ovider\common_pool_jars\commons-pool-
1.2.jar;C:\bea\user_projects\domains\kaajeewebdomain\kaajee_security_provider\commo
n_pool_jars\commons-dbcp-
1.2.1.jar;C:\bea\user_projects\domains\kaajeewebdomain\kaajee_security_provider\com
mon_pool_jars\commons-collections-3.1.jar;
.
.
.
.
```

Other Managed Server classpaths will follow.

Figure 4.3-20. Windows Managed Server—KAAJEE SSPI classpath additions/replacements on the Remote Start tab (*Actual* example *without* <Alias> placeholders)

NOTE: Other VistALink- and WebLogic-specific classpaths (e.g., vljConnector-1.5.1.**xxx**.jar and vljFoundationsLib-1.5.1.**xxx**.jar) will also be displayed in this field.

4.3.2.4.4.2 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:**<USER_DOMAIN_HOME>**/managed_mylog4j.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 **Remote Start** tab on the Managed Server(s). The arguments are added or replaced in one long string, as shown below:

```
-Xmx256m -Dweblogic.Name="<MANAGED_SERVER_NAME>" -
Dgov.va.med.environment.servertype=WEBLOGIC -
Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:<USER_DOMAIN_HOME (with forward
slashes)>/managed_mylog4j.xml -
Dweblogic.alternateTypesDirectory=<SSPI_STAGING_FOLDER>\kaajee_security_provider -
Dweblogic.ProductionModeEnabled=""
```

Figure 4.3-21. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Remote Start tab
(*Generic* example *with* <Alias> placeholders)

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

- **<MANAGED_SERVER_NAME>** = devKAAJEE1
- **<USER_DOMAIN_HOME> (NOTE: with *forward* slashes)**= C:/bea/user_projects/domains/kaajeewebdomain
- **<SSPI_STAGING_FOLDER>** = C:\bea\user_projects\domains\kaajeewebdomain

```
-Xmx256m -Dweblogic.Name="devKAAJEE1" -Dgov.va.med.environment.servertype=WEBLOGIC
-Dgov.va.med.environment.production=false -
Dlog4j.configuration=file:///C:/bea/user_projects/domains/kaajeewebdomain/managed_m
ylog4j.xml -
Dweblogic.alternateTypesDirectory=C:\bea\user_projects\domains\kaajeewebdomain\kaaj
ee_security_provider -Dweblogic.ProductionModeEnabled=""
```

Figure 4.3-22. Windows Managed Server—KAAJEE SSPI argument additions/replacements on the Remote Start tab
(*Actual* example *without* <Alias> placeholders)

4.3.2.4.4.3 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 **Remote Start** tab on the Managed Server(s):

```
<BEA_HOME>\weblogic81\server\lib\weblogic.policy
```

Figure 4.3-23. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Remote Start tab (*Generic* example *with* <Alias> placeholders)

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

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

```
C:\bea\weblogic81\server\lib\weblogic.policy
```

Figure 4.3-24. Windows Managed Server—KAAJEE SSPI Security Policy File field addition/replacement on the Remote Start tab (*Actual* example *without* <Alias> placeholders)

END: Microsoft Windows Instructions

4.3.2.4.5 (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 Website.

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

4.3.2.4.5.1 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.

4.3.2.4.5.2 Create KAAJEE SSPI Tables

KAAJEE requires the following two SSPI SQL database tables:

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.

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

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_1.0.0.010.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 4-4):

```
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));
```

Figure 4.3-25. Oracle Database—Sample SSPI SQL script for KAAJEE table definitions

4.3.2.4.5.3 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 4.3.2.4.7.

4.3.2.4.6 (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 Website.

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

4.3.2.4.6.1 Create KAAJEE User ID & Schema

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

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

REDACTED

4.3.2.4.6.2 Create KAAJEE SSPI Tables

KAAJEE requires the following two SSPI SQL database tables:

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.

Table 4-5. Caché Database—KAAJEE SSPI SQL table definitions

ℹ **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_1.0.0.010.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 4-5):

```
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));
```

Figure 4.3-26. Caché Database—Sample SSPI SQL script for KAAJEE table definitions

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 Website:

REDACTED

4.3.2.4.6.3 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.

4.3.2.4.7 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_1.0.0.010.zip). The KaajeeDatabase.properties file is located in the following directory:

<SSPI_STAGING_FOLDER>/kaajee_security_provider/props

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

Figure 4.3-27. Sample KaajeeDatabase.properties file as delivered with KAAJEE

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 #4.3.2.4.5 [(Oracle Database) Create KAAJEE Schema & SSPI Tables"] or #4.3.2.4.6 ["(Caché Database) Create KAAJEE Schema & SSPI Tables"] and Table 4-4 or Table 4-5 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 #4.3.2.4.5 or #4.3.2.4.6 in this manual.

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

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

Figure 4.3-28. Oracle Database—Sample Driver and URL

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

Figure 4.3-29. Caché Database—Sample Driver and URL

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

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.

BEGIN: Linux Instructions

4.3.2.4.8 (Linux: Admin Server) Restart the WebLogic Application Server Domain (startWeblogic.sh)

4.3.2.4.8.1 Change the Directory

Change the directory to the <DOMAIN_NAME> (e.g., kaajeewebdomain is the WebLogic server domain name):

<USER_DOMAIN_HOME>

For example:

```
cd /u01/app/bea/user_project/domains/kaajeewebdomain
```

4.3.2.4.8.2 Enter the Start Command

Enter the following command after the "<USER_DOMAIN_HOME>" prompt:

```
./startWebLogic.sh
```

4.3.2.4.8.3 Wait for the Server to Come Up Before Proceeding

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

END: Linux Instructions

⏭ Linux users, skip to 4.3.2.4.10.

BEGIN: Microsoft Windows Instructions

4.3.2.4.9 (Windows: Admin Server) Restart the WebLogic Application Server Domain (startWeblogic.cmd)

4.3.2.4.9.1 Create a New DOS Shell

4.3.2.4.9.2 Change the Directory

Change the directory to the <DOMAIN_NAME> (e.g., kaajeewebdomain is the WebLogic server domain name):

<USER_DOMAIN_HOME>

For example:

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

4.3.2.4.9.3 Enter the Start Command

Enter the following command after the "<USER_DOMAIN_HOME>" prompt:

