Sun Java System Application Server Enterprise Edition 8.2 Release Notes for Microsoft Windows

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819–5689–10
February 2007

070330@17039

◆ ◆ ◆ CHAPTER 1

Overview

The Sun Java™ System Application Server Enterprise Edition 8.2 product simplifies the task of creating and administering-J2EE™ applications and web services. This server provides improved performance, clustering, and high availability features for scalable services that continue to operate despite software and hardware faults.

About These Notes

These Release Notes contain important information available at the time of release of Sun Java System Application Server 8.2. Enhancements, known problems, and other late-breaking issues are addressed here. Read this document before you begin using Application Server Enterprise Edition8.2.

The most recent version of these release notes can be found at the Sun Java System documentation Web site (`http://docs.sun.com/db/prod/s1appsrv#hic/`). Check the Web site prior to installing and setting up your software and then periodically thereafter to view the most recent release notes and product documentation.

Third-party URLs are referenced in this document and provide additional related information.

Note – Sun is not responsible for the availability of third-party Web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Accessibility Features

To obtain accessibility features that have been released since the publishing of this media, consult Section 508 product assessments available from Sun on request to determine which versions are best suited for deploying accessible solutions. Updated versions of applications can be found at `http://sun.com/software/javaenterprisesystem/get.html`.

For information on Sun's commitment to accessibility, see `http://sun.com/access`.

Related Documentation

The Application Server Enterprise Edition 8.2 product includes an entire set of documentation that can be found at `http://docs.sun.com/app/docs/prod/sjs.asse#hic`.

The following table summarizes the books included in the Application Server Enterprise Edition 8.2 documentation set.

TABLE 1–1 Books in This Documentation Set

Book Title	Description
Sun Java System Application Server Enterprise Edition 8.2 Documentation Center	One-stop location to access all Application Server topics.
Sun Java System Application Server Enterprise Edition 8.2 Quick Start Guide	How to get started with the Sun Java System Application Server product.
Sun Java System Application Server Enterprise Edition 8.2 Installation Guide[1]	Installing the Sun Java System Application Server software and its components.

TABLE 1–1 Books in This Documentation Set *(Continued)*

Book Title	Description
Sun Java System Application Server Enterprise Edition 8.2 Deployment Planning Guide	Evaluating your system needs and enterprise to ensure that you deploy Sun Java System Application Server in a manner that best suits your site. General issues and concerns that you must be aware of when deploying an application server are also discussed.
Sun Java System Application Server Enterprise Edition 8.2 Developer's Guide	Creating and implementing Java 2 Platform, Enterprise Edition (J2EE™ platform) applications intended to run on the Sun Java System Application Server that follow the open Java standards model for J2EE components and APIs. Includes general information about developer tools, security, assembly, deployment, debugging, and creating lifecycle modules.
Sun Java System Application Server Enterprise Edition 8.2 J2EE Tutorial	Using J2EE 1.4 platform technologies and APIs to develop J2EE applications and deploying the applications on the Sun Java System Application Server.
Sun Java System Application Server Enterprise Edition 8.2 Administration Guide	Configuring, managing, and deploying the Sun Java System Application Server subsystems and components from the Administration Console.
Sun Java System Application Server Enterprise Edition 8.2 High Availability Administration Guide	Postinstallation configuration and administration instructions for the high-availability database.
Sun Java System Application Server Enterprise Edition 8.2 Administration Reference	Editing the Sun Java System Application Server configuration file, `domain.xml`.
Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide	Migrating your applications to the new Sun Java System Application Server programming model, specifically from Application Server 6 and compatible versions and 7. This guide also describes differences between adjacent product releases and configuration options that can result in incompatibility with the product specifications.
Sun Java System Application Server Enterprise Edition 8.2 Performance Tuning Guide	Tuning the Sun Java System Application Server to improve performance.
Sun Java System Application Server Enterprise Edition 8.2 Troubleshooting Guide	Solving Sun Java System Application Server problems.
Sun Java System Application Server Enterprise Edition 8.2 Error Message Reference	Solving Sun Java System Application Server error messages.
Sun Java System Application Server Enterprise Edition 8.2 Reference Manual	Utility commands available with the Sun Java System Application Server, written in manpages style. Includes the `asadmin` command-line interface.

TABLE 1–1 Books in This Documentation Set *(Continued)*

Book Title	Description
1 The *Sun Java System Application Server Enterprise Edition 8.2 Installation Guide* is provided for a standalone installation of the Application Server.	

How to Report Problems and Provide Feedback

If you have problems with Sun Java System Application Server, contact Sun customer support by using one of the following mechanisms:

- Feedback Submittal form (`http://java.sun.com/docs/forms/J2EE14SubmittalForm.html`) – A form for submitting feedback on the Application Server product.
- J2EE-INTEREST list (`http://archives.java.sun.com/archives/j2ee-interest.html`) – A mailing list for J2EE questions.
- Bug database on Java Developer Connection (`http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml`) – To view bugs or to submit a bug, use the Java Developer Connection Bug Parade.
- Java Technology Forums (`http://forum.java.sun.com/`) – An interactive message board for sharing knowledge and questions about Java technologies and programming techniques; use the J2EE SDK forum for discussions related to the Sun Java System Application Server Enterprise Edition 8.2 product.
- Sun Software Support services (`http://www.sun.com/service/sunone/software`) – Links to the Knowledge Base, Online Support Center, and Product Tracker, as well as to maintenance programs and support contact numbers.
- The telephone dispatch number associated with your maintenance contract so that Sun can best assist you in resolving problems, please have the following information available when you contact support:
  - Description of the problem, including the situation where the problem occurs and its impact on your operation
  - Machine type, operating system version, and product version, including any patches and other software that might be affecting the problem
  - Detailed steps on the methods you have used to reproduce the problem
  - Any error logs or core dumps

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to `http://docs.sun.com` and click Send Comments. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document. For example, the title of this book is *Sun Java System Application Server Enterprise Edition 8.2 Release Notes*, and the part number is 819-4728.