```
startWeblogic.cmd
```

NOTE: If you allowed the Configuration Wizard to start the server in the Windows Start Menu, you can replace Steps #4.3.2.4.10.1 through #4.3.2.4.10.3 with the following:

> Start > All Programs > WebLogic Platform 8.1 > User Projects > kaajeewebdomain > Start Server

4.3.2.4.9.4 Wait for the Server to Come Up Before Proceeding

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

END: Microsoft Windows Instructions

4.3.2.4.10 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:

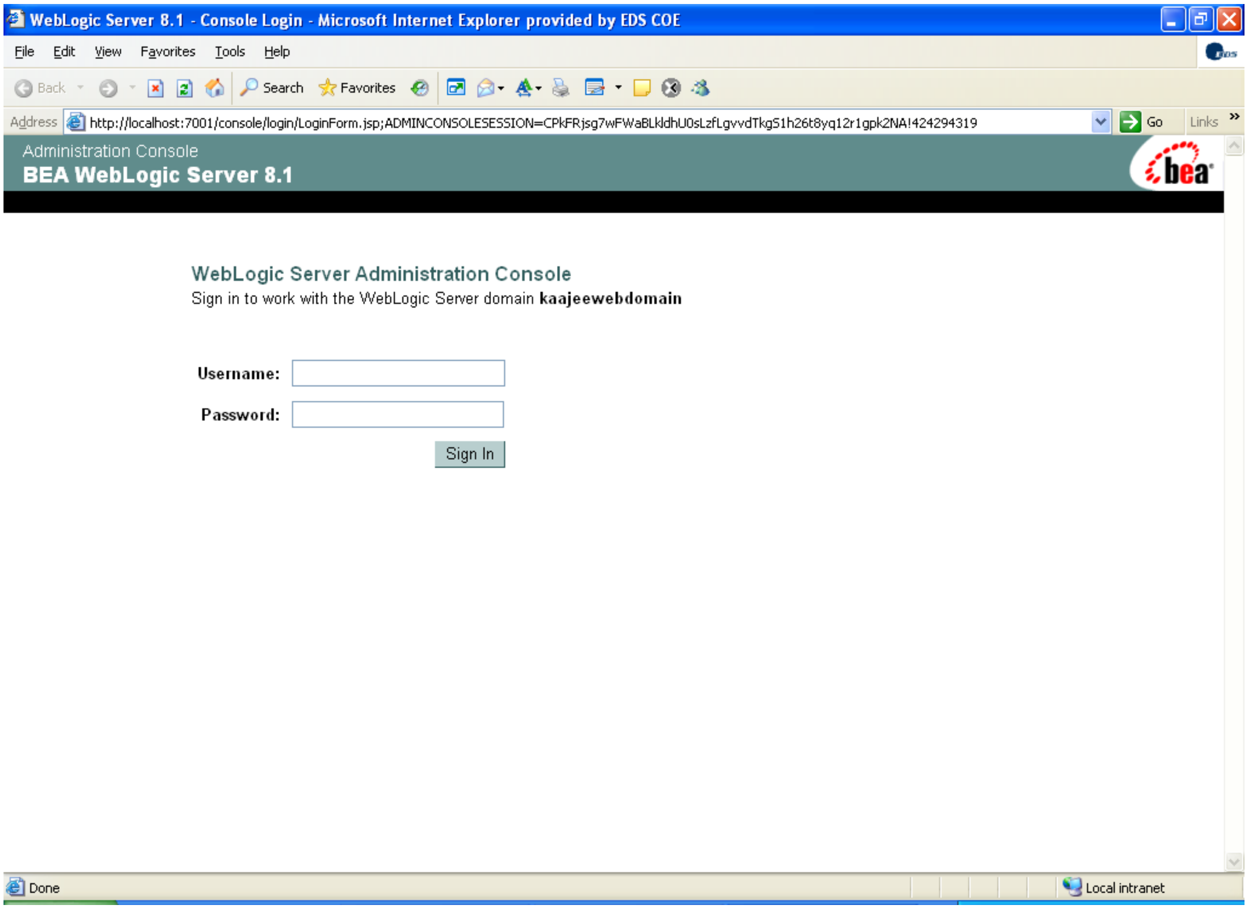
4.3.2.4.10.1 Connect to the WebLogic Console

For example:

```
URL    : http://localhost:7001/console/
```

4.3.2.4.10.2 Log onto the WebLogic Server Administration Console

Log onto the WebLogic Server Administration Console using the Boot User Name and User Password, as shown below:



WebLogic Server 8.1 - Console Login - Microsoft Internet Explorer provided by EDS COE

http://localhost:7001/console/login/LoginForm.jsp;ADMINCONSOLESESSION=CPkFRjsg7wFWaBLkldhU0sLzfLgvvdTkgS1h26t8yq12r1gpk2NA!424294319

Administration Console
BEA WebLogic Server 8.1

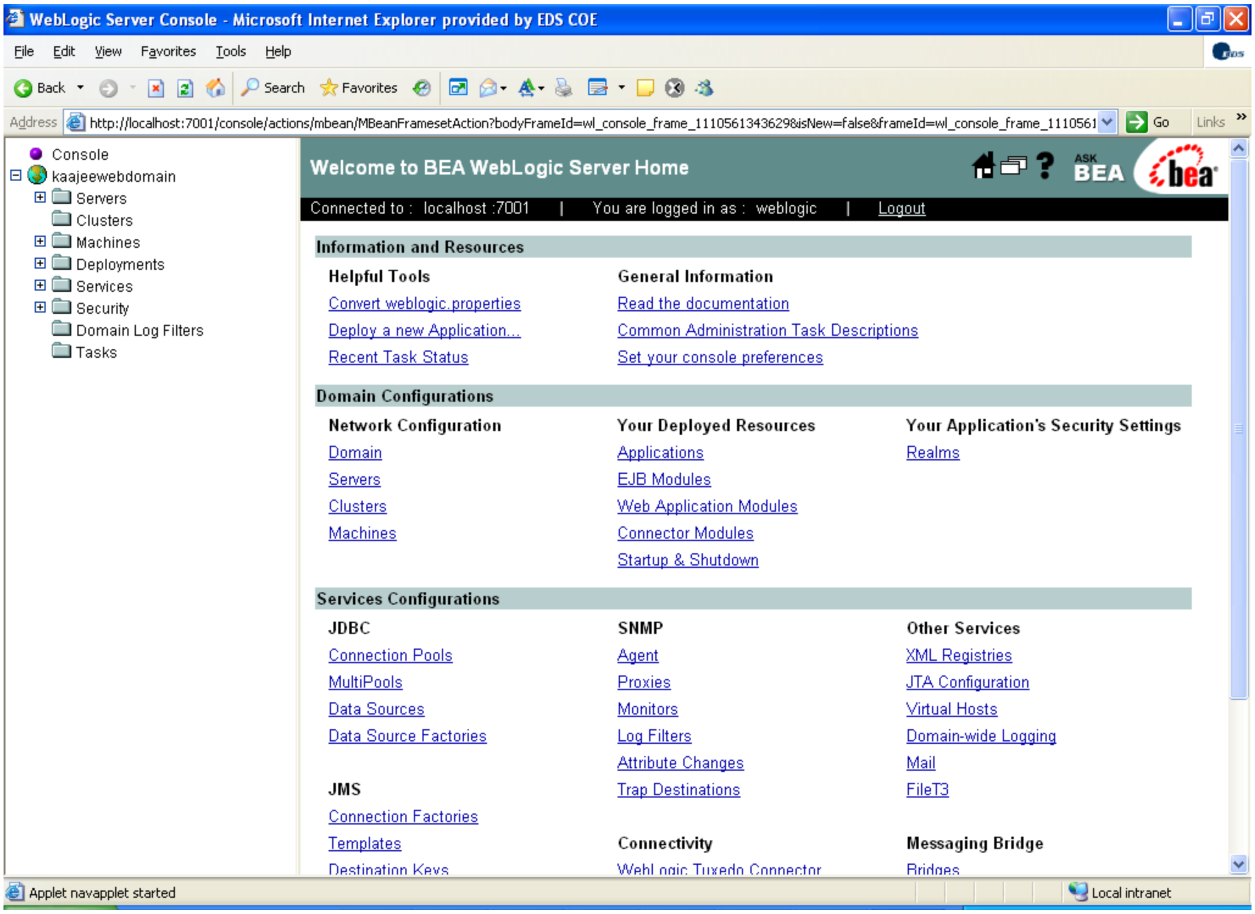
WebLogic Server Administration Console
Sign in to work with the WebLogic Server domain **kaajeewebdomain**

Username:

Password:

Figure 4.3-30. WebLogic Server Console Screen: Signon screen

After signing on (Figure 4.3-30), you are presented with the following WebLogic Server Administration Console screen:


WebLogic Server Console - Microsoft Internet Explorer provided by EDS COE
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
Go
Links
ASK BEA
bea
Applet navapplet started
Local intranet

http://localhost:7001/console/actions/mbean/MBeanFramesetAction?bodyFrameId=wl_console_frame_1110561343629&isNew=false&frameId=wl_console_frame_1110561

- Console
- kaajeewebdomain
  - Servers
  - Clusters
  - Machines
  - Deployments
  - Services
  - Security
  - Domain Log Filters
  - Tasks

Welcome to BEA WebLogic Server Home

Connected to : localhost :7001 | You are logged in as : weblogic | Logout

Information and Resources

Helpful Tools	General Information
Convert weblogic.properties	Read the documentation
Deploy a new Application...	Common Administration Task Descriptions
Recent Task Status	Set your console preferences

Domain Configurations

Network Configuration	Your Deployed Resources	Your Application's Security Settings
Domain	Applications	Realms
Servers	EJB Modules	
Clusters	Web Application Modules	
Machines	Connector Modules	
	Startup & Shutdown	

Services Configurations

JDBC	SNMP	Other Services
Connection Pools	Agent	XML Registries
MultiPools	Proxies	JTA Configuration
Data Sources	Monitors	Virtual Hosts
Data Source Factories	Log Filters	Domain-wide Logging
	Attribute Changes	Mail
JMS	Trap Destinations	FileT3
Connection Factories		
Templates	**Connectivity**	**Messaging Bridge**
Destination Keys	WebLogic Tuxedo Connector	Bridges

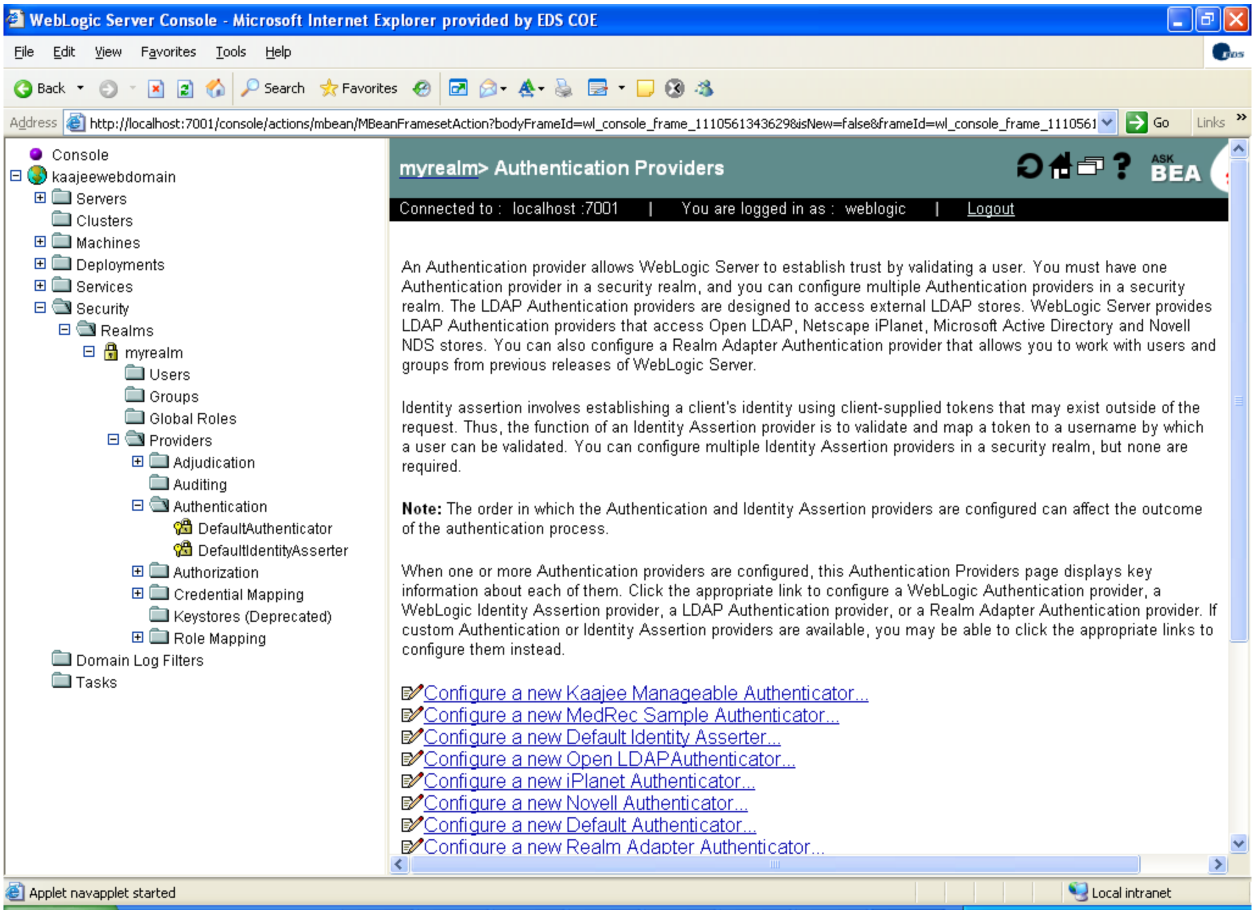
Figure 4.3-31. WebLogic Server Console Screen: WebLogic server home

4.3.2.4.10.3 Navigate to the Authentication Directory

Navigate to the Authentication directory, as shown in Figure 4.3-32:

Security > Realms > myrealm > Providers >Authentication

Click on **Authentication**, as shown below:


WebLogic Server Console - Microsoft Internet Explorer provided by EDS COE
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:7001/console/actions/mbean/MBeanFramesetAction?bodyFrameId=wl_console_frame_1110561343629&isNew=false&frameId=wl_console_frame_1110561
Go Links
Console
kaajeewebdomain
Servers
Clusters
Machines
Deployments
Services
Security
Realms
myrealm
Users
Groups
Global Roles
Providers
Adjudication
Auditing
Authentication
DefaultAuthenticator
DefaultIdentityAsserter
Authorization
Credential Mapping
Keystores (Deprecated)
Role Mapping
Domain Log Filters
Tasks
myrealm> Authentication Providers
ASK BEA
Connected to : localhost :7001 | You are logged in as : weblogic | Logout
Applet navapplet started
Local intranet

An Authentication provider allows WebLogic Server to establish trust by validating a user. You must have one Authentication provider in a security realm, and you can configure multiple Authentication providers in a security realm. The LDAP Authentication providers are designed to access external LDAP stores. WebLogic Server provides LDAP Authentication providers that access Open LDAP, Netscape iPlanet, Microsoft Active Directory and Novell NDS stores. You can also configure a Realm Adapter Authentication provider that allows you to work with users and groups from previous releases of WebLogic Server.

Identity assertion involves establishing a client's identity using client-supplied tokens that may exist outside of the request. Thus, the function of an Identity Assertion provider is to validate and map a token to a username by which a user can be validated. You can configure multiple Identity Assertion providers in a security realm, but none are required.

Note: The order in which the Authentication and Identity Assertion providers are configured can affect the outcome of the authentication process.

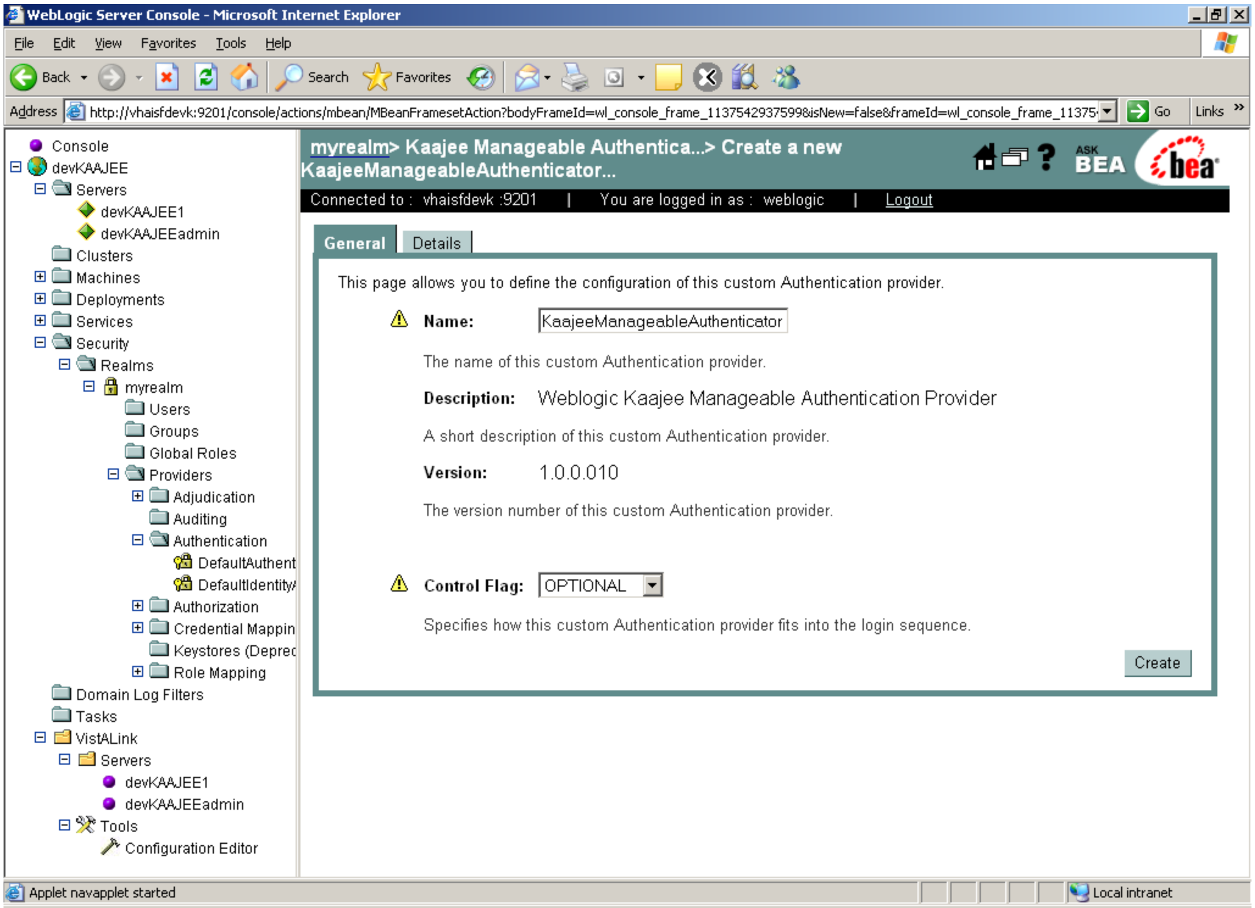
When one or more Authentication providers are configured, this Authentication Providers page displays key information about each of them. Click the appropriate link to configure a WebLogic Authentication provider, a WebLogic Identity Assertion provider, a LDAP Authentication provider, or a Realm Adapter Authentication provider. If custom Authentication or Identity Assertion providers are available, you may be able to click the appropriate links to configure them instead.

Configure a new Kaajee Manageable Authenticator...
Configure a new MedRec Sample Authenticator...
Configure a new Default Identity Asserter...
Configure a new Open LDAPAuthenticator...
Configure a new iPlanet Authenticator...
Configure a new Novell Authenticator...
Configure a new Default Authenticator...
Configure a new Realm Adapter Authenticator...

Figure 4.3-32. WebLogic Server Console Screen: myrealm> Authentication Providers screen