Additional Sun Resources

Useful information can be found at the following locations:

- Application Server product information (`http://wwws.sun.com/software/products/appsrvr/home_appsrvr.html`)
- Application Server product documentation (`http://docs.sun.com/db/prod/s1appsrv#hic/`)
- Sun Java System documentation (`http://docs.sun.com/prod/java.sys`)
- Sun Java System Professional Services (`http://www.sun.com/service/sunps/sunone`)
- Sun Java System Software Products and Service (`http://www.sun.com/software`)
- Sun Java System Software Support Services (`http://www.sun.com/service/sunone/software`)
- Sun Java System Support and Knowledge Base (`http://www.sun.com/service/support/software`)
- Sun Support and Training Services (`http://training.sun.com`)
- Sun Java System Consulting and Professional Services (`http://www.sun.com/service/sunps/sunone`)
- Sun Java System developer information (`http://developers.sun.com`)
- Sun Developer Support Services (`http://www.sun.com/developers/support`)
- Sun Java System Software Training (`http://www.sun.com/software/training`)
- Sun software data sheets (`http://wwws.sun.com/software`)
- Sun Microsystems product documentation (`http://docs.sun.com/`)

◆ ◆ ◆ CHAPTER 2

About Application Server Enterprise Edition 8.2

The Sun Java System Application Server Enterprise Edition 8.2 is a J2EE 1.4 platform-compatible server for the development and deployment of J2EE applications and Java technology-based web services in large-scale production environments.

This chapter includes the following sections:

What's New in the 8.2 Release

The Application Server Enterprise Edition 8.2 includes the following enhancements:

- **Improved Administration** – The Application Server supports the remote secure management of complex multisystem enterprise deployments by using either a browser-based console or a scriptable command-line interface. The Application Server also provides a rich JMX based API allowing remote, secure, programmatic access to administrative and monitoring functions.
- **Message Broker** – The Application Server includes an integrated enterprise-class message broker that provides highly available, reliable, high-performance, and scalable messaging.
- **Message Queue 3.7 UR 1** – The Application Server now implements MQ 3.7 UR 1.

- **Expanded Platform Support** – Additional operating systems, databases, locales, and hardware are supported.
- **Sun Java Enterprise System** – As a key component of the Sun Java Enterprise System, the Application Server is tightly integrated with portal and network identity services.
- **Migration and Upgrade Tools** – These tools enable you to verify J2EE applications for standards conformance and portability, help with migrations from other J2EE Application Servers (JBoss, WebLogic, WebSphere), and aid in upgrading from previous versions of Sun ONE Application Server/ iPlanet Application Server.
- **Java 2 Standard Edition 5.0 Support** – The Application Server supports the Java 2 Standard Edition 5.0, which includes enhanced management and monitoring features and many performance and scalability improvements.
- **Java Web Services Developer Pack 1.6 (JWDSP) Plugin Support** – All JWSDP plug-ins are now supported. The JWSDP 1.6 can be downloaded for free from `http://java.sun.com/webservices/downloads/1.6/index.html`.
- **Java DB Database Support** – Application Server includes the Java DB database, based onApache Derby (`http://db.apache.org/derby/`). Backward compatibility with Pointbase database is maintained, but any new databases created on the server will use Java DB by default. After an upgrade from Application Server 8.1 PE or 8.1 EE, existing domains will continue to use Pointbase database, but any new domain created after the upgrade will use Java DB.
- **JDBC Drivers** – The Application Server ships with Sun JDBC drivers.
- **Web Services Security** – These container message security mechanisms implement message-level authentication (for example, XML digital signature and encryption) of SOAP web services invocations by using the X509 and username and password profiles of the OASIS WS-Security standard.
- **WS-I Basic Profile 1.1** – As mandated by the J2EE 1.4 specification, this release implements Web Services Interoperability (WS-I) Basic Profile 1.1 to enable interoperability for web services applications.
- **Backend Connectivity with iWay Adapters** – Sun Microsystems now resells and supports 22 iWay adapters to key backend systems (SAP, Siebel, Oracle, CICS, and IBM MQ Series) to help you leverage existing IT applications from within the Application Server environment. These adapters support the J2EE Connector Architecture 1.5 specification and Web services (SOAP) standards, and include developer tools to reduce time to connect to backend applications.
- **Latest HADB Management System** – The UNIX® platforms contain the new high-availability database (HADB) management system (HADB version 4.4.2–7), which includes a database server, ODBC 2.5 driver, JDBC 3.0 type 4 driver, `clusql` (an interactive program to enter and execute SQL statements), and a management system. This version eliminates the dependency on SSH or RSH, but requires that the network be configured for UDP multicast. See the *Sun Java System Application Server Enterprise Edition 8.2 High Availability Administration Guide* for the details about HADB requirements and limitations.