4.3.2.4.10.4 Create a new KAAJEE Manageable Authenticator

Click on the Configure a new Kaajee Manageable Authenticator option (Figure 4.3-32). You will then see the following screen:



myrealm> Kaajee Manageable Authentica...> Create a new KaajeeManageableAuthenticator...

Connected to : vhaisfdevk :9201 | You are logged in as : weblogic | Logout

This page allows you to define the configuration of this custom Authentication provider.

Name: KaajeeManageableAuthenticator

The name of this custom Authentication provider.

Description: Weblogic Kaajee Manageable Authentication Provider

A short description of this custom Authentication provider.

Version: 1.0.0.010

The version number of this custom Authentication provider.

Control Flag: OPTIONAL

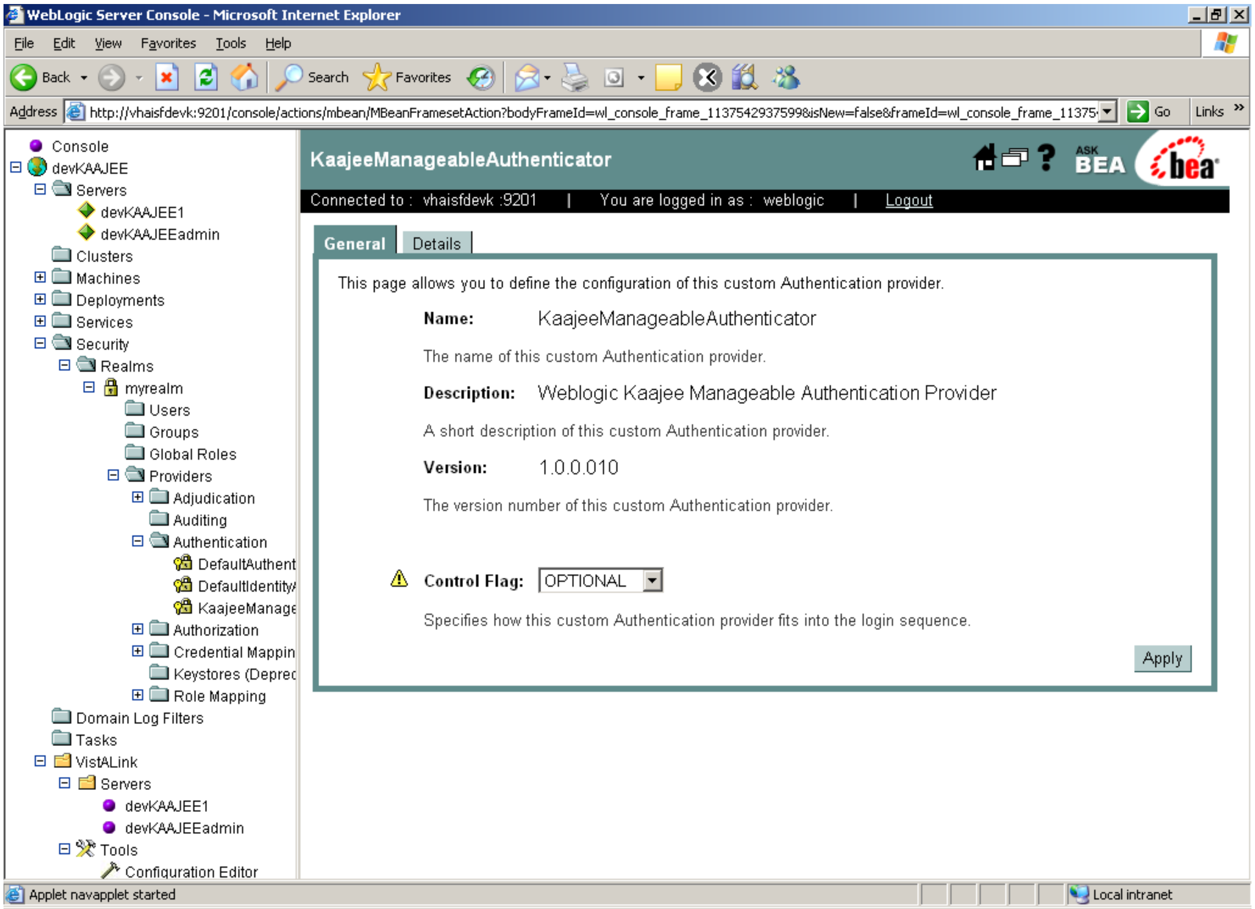
Specifies how this custom Authentication provider fits into the login sequence.

Figure 4.3-33. WebLogic Server Console Screen: Create a new KaajeeManageableAuthenticator screen

Users *must* accept the default settings (e.g., KaajeeManageableAuthenticator name) and click **Create**.

4.3.2.4.10.5 Configure a new KAAJEE Manageable Authenticator

After creating the KAAJEE authenticator, you will then see the following screen:



KaajeeManageableAuthenticator

Connected to : vhaisfdevk :9201 | You are logged in as : weblogic | Logout

This page allows you to define the configuration of this custom Authentication provider.

Name: KaajeeManageableAuthenticator

The name of this custom Authentication provider.

Description: Weblogic Kaajee Manageable Authentication Provider

A short description of this custom Authentication provider.

Version: 1.0.0.010

The version number of this custom Authentication provider.

⚠ **Control Flag:** OPTIONAL

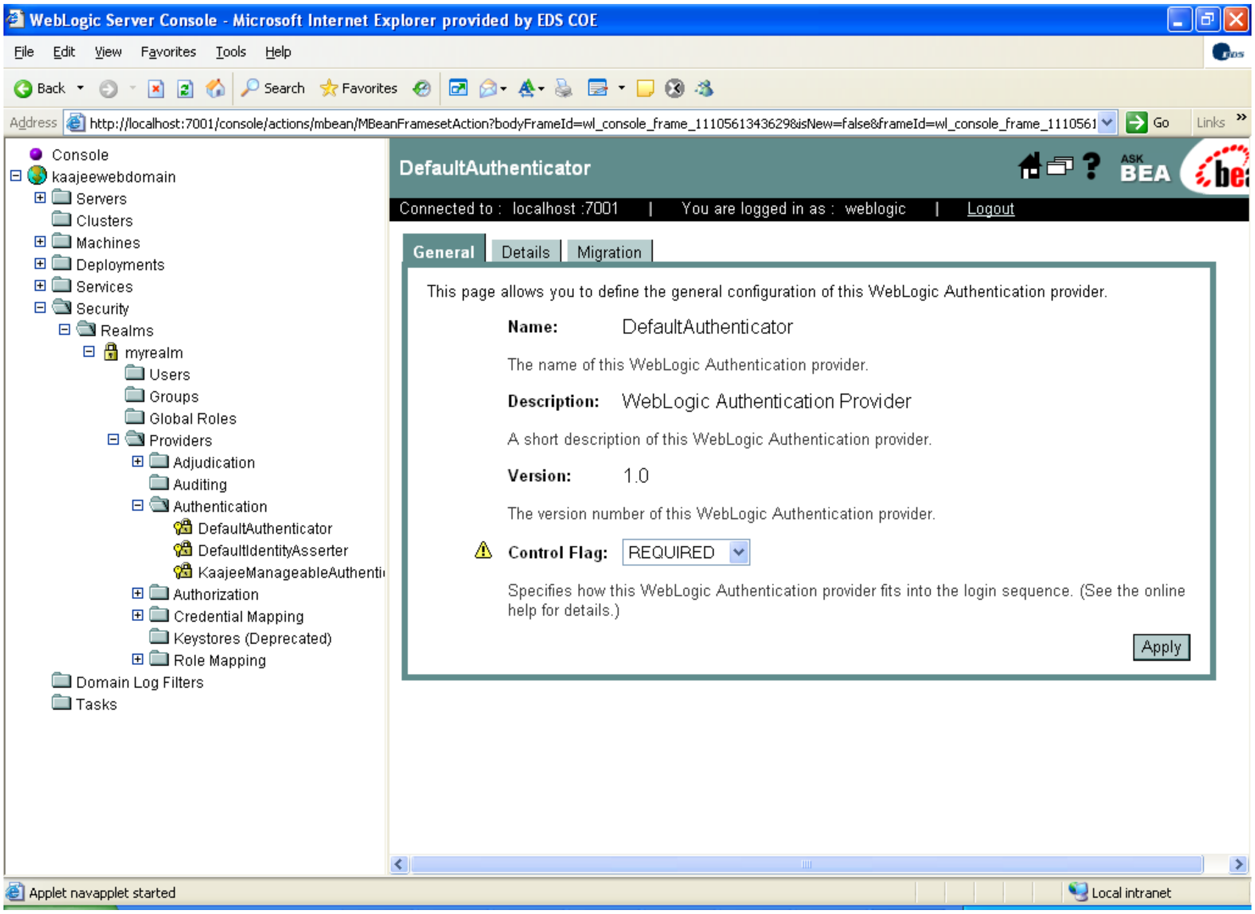
Specifies how this custom Authentication provider fits into the login sequence.

Figure 4.3-34. WebLogic Server Console Screen: KaajeeManageableAuthenticator screen

Users *must* accept the default settings and click **Apply**.

4.3.2.4.10.6 Configure the DefaultAuthenticator

Click on DefaultAuthenticator as shown below:



http://localhost:7001/console/actions/mbean/MBeanFramesetAction?bodyFrameId=wl_console_frame_1110561343629&isNew=false&frameId=wl_console_frame_1110561

DefaultAuthenticator

Connected to : localhost :7001 | You are logged in as : weblogic | Logout

This page allows you to define the general configuration of this WebLogic Authentication provider.

Name: DefaultAuthenticator

The name of this WebLogic Authentication provider.

Description: WebLogic Authentication Provider

A short description of this WebLogic Authentication provider.

Version: 1.0

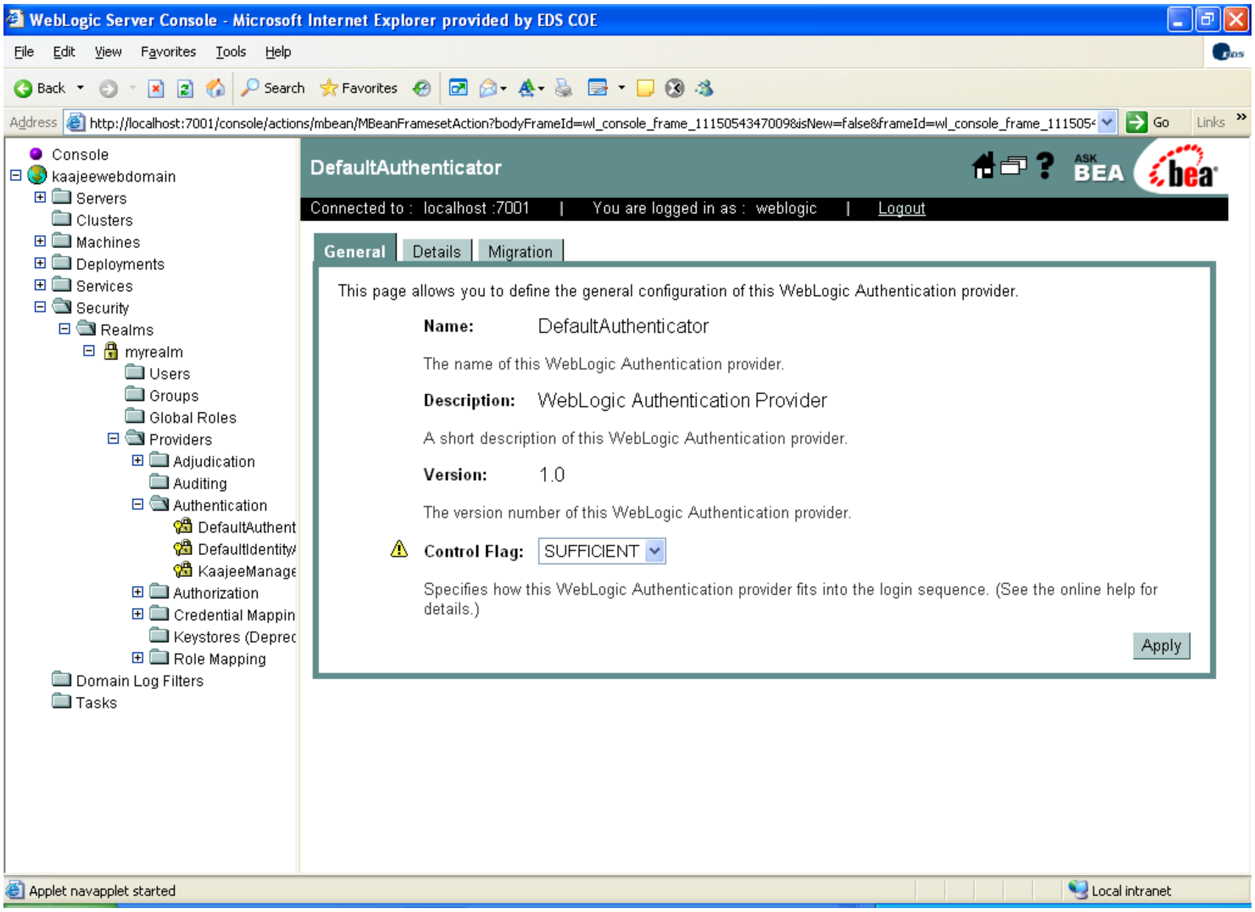
The version number of this WebLogic Authentication provider.

⚠ **Control Flag:** REQUIRED

Specifies how this WebLogic Authentication provider fits into the login sequence. (See the online help for details.)

Figure 4.3-35. WebLogic Server Console Screen: DefaultAuthenticator screen—Change Control Flag setting

Use the dropdown box next to the Control Flag field to change the setting to **SUFFICIENT** and then click **Apply**, as shown in Figure 4.3-36.



DefaultAuthenticator

Connected to : localhost :7001 | You are logged in as : weblogic | Logout

This page allows you to define the general configuration of this WebLogic Authentication provider.

Name: DefaultAuthenticator

The name of this WebLogic Authentication provider.

Description: WebLogic Authentication Provider

A short description of this WebLogic Authentication provider.

Version: 1.0

The version number of this WebLogic Authentication provider.

⚠ **Control Flag:** SUFFICIENT

Specifies how this WebLogic Authentication provider fits into the login sequence. (See the online help for details.)

Figure 4.3-36. WebLogic Server Console Screen: DefaultAuthenticator screen—SUFFICIENT Control Flag setting

4.3.2.4.10.7 Stop the WebLogic Application Server

Stop the WebLogic Application Server using the WebLogic console software (i.e., WebLogic Server 8.1 Console Login).

4.3.2.4.10.8 Reboot/Restart the WebLogic Application Server

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

4.3.2.4.10.9 Verify all Changes Have Taken Place

Use the WebLogic console software (i.e., WebLogic Server 8.1 Console Login) to navigate to the following locations:

- Kaajeewebdomain/Security/Realms/myrealm/Users.
- Kaajeewebdomain/Security/Realms/myrealm/Groups.

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

4.4 Configure SDS 3.0 (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 Website.

NOTE: KAAJEE works with SDS 3.0 or higher; however, KAAJEE 1.0.1.xxx distributes SDS 13.0 client jar files as part of the Sample Web Application. If you deploy the KAAJEE Sample Web Application and intend to use a different version of SDS, those client jar files will need to be swapped out for the appropriate version of the SDS client jar files.

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*.

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

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

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

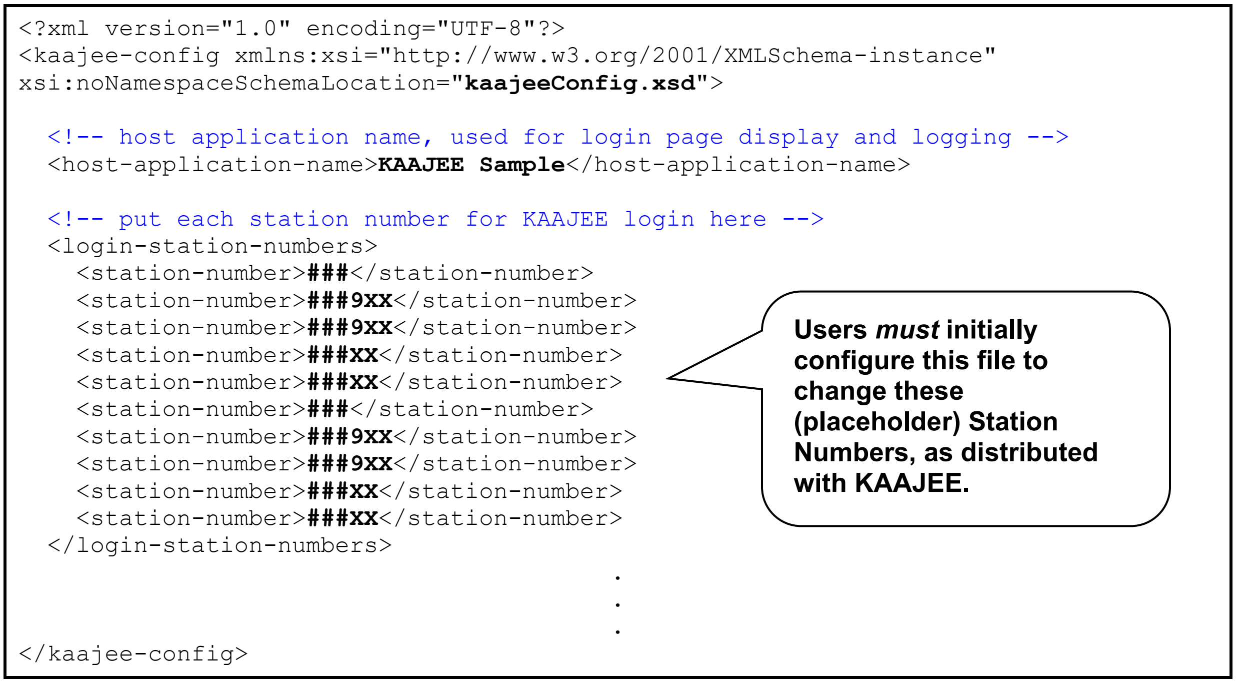
1. Unzip the application's ear file—Explode the artifact.
2. For any war file that implements KAAJEE authentication inside the ear file, unzip the war file.

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 1.0.1.xxx:



```
<?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.

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

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 Implement SSL in the kaajeeConfig.xml File *(optional)*

KAAJEE provides Secured Socket Layer (SSL) support. KAAJEE can be configured to use SSL when the Web login page is presented to the user. In order to configure use of SSL, the KAAJEE configuration file (i.e. kaajeeConfig.xml) *must* be edited as described below:

1. Uncomment the <ssl-listen-port-number> xml tag.
2. Change the default value from "7002" with an appropriate value that represents the SSL listen port on your J2EE Application Server.

   From:

```
  <!-- BEAWeblogic Server SSL listen port , used for login page to implement
SSL   -->
  <!--   <ssl-listen-port-number>7002</ssl-listen-port-number>  -->
```

Figure 4.5-2. Sample SSL excerpt of the kaajeeConfig.xml file (*before* edits)

   To:

```
  <!-- BEAWeblogic Server SSL listen port , used for login page to implement
SSL   -->
<ssl-listen-port-number>####</ssl-listen-port-number>
```

Figure 4.5-3. Sample SSL excerpt of the kaajeeConfig.xml file (*after* edits)

   Where "####" represents the SSL listen port on your J2EE Application Server.

Currently, KAAJEE does *not* account for multiple environment scenarios with SSL. For example:

- It does *not* account for mapping of *non*-SSL to SSL and vice versa from an external source such as a load balancer (e.g., BIG-IP from F5).
- In addition, when the Web login page is in SSL, KAAJEE assumes that upon return from the Web login page to the targeted protected page that the desired mode should be *non*-SSL.

REF: To request an SSL Certificate for use with your server in the Department of Veterans Affairs, visit the following Website:

https://onsite.verisign.com/USDepartmentofVeteransAffairs/serverEnroll.htm

4.6 Test the KAAJEE Installation Using the KAAJEE Sample Web Application *(recommended)*

KAAJEE 1.0.1.xxx distributes a KAAJEE Sample Web Application as both an exploded and packaged EAR file. Both the exploded ear and packaged ear files are located in the following directory:

- Packaged:

  <STAGING_FOLDER>/kaajee-1.0.1.xxx/samples/kaajeeSampleApp-1.0.1.xxx.ear

- Exploded:

  <STAGING_FOLDER>/kaajee-1.0.1.xxx/samples/exploded/kaajeeSampleApp-1.0.1.xxxEAR/

Neither of these files is plug-and-play. Thus, it requires that some additional procedures be performed by developers/Web administrators, as outlined in the steps that follow.

The KAAJEE Sample Web Application is a J2EE Web-based application that has a protected resource, which is protected by the XUKAAJEE_SAMPLE VistA M security key (security role). In addition, this application is configured to use Form-based Authentication as its authentication method.

The form login configuration is configured to use KAAJEE as the login provider for this Form-based Authentication method. To complete a successful login, the login user *must* pass both authentication and authorization requirements.

REF: For more information on Form-based Authentication, please refer to the "J2EE Form-based Authentication" topic in Chapter 1, "KAAJEE Overview," in the *KAAJEE Deployment Guide*.

First, an initial authentication occurs against a VistA M Server (i.e., Access and Verify codes). Then, if the login user passes this phase, the XUKAAJEE_SAMPLE VistA M security key is used by the KAAJEE Sample Web Application to create a J2EE group/principal of the same name on the J2EE Application Server if not already created. In addition, the login user will be assigned membership to this group on the J2EE Application Server during the login session. This membership is necessary as the authorization aspect of Form-based Authentication validates the role-based access by the membership of the associated group/principal.

NOTE: The XUKAAJEE_SAMPLE VistA M security key is only required by the KAAJEE Sample Web Application.

Running the KAAJEE Sample Web application is *not* required; however, it may benefit first time developers new to KAAJEE or those who want to test KAAJEE outside of their own Web application. It may be more useful to SQA and/or Testing Services testing KAAJEE as a baseline outside of the consuming Web application.

4.6.1 Allocate the XUKAAJEE_SAMPLE Security Key *(required)*

On the VistA M Server, use VA FileMan to allocate (add) the XUKAAJEE_SAMPLE VistA M security key, added with Kernel Patch XU*8.0*451, to any user who wants to test and log into VistA using the KAAJEE Sample Web Application.

One way to check whether or not you have the XUKAAJEE_SAMPLE VistA M security key in the SECURITY KEY file (#19.1) on the VistA M Server is to use VA FileMan as follows:

```
>D Q^DI


VA FileMan 22.0


Select OPTION: INQUIRE TO FILE ENTRIES


OUTPUT FROM WHAT FILE: SECURITY KEY// <Enter>
Select SECURITY KEY NAME: XUKAAJEE_SAMPLE
ANOTHER ONE: <Enter>
STANDARD CAPTIONED OUTPUT? Yes// <Enter>  (Yes)
Include COMPUTED fields:  (N/Y/R/B): NO// <Enter> - No record number (IEN), no
Computed Fields

NAME: XUKAAJEE_SAMPLE
  DESCRIPTIVE NAME: KAAJEE SAMPLE WEB APPLICATION
  SEND TO J2EE: Yes
 DESCRIPTION:  This key is required to access the KAAJEE Sample Web
 Application.

Select SECURITY KEY NAME: <Enter>


Select OPTION: <Enter>
>
```

Figure 4.6-1. Verifying that the XUKAAJEE_SAMPLE VistA M security key is installed on your system

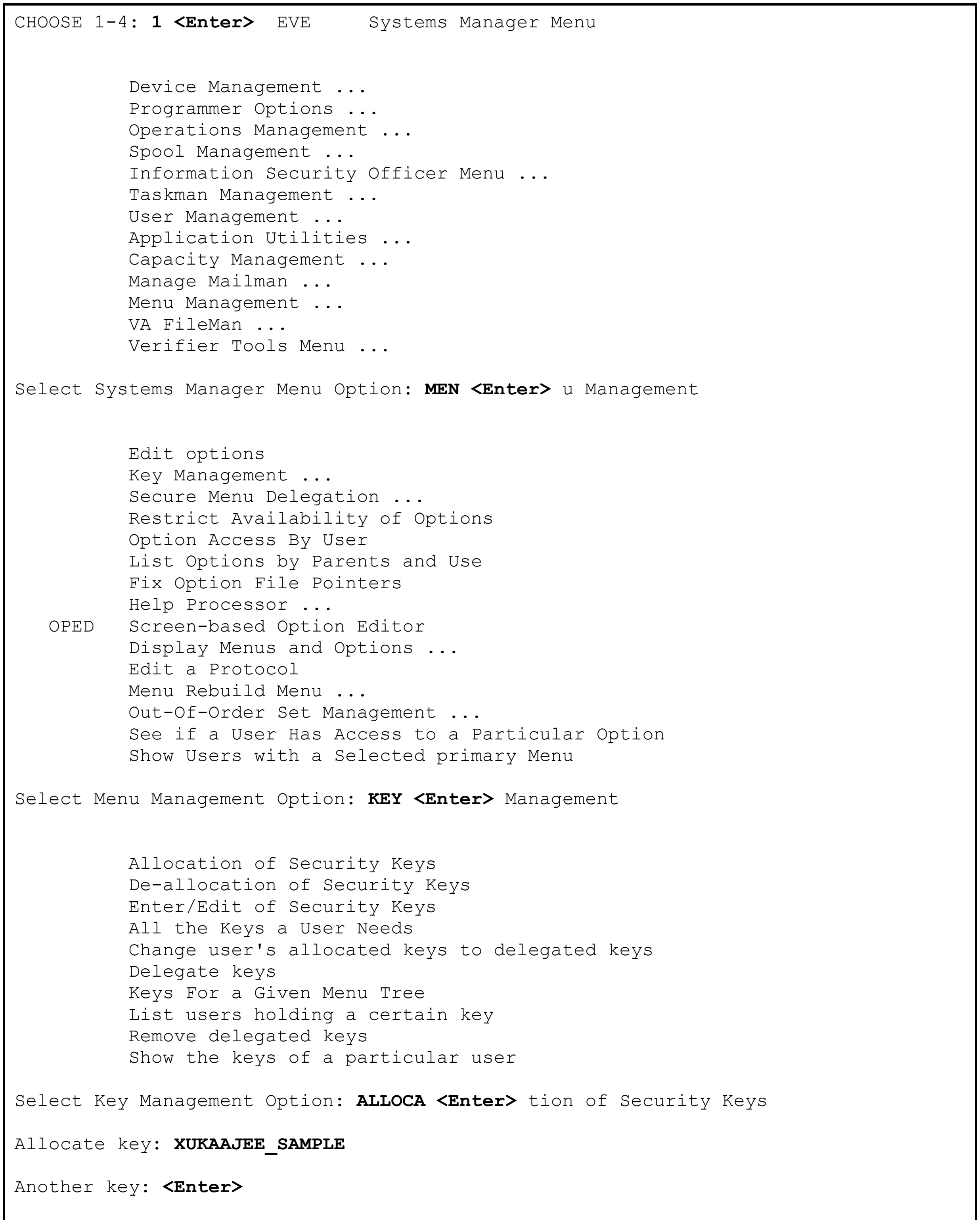
If you do *not* get double question marks ("**??**") at the "Select SECURITY KEY NAME:" prompt when you enter "**XUKAAJEE_SAMPLE**," then you have this VistA M security key installed on your system.