- **Dynamic Content Technology Support Discontinued** – Dynamic content technologies, such as CGI and SHTML, are no longer supported.

Hardware and Software Requirements

This section lists the requirements that must be met before installing the Sun Java System Application Server Enterprise Edition 8.2 product.

Platform Requirements

The following table lists the operating systems that are supported for Sun Java System Application Server Enterprise Edition 8.2 product. Additionally, the minimum and recommended memory requirements are identified for installing and running the Application Server.

TABLE 2–1 Sun Java System Application Server 8.2 Platform Requirements

Operating System	Minimum Memory	Recommended Memory	Minimum Disk Space	Recommended Disk Space	JVM[1]
Windows 2000 Advanced Server SP4 Windows XP SP2 Windows 2003 Enterprise Server SP1 (32 bit) Windows 2003 Enterprise Server SP1 (64 bit)	1 GB	2 GB	500 MB free	1 GB free	J2SE 1.4.2_10, J2SE 5_06

[1] Only 32–bit (rather than 64–bit) JVMs are supported.

Note – You must use the NTFS file system rather than FAT or FAT32 when running the Application Server on any Microsoft Windows platform.

Important Patch Information

For the current list of required patches for Sun Java System Application Server Enterprise Edition 8.2, see `http://sunsolve.sun.com` and search for "`app server 8.1 patch`." Follow the Sun Java System Application Server Enterprise Edition 8.2 links. As operating system patch requirements change and patches to Java Enterprise System components become available, updates will be made available on the SunSolve℠ web site, initially in the form of recommended patch clusters.

JDBC Drivers and Databases

The Sun Java System Application Server is designed to support connectivity to any DBMS with a corresponding JDBC driver. For a list of components that Sun has tested and found to be acceptable for constructing J2EE compatible database configurations, refer to the following table.

TABLE 2–2 J2EECompatible JDBC Drivers

JDBC Vendor	JDBC Driver Type	Supported Database Server
i-net Software	Type 4	Oracle® 8.1.7, 9i, 9.2.0.3+, 10.1.*x*, 10.2.*x* Sybase ASE 12.5. Microsoft SQL Server 2000 4.0 Service Pack 1
IBM	Type 2	IBM DB2 8.1 Service Pack 3+
Java DB	Type 4	Apache Derby 10.1.2.1
PointBase	Type 4	PointBase Network Server 5.2
DataDirect	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3+, 10.1.*x*, 10.2.*x* Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
MySQL	Type 4	5.*x*
Sun Java System JDBC Driver for Oracle	Type 4	Oracle (R) 9.2.0.3, 10G
Sun Java System JDBC Driver for DB2	Type 4	IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Sybase	Type 4	Sybase ASE 12.5.2

TABLE 2–2 J2EECompatible JDBC Drivers *(Continued)*

JDBC Vendor	JDBC Driver Type	Supported Database Server
Sun Java System JDBC Driver for Microsoft SQL Server	Type 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	Type 4, Type 2	Oracle (R) 9.2.0.3, 10G

Using the Bundled Java DB Database

This section provides instructions for using the Java DB database implementation bundled with Application Server 8.2.

- "Starting and Stopping the Java DB Database" on page 13
- "Java DB Utility Scripts" on page 13

Starting and Stopping the Java DB Database

Sun Java System Application Server 8.2 introduces two new `asadmin` commands for starting and stopping the Java DB Network Server.

- The `start-database` command starts an instance of the Java DB network server:

  ```
  start-database [--dbhost 0.0.0.0] [--dbport 1527] [--dbhome "path/derby"]
  ```

  The default value for the host is `0.0.0.0`, which allows for Java DB to listen on `localhost` as well as the IP and hostname interfaces. The value for the `dbhome` property is the location of the Java DB databases. The default *path* is *appserver_install_dir*`/derby`.

- The `asadmin stop-database` command is used to shut down an instance of the Java DB network server that is running:

  ```
  stop-database [--dbhost 0.0.0.0] [--dbport 1527]
  ```

Java DB Utility Scripts

The Java DB configuration that ships with Application Server 8.2 also includes several useful scripts that can help you use Java DB. The following scripts are available for use in the *appserver_install_dir*`/derby/frameworks/NetworkServer/bin` directory:

- `startNetworkServer.ksh/bat` – Script to start the network server
- `stopNetworkServer.ksh/bat` – Script to stop the network server
- `ij.ksh/bat` – interactive JDBC scripting tool
- `dblook.ksh/bat` – Script to view all or part of the DDL for a database
- `sysinfo.ksh/bat` – Script to display versioning information regarding the Java DB environment

- `NetworkServerControl.ksh/bat` – Script that provides a means of running commands on the `NetworkServerControl` API

▼ To Configure Your Environment to Run the Java DB Utility Scripts

1. **Set the `DERBY_INSTALL` environment variable to point to the *appserver_install_dir*`/derby` directory.**
2. **Unset your `CLASSPATH` environment variable.**
3. **(Optional) You can also set the following properties:**
   - Set `DERBY_SERVER_HOST` to the host on which the network server will listen.

     Can also be set to `0.0.0.0` to enable all listeners.
   - Set `DERBY_SERVER_PORT` to the port number on which the network server will listen.

See Also For more information about these utilities, see the Derby Tools (`http://db.apache.org/derby/docs/10.1/tools/`) and Admin (`http://db.apache.org/derby/docs/10.1/adminguide/`) guides.

Web Servers

This section lists the web servers that are supported for the Sun Java System Application Server Enterprise Edition 8.2.

TABLE 2–3 Supported Web Servers

Web Server	Version	Operating System
Sun Java System Web Server	7	Windows 2000 Advanced Server SP4
		Windows XP SP2
		Windows 2003 Enterprise Server SP1 (32-bit)
		Windows 2003 Enterprise Server SP1 (64-bit)

Browsers

This section lists the browsers that are supported by the Sun Java System Application Server Enterprise Edition 8.2.

TABLE 2–4 Supported Web Browsers

Browser	Version
Mozilla	1.4, 1.5, 1.6, 1.7.*x*
Netscape™ Navigator	4.79, 6.2, 7.0, ***
Internet Explorer	5.5 Service Pack 2, 6.0
Firefox	***

HADB Requirements and Supported Platforms

In addition to the requirements listed in "Hardware and Software Requirements" on page 11, verify that your system meets the following requirements for running HADB.

Note – The Java components of the system have been built with JDK 1.4.2_02 and the system has been tested on JDK 1.5.

Supported Platforms

- **Microsoft Windows** – Microsoft Windows 2000 Advanced Server Service Pack 4 and Microsoft Windows 2003 Enterprise Edition. Note that HADB does not support any of the forthcoming Microsoft Windows operating system versions in 64–bit mode.

HADB Server Host Requirements

- **Minimum memory** – 512 MBytes per node.
- **Minimum free disk space** – 70 MBytes for HADB binaries per host. In addition, disk space is needed for the data devices: 512 MBytes for a test installation per node.
- **Recommended memory** – 1 GByte per node.
- **Recommended free disk space** – 70 MBytes for HADB binaries per host. In addition, disk space is needed for the data devices: 1200 MBytes for a test installation per node.

HADB Management Host Requirements

- **Minimum memory** – 128 MBytes
- **Minimum free disk space** – 70 MBytes for HADB binaries per node

HADB Client Host Requirements

- **Minimum memory** – 120 MBytes
- **Minimum free disk space** – 20 MBytes

Bugs Fixed in the Enterprise Edition 8.2 Release

This section lists the customer-escalated issues resolved for the Sun Java System Application Server Enterprise Edition 8.2 product.

Bug Number	Description
4887079	Programmatic APIs for deploying or undeploying and finding out which applications are deployed.
4911462	Incorrect message when port outside of available range.
4918535	`sun-appserv-deploy()` doesn't take flag to support `createAndDropTables()`.
4939749	`xml:lang()` value should not be inserted automatically by the Deploytool.
4946914	Deployment support for cluster.
4979136	Directory-based deployment copies the application in a backup directory.
4987274	Deployment fails if remote interface for the bean is named `Util()`.
4988818	Transparent Persistence runtime tests fail when J2SE 1.5 is used.
4992295	Deploying a system component succeeds on the command-line interface, but an error is logged in server's log file.
4994790	JSP deployed with *precompilejsp=true* does not use compiler flags in `sun-web.xml`.
4996876	Verifier versus deploy with *verify=true*, different reports.
5003356	Recent `server.policy` file updates are not accounted by Upgradetool.
5006854	`asadmin deploy --virtualservers` fails to deploy.
5007309	Inappropriate default value for HTTP listener accepter threads.
5008941	JSR88 start operation fails when an application is deployed again after it is undeployed.
5016848	On Windows, the JDK JAR file caching and open files prevent some redeployments.
5017956	`list -m` at the JAR module level does not list the EJBs.
5030425	The `deploydir` command ignores `security-role-mapping` changes.

Bug Number	Description
5041343	Not checking `servlet-mapping` `url-pattern--directory` always wrapped by /.
5046120	SEVERE log messages while deploying big applications.
6041268	No mechanism to disable `HTTP TRACE`.
6062410	Upgradetool is launched in English on a localized machine.
6067341	The `deploydir` command on a web application with `ejb-refs` fails on `rmic` the remote interfaces.
6152752	An `outofbound` exception is logged during SPEC J2004 test runs.
6154949	Connection Validation does not work.
6157310	Runtime reloads the `Collection` field during relationship management.
6165491	Failed to start a domain if it was created on a different path than the default domain.
6171667	The lifecycle modules property elements are not created in `domain.xml`.
6171729	Non-String `RA ActivationSpec` properties result in an `IllegalArgumentException` during MDB deployment.
6172178	OSS/J TT TCK failed to get JMS connection factory from a remote application server.
6172589	Optimize calls to security manager.
6183492	[DataDirect] DB2: Some transparent persistence application server tests failed with exception thrown during EJB invocation.
6184864	EJB QL query returns no results using OR operator. Expression contains null single-valued CMRS.
6197393	The Deploytool often won't create message-destination element in deployment descriptor.
6198796	EE samples `asadmin` commands needs to include `availabilityenabled=true()` option when deploying the application.
6198981	Missing `xalan.jar` file from classpath causes empty drop-downs and failure of web service wizard.
6199076	Unable to run the Duke bookstore sample failover test with the `asant` script.
6202363	Cluster name hard coded in one `ant` target in `mq-failover` sample application.
6202606	JMS service configuration cannot be used for SSL JMS between JMS and Message Queue.
6206176	Application Server 8.1 requires `startserv` and `stopserv` to have permissions of 755.

Bug Number	Description
6207297	Accessing the Application Server without the default SSL Port number (443) does not work.
6207862	`asadmin create-domain --help` produces some garbled text.
6363339	`managementws` sample needs to update `MANIFEST.MF` references from `castor-0.9.3.9-xml.jar` to `castor-0.9.9.1.jar`.
6372759	Specific Java System Properties are not handled correctly by AS v8.2 Startup.

Additional HADB Information

This section describes important additional information about the HADB implementation that is included in Application Server 8.2.

HADB Enhancements

- A new management command, `hadbm setadminpassword`, has been implemented to allow changing the password used for database administration. The command takes options indicating which management agent to use, and the old password and the new password. For more information, see the `hadbm setadminpassword` man page.
- The existing management command, `hadbm listpackages`, has been modified. Previously, the command took no operands, and listed all packages in the relevant management domain. The modifications introduce an optional package name operand, and list only packages with that name. If the operand is not provided, all packages are listed. For more information, see the `hadbm listpackages` manpage.
- The existing management command, `hadbm createdomain`, has been modified. The *hostlist* operand is extended to also specify the port number of the management agent. In this way, the domain is completely specified by using only the *hostlist* operand. The old behavior is still supported for backward compatibility. For more information, see the `hadbm createdomain` manpage.
- Some of the error messages from the management system have been modified. The modifications are intended to improve comprehension, consistency, and accuracy of the error messages. The actual modifications are not listed in these release notes.
- The installation and uninstallation behavior has been slightly changed. Installing or uninstalling the HADB should always preserve the link `/opt/SUNWhadb/4`, but this has not always been the case.

- The possibility of typing passwords on the command line as a command option is deprecated. This deprecation is relevant to all `hadbm` commands that accept passwords as command-line options. For `hadbm` commands, it has previously been possible to type a password as:

  1. A password file
  2. A command line option
  3. An interactive input

  The command-line option is considered unsafe and is therefore deprecated. A warning message is issued if a password is typed in this way. Instead, use a password file or interactive output. Note this applies to all `hadbm` commands that accept a command-line password option.
- HADB has been upgraded to use JGroups Version 2.2, and its source code is distributed along with the HADB. To support online upgrade from a previous HADB version, both JGroups 2.1 and 2.2 are delivered with HADB. For JGroups 2.1, byte code is delivered only.

Known SQL Limitations

- You cannot create a `UNIQUE` secondary index on a table.
- The expression `(DISTINCT column)` is not allowed in an aggregate expression, unless this is the only selected expression.
- All tables must be created with a primary key specification. That is, tables without primary keys are not supported.
- `FULL OUTER JOIN` is not supported.
- `IN` subqueries that are table subqueries are not supported, for example:

```
SELECT SNAME FROM S WHERE (S1#,S2#) IN (SELECT S1#,S2# FROM SP
WHERE P#='P2')
```

- Constraints other than `NOT NULL` and `PRIMARY KEY` are not supported.
- You can assign a new owner to a resource. When making this change, however, privileges granted to the current owner are not granted to the new owner.
- Queries with two or more nested `NOT EXISTS` subqueries where each subquery is not directly correlated to outer level of queries is not supported.
- Column privileges are not supported.
- Row value constructors are allowed only in a `VALUES` clause.
- Subqueries are not accepted as value expressions in row value constructors.
- The following data types cannot be used when creating primary keys:
  - `REAL`
  - `FLOAT`

- DOUBLE PRECISION
- DECIMAL
- NUMERIC

High-Availability Load Balancing

The Application Server includes load balancing for the following:

- HTTP, IIOP, and JMS clients
- HTTP session failover support
- EJB clustering and failover support
- Highly available EJB timers
- Distributed transaction recovery
- Support for rolling application upgrades
- High-availability database for storing the transient state of J2EE applications

Availability allows for failover protection of Application Server instances in a cluster. If one Application Server instance fails, another Application Server instance takes over the sessions that were assigned to the unavailable server. Session information is stored in the HADB. HADB supports the persistence of HTTP sessions, Stateful Session Beans, and Single Sign On credentials.

Application Server Product Releases

The Application Server product is delivered in various ways. The following table identifies the delivery mechanisms by product release.

Application Server Product Release	Delivery Mechanism
Application Server Enterprise Edition component within the Sun Java Enterprise System.	File-based distribution Patch installation needed through SunSolve
Application ServerStandard and Enterprise Edition Standalone product	File-based and package-based distribution

Compatibility Issues

Deploytool

Deploytool is no longer be available. The equivalent functionality is available in the NetBeans IDE. For more information and to plan a migration, please see J2EE 1.4 tutorial for NetBeans 4.1 at `http://www.netbeans.org/kb/41/j2ee-tut/index.html`.

Verifier

- Verifier GUI mode (invoked by `verifier -u`) is no longer be available. The equivalent functionality will be available in the NetBeans IDE.
- The default mode for application verification when using verifier tool has been changed from "Verify J2EE Rules" to "Verify J2EE Rules and Sun Application Server Configuration Rules." In other words, by default verifier tests whether an application meets J2EE rules and is configured to run on Sun Application Server. The verifier command contains a command-line switch to test an application for J2EE rules only.

Classloader Changes

In the current release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` attributes of `domain.xml` (application server configuration file) are available in the JVM system classpath. An application depending on this behavior might be using the following methods from the class `java.lang.ClassLoader` to access classes or other resources from the JVM system classpath:

- `getSystemClassLoader()`
- `getSystemResource()`
- `getSystemResourceAsStream()`
- `getSystemResources`

In the next major release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` will no longer be available in the JVM system classpath. If an application uses one of the previously mentioned methods, Sun strongly recommends using an equivalent method that does not assume that the resources are available in the system classpath. The equivalent methods that do not rely on the JVM system classpath are available in `java.lang.ClassLoader` and should be used when possible, for example:

EXAMPLE 2–1 Old Code

```
java.net.URL url = ClassLoader.getSystemResource
("com/acme/tools/tools.properties");
```

EXAMPLE 2–2 Suggested Change

```
java.net.URL url = this.getClass().getClassLoader().getResource
("com/acme/tools/tools.properties");
```

If you cannot change the code, then you can choose to use a new configuration option that will be added in the next release to set the JVM system classpath.

Web Service Security Configuration

Security for Web services can be configured by using the files `wss-client-config.xml` and `wss-server-config.xml`. Note that the content and names of these configuration files are unstable and likely to change. The equivalent functionality will continue to be available.

J2EE Support

The Sun Java System Application Server Enterprise Edition 8.2 supports the J2EE 1.4 platform. The following table describes the enhanced APIs available on the J2EE 1.4 platform.

TABLE 2–5 APIs Available on the J2EE 1.4 Platform

API	Description
Components	
Application and Application Client	Implementation of standard deployment descriptors by means of XML schemas.
Enterprise JavaBeans (EJB) 2.1	Timer service and EJB Web-service endpoint.
Java Servlet 2.4	Web-service endpoint filter.
JavaServer Pages (JSP) 2.0 architecture	Expression language and tag library.
J2EE Connector Architecture 1.5	Inbound resource adapter and Java Message Service (JMS) pluggability.
Web Services	
Java Web Services Developer Pack 1.5	Integrated toolkit for building, testing, and deploying XML applications, Web services, and Web applications.
Java API for XML-based Remote Procedure Calls (JAX-RPC) 1.1	Mapping for WSDL and Java technology and support for development of Web-service clients and endpoints.
WS-I Basic Profile 1.0	The enabling element for interoperability by using WSDL and SOAP.
SOAP with attachment API for Java (SAAJ) 1.2	An API for SOAP-based messaging. Fosters the creation of SOAP messages with attachments.
Java APIs for XML Registries (JAXR) 1.0	A uniform and standard API for accessing XML registries, such as those for Universal Description Discovery and Integration (UDDI and ebXML).
Other	
J2EE Deployment 1.1	Standard APIs that enable deployments of J2EE components and applications.

TABLE 2–5 APIs Available on the J2EE 1.4 Platform *(Continued)*

API	Description
J2EE Management 1.0	Definitions for the information model for managing the J2EE platform.
Java Management Extensions (JMX) 1.2	Standard management API.
Java Authorization Contract for Containers (JACC) 1.0	Definitions of security contracts between a J2EE Application Server and the authorization policy provider.
Java API for XML Processing (JAXP) 1.2	An API with which applications can parse and transform XML documents. Also adds support for processing of XML schemas.
JMS 1.1	A messaging standard that enables J2EE application components to create, send, receive, and read messages. Also adds support for uniform APIs for queues and topics.
JavaMail 1.3	A set of abstract classes that model a mail system. Also includes minor updates to the APIs.

Switching to Another Supported Java Version

Sun Java System Application Server 8.2 requires J2SE 5.0 or or a compatible version as the underlying JVM. If you want to switch from one Java version to another, perform the following general steps for both Windows and Unix OS.

▼ To Switch to Another Supported Java Version

1 **Download the Java SDK (not the JRE) and install it on your system, if you have not already done so.**

The Java SDK can be downloaded from `http://java.sun.com/j2se`.

2 **Completely stop the Application Server.**

You can use the following command line:

```
ApplicationServer-base\bin\asadmin.bat stop-domain
```

Alternatively, you can use the Administration Console GUI:

a. **Click the Application Server node.**

b. **Click** *Stop Instance*.

3 **Edit the** `install_dir\config\asenv.conf` **file (**`asenv.bat` **on Windows), changing the value for** `AS_JAVA` **to point to the new J2SE home directory.**

4. **Edit the `as-install\samples\common.properties` file, changing the line beginning `com.sun.aas.javaRoot...` to reference the new J2SE home directory.**

5. **Restart the Application Server.**

   ```
   ApplcationServer-base\bin\asadmin.bat start-domain
   ```

High Performance

The Application Server includes a high-performance EJB container, web container and services, and supports concurrent message delivery with the Sun Java System Message Queue software.

Scalability

The Application Server supports horizontal scalability through clustering of server instances and requests for load balancing. The Application Server also achieves class leading vertical scalability that supports large multiprocessor machines. The integrated message broker can be clustered for better scalability and availability. Client access from HTTP clients, RMI/IIOP based Rich Client Applications, Web Services Clients, and JRM Clients can be load balanced to Application Server clusters.

JavaServer Faces 1.1 Support

The Sun Java System Application Server Enterprise Edition 8.2 supports JavaServer Faces 1.1 technology. The JavaServer Faces technology consists of a set of server-side APIs that represent user-interface components that manage their state, event, handling, and input validation. The APIs also define page navigation and support internationalization and accessibility. You can add custom user-interface components with a JSP custom tag library.

While developing with JavaServer Faces technology, each member of a development team can focus on a single piece of the process. A simple programming model then links the pieces, resulting in a more efficient and simpler development cycle.

◆ ◆ ◆ CHAPTER 3

Known Issues and Limitations

This chapter describes known problems and associated workarounds for the Sun Java System Application Server Enterprise Edition 8.2 software. If a summary statement does not specify a particular platform, the problem applies to all platforms. This information is organized into the following sections:

Administration

This section describes known administration issues and associated solutions.

Load Balancer Feature Not Supported With Application Server in Configure Automatically During Installation Option (6463858)

The loadbalancer feature is not supported with Application Server in Configure Automatically During Installation option.

Workaround: The loadbalancer feature can be configured after Application Server installation.

Note – You need to have Application Server and Web Server installed on your system to configure the loadbalancer feature.

To configure the loadbalancer feature, follow these steps:

1. Set the value of IS_LB to true and Cfgr_LB to false value in the registry `HKEY_LOCAL_MACHINE -> Sun Microsystem -> EntSys -> Installer -> Application Server`.
2. Change to the `setup` directory.

   ```
   cd JavaES-Install-Dir\setup\
   ```
3. Run the `ASConfigure.bat` batch file.
4. Follow the instructions and provide the appropriate value.

   Note – For AS_LB Plug-in, type Sun Java System Web Server [Mandatory] as this is the only supported plug-in under Java ES 5.

   ---
5. Reboot the system.

`package-appclient` Script Not Working if `domain1` is Missing (ID 6171458)

By default, a hard-coded value is in *JavaES-Install-Dir*`\lib\lib\package-appclient.xml` for the `AS_ACC_CONFIG` variable for `domain1` that is pointed to by `asenv.conf`. If `domain1` is deleted and a new domain created, the `AS_ACC_CONFIG` variable is not updated with the new domain name, which causes the `package-appclient` script to fail.

Solution

Do one of the following:

- Leave `domain1` intact, and create your other domains around it.

- Remove `domain1` and replace the hard-coded value for `domain1` in *JavaES-Install-Dir*`\lib\lib\package-appclient.xml` with the new domain name. Repeat this step every time a new domain is created if `domain1` is not present.

Installing the Load Balancing Plug-in Overwrites an Existing Plug-in (ID 6172977)

If you install the Load Balancing plug-in against an installation of the Application Server that already has a load balancing plug-in installed (for example, from 7.1EE), the 8.2EE plug-in silently replaces any existing load balancer, even if you have created a new server instance in which to run the plug-in.

The plug-in files are installed by default under the `install_dir/plugins/lbplugin` directory, which means that only one version of a plug-in can be used with any one Application Server installation. Note that the console installer does display a message indicating that an uninstallation is being performed, but this message can easily be missed.

Solution

Not everyone will encounter this problem. If you do encounter the problem, remove the old Application Server installation and perform a fresh installation rather than performing an upgrade installation.

Several Changes in `asadmin` Script in Java ES3 Application Server 8.2 compared to Java ES2 Application Server 7 (ID 6189433, 6189436)

Several changes have been made to the `asadmin` command in Application Server 8.2 compared to Application Server 7 and compatible versions. For example, Application Server 7 and compatible versions the command to start a server instance is as follows:

```
asadmin start-instance
```

In version 8.2, the equivalent command is as follows:

```
asadmin start-domain --user admin domain1
```

Refer to the following documents for complete information about the latest `asadmin` command syntax:

- *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide*
- *Sun Java System Application Server Enterprise Edition 8.2 Reference Manual*

- *Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide*

Default Ports Changed in Application Server (ID 6198555)

When upgrading to Java ES5Application Server 8.2 from Java ES2Application Server 7and compatible versions, you might experience incompatibilities or errors because the default ports have changed.

Cannot Restore Backed-up Domain With Another Name. (ID 6196993)

Mirroring of a domain on the same Application Server installation cannot be performed by using the `backup-domain` and `restore-domain` commands because the domain cannot be restored by using a different name than the original, even though the `asadmin restore-domain` command provides an option to rename the domain. Renaming the backed-up domain appears to succeed, but attempts to start the renamed domain fail because the entries in the domain configuration are not changed, and `startserv` and `stopserv` use the original domain name to set paths.

Solution

The domain name used for `restore-domain` must be the same as that used for the original `backup-domain` command. The `backup-domain` and `restore-domain` commands in Application Server 8.2 work only for backing up and restoring the same domain on the same machine.

Starting Application Server With Additional JMX Agent Not Supported (ID 6200011)

J2SE 1.4., 5.0, and compatible versions can be configured on the Application Server. An integral feature of the J2SE 5.0 platform is the ability to start a JMX agent. This agent is activated when you explicitly set system properties at the server startup.

Example values include the following:

```
name="com.sun.management.jmxremote" value="true"
name="com.sun.management.jmxremote.port" value="9999"
name="com.sun.management.jmxremote.authenticate" value="false"
name="com.sun.management.jmxremote.ssl" value="false"
```

After configuring JMX properties and starting the server, a new `jmx-connector` server is started within the Application Server VM. An undesirable side-effect of this is that the administration functions are affected adversely, and the Application Server administration GUI and CLI might produce unexpected results. These results are caused by some conflicts between the built-in `jmx-connector` server and the new `jmx-connector` server.

Solution

If using `jconsole` or any other JMXcompliant client, consider reusing the standard JMX Connector Server that is started atApplication Server startup.

When the server starts, a line similar to the one shown in the next paragraph appears in the `server.log`. You can connect to the `JMXServiceURL` specified there and perform the same management and configuration operations after successfully providing the credentials, for example:

```
[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.
system.tools.admin|_ThreadID=10;|ADM1501: Here is the JMXServiceURL for the
JMXConnectorServer: [service:jmx:rmi:///jndi/rmi://hostname:8686/management/
rmi-jmx-connector]. This is where the remote administrative clients should
connect using the JSR 160 JMX Connectors.|#]
```

For more information, refer to the *Sun Java System Application Server 8.2 Administration Guide*.

Load Balancer Configuration File Not Created With the Endpoint URL of Any Web Service (ID 6236544, 6275436)

When setting up the load balancer configuration with an application that has an EJB module that exports a web service URL, the context root for the web service isnot in the resulting `loadbalancer.xml` file.

Solution

1. Edit the `loadbalancer.xml` file to add the missing web module as follows:

   ```
   <web-module context-root="context-root-name"
   disable-timeout-in-minutes="30" enabled="true"/>
   ```

2. Replace *context-root-name* value with the context root name of the web service that was exposed as an EJB.

Java Home Setting Inside Configuration Not Effective (ID 6240672)

Application Server domains and servers do not use the JDK that is pointed by the `java-home` attribute of `java-config` element of the associated configuration.

Solution

The JDK used by the Application Server processes for all the domains in a given server installation is determined by the `appserver-installation-dir/config/asenv.conf` file. The property `AS_JAVA` in this file determines the JDK used and is set at the time of installation. If a different JDK is to be used by Application Server processes after the installation is completed, this value can be modified to point to another JDK. Note that all domains in this installation are affected by this change.

Note – Manual changes to `asenv.conf` file are not checked for validity, so be cautious while changing them. Check the product documentation for minimum JDK version requirements when modifying the value for `AS_JAVA`.

Application Server Restart Using `sun-appserv-admin` Causes `LoginException` Error (ID 6288893)

This problem is caused by a wrong value for `%CONFIG_HOME%`.

Solution

1. Rename the existing `asant`to `asant.bak`.
2. Copy the `asant.template` file in *as_install*`/lib/install/templates/ee` for SE or EE version to the *as_install*`/bin/` directory and rename the `asant` file.
3. Edit the newly copied *as_install*`/bin/asant` file, replacing the `%CONFIG_HOME%` token with *as_install*`/config` value.
4. If any manual changes were made to the original `asant.bak` file, merge them into the new `asant` file.

`.asadmintruststore` File Not Described in the Application Server Documentation (ID 6315957)

If this file does not exist in the server administrator's `home` directory, you might experience serious bugs when upgrading certain applications hosted on the server.

Solution

- If possible, have the `asadmin start-domain domain1` command run by the user who installed the server.
- If the command is not run by that user, the `.asadmintruststore` should be moved or copied from the `home` directory of the installing user to the `home` directory of the running user.
- Note that if the file is moved (not copied) from the installing user's `home` directory to the running user's `home` directory, you might experience application upgrade problems, as described in bugs 6309079, 6310428 and 6312869. These problems occur because the upgrade or installation user does not have the `.asadminstruststore` file in the `home` directory.

Domain Fails to Start When `create-domain` Master Password Has Special Characters (ID 6345947)

The domain does not start when the domain's master password contains the percent (`%`) character.

Solution

The domain's master password should not contain a percent character (`%`). This solution applies when creating a new domain or changing the master password for an existing domain.

Load Balancer Configuration Changes in `magnus.conf` and `obj.conf` Are Overwritten (ID 6394181)

After you have created a secure `http-listener` and installed `lbplugin`, the `magnus.conf` and `obj.conf` files under the *webserver_instance_dir*`/config` directory are modifiedand the `lbplugin` contents are being removed.

The installer modifies the `magnus.conf` and `obj.conf` configuration files on the Application Server as part of the installation of the load balancer plug-in. If you log in to the Application Server admin console and try to manage the instance configuration for the instance on which the load balancer has been installed, the Application Server issues a warning message stating that it has detected a manual edit in the configuration. This warning is in fact referring to the changes made by the installer.

Solution

Verify that the changes made by the installer have not been overwritten.

Application Client

This section describes known application client issues and associated solutions.

Library JAR Packaged in Application Client Archive Overwrites MANIFEST File (ID 6193556)

If you have a top-level JAR file inside your client JAR (in this case, `