If you have the KAAJEE Sample Web Application deployed, then you can test the KAAJEE login.

Before allocating this VistA M security key to your J2EE users, you will get a Forms Authentication error page (loginerror.jsp).

After allocating this VistA M security key to your J2EE users, you and your users should be able to login successfully using the KAAJEE Sample Web Application provided that you and your users have this XUKAAJEE_SAMPLE VistA M security key.

Figure 4.6-2 is a screen capture of how to allocate the XUKAAJEE_SAMPLE VistA M security key to your J2EE users. This example assumes that the VistA M system administrator has the Kernel EVE menu:



```
CHOOSE 1-4: 1 <Enter>  EVE     Systems Manager Menu


          Device Management ...
          Programmer Options ...
          Operations Management ...
          Spool Management ...
          Information Security Officer Menu ...
          Taskman Management ...
          User Management ...
          Application Utilities ...
          Capacity Management ...
          Manage Mailman ...
          Menu Management ...
          VA FileMan ...
          Verifier Tools Menu ...

Select Systems Manager Menu Option: MEN <Enter> u Management


          Edit options
          Key Management ...
          Secure Menu Delegation ...
          Restrict Availability of Options
          Option Access By User
          List Options by Parents and Use
          Fix Option File Pointers
          Help Processor ...
   OPED   Screen-based Option Editor
          Display Menus and Options ...
          Edit a Protocol
          Menu Rebuild Menu ...
          Out-Of-Order Set Management ...
          See if a User Has Access to a Particular Option
          Show Users with a Selected primary Menu

Select Menu Management Option: KEY <Enter> Management


          Allocation of Security Keys
          De-allocation of Security Keys
          Enter/Edit of Security Keys
          All the Keys a User Needs
          Change user's allocated keys to delegated keys
          Delegate keys
          Keys For a Given Menu Tree
          List users holding a certain key
          Remove delegated keys
          Show the keys of a particular user

Select Key Management Option: ALLOCA <Enter> tion of Security Keys

Allocate key: XUKAAJEE_SAMPLE

Another key: <Enter>
```

```
Holder of key: XUUSER,ONE <Enter>      AC

Another holder: TESTER, KAAJEE <Enter>  TESTER,KAAJEE WEBKAT       KWT

Another holder: <Enter>

You've selected the following keys:

XUKAAJEE_SAMPLE

You've selected the following holders:

XUUSER,ONE                 TESTER,KAAJEE WEBKAT

You are allocating keys.  Do you wish to proceed? YES// <Enter>

XUKAAJEE_SAMPLE being assigned to:
     XUUSER,ONE                Person already holds key - no action taken
     TESTER,KAAJEE WEBKAT      Person already holds key - no action taken


          Allocation of Security Keys
          De-allocation of Security Keys
          Enter/Edit of Security Keys
          All the Keys a User Needs
          Change user's allocated keys to delegated keys
          Delegate keys
          Keys For a Given Menu Tree
          List users holding a certain key
          Remove delegated keys
          Show the keys of a particular user

Select Key Management Option:
```

This text is only returned if the user has already been allocated the XUKAAJEE_SAMPLE VistA M security key.

Figure 4.6-2. Allocating the XUKAAJEE_SAMPLE VistA M security key—Sample user dialogue

4.6.2 Edit jdbc.properties File *(required)*

KAAJEE 1.0.1.xxx distributes two template versions of the jdbc.properties file, configured based on the database type:

- jdbc.properties.cache
- jdbc.properties.oracle

REF: For samples of these jdbc.properties files, please refer to the "Access VA Standard Data Services (SDS) Tables" topic in Chapter 4, "Integrating KAAJEE with an Application," in the *KAAJEE Deployment Guide*.

These files are located in following directory:

<STAGING_FOLDER>\samples\exploded\kaajeeSampleApp-1.0.1.xxxEAR\APP-INF\classes\gov\va\stddata\factory\db

Edit the appropriate file for your database and rename it to:

jdbc.properties

REF: For proper configuration procedures, please follow the instructions as provided in the SDS documentation.

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

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

NOTE: SDS also distributes a jdbc.properties.hsqldb file; however, KAAJEE has *not* tested against this type of database and therefore does *not* include this file as part of the KAAJEE Sample Web Application distribution.

4.6.3 Edit the kaajeeConfig.xml File *(required)*

If you have not already edited the kaajeeConfig.xml file as described in Step #4.5, you *must* edit the kaajeeConfig.xml file and replace the placeholder Station Numbers with the Station Numbers appropriate to your Web application.

4.6.4 Edit the web.xml File *(required)*

Edit the web.xml file. Change the value in the <run as> tag to the appropriate admin user for your J2EE Application Server (e.g., weblogic).

REF: For a sample entry of the web.xml file, please refer to the "Review/Use KAAJEE Files for Web-based Applications" topic in Chapter 3, "KAAJEE Installation Instructions for Developers," in the *KAAJEE Deployment Guide*.

4.6.5 Deploy and Test the KAAJEE Sample Web Application with the Updated kaajeeConfig.xml File *(required)*

Use WebLogic to deploy the KAAJEE Sample Web application ear or standalone war file with the updated kaajeeConfig.xml file (see #4.5 and 4.6.3) on all appropriate application servers. Test the deployed 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.

4.7 (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 Website.

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., startWebLogic.sh or startWebLogic.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 Health***e***Vet-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*. Also, sample log4j configuration files are included with the VistALink 1.5 software distribution.

REF: For more information on VistALink, please refer to the VistALink documentation located on the VHA Software Document Library (VDL) Website at the following Website:

http://www.va.gov/vdl/application.asp?appid=163

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 Health**e**Vet-VistA Applications* document available at the following Website:

http://vista.med.va.gov/vistaarch/healthevet/Documents/Log4j%20Guidance%20v1.0.doc

4.7.1 Configure Application for log4j

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

4.7.2 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:

```
  <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>
```

Figure 4.7-1. Sample excerpt of the mylog4j.xml file—Editing common log file name and location (Windows)

In this example (Figure 4.7-1), the following common log file name and location is indicated:

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

The Enterprise Management Center (EMC) or 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.

4.7.3 Add KAAJEE-specific Logger Tags

Add the following 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 4.7-2 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:

```
.
.
.

  <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>
.
.
.
```

Figure 4.7-2. Sample excerpt of the mylog4j.xml file—Adding KAAJEE logger information

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 HealtheVet-VistA Web-based applications that use KAAJEE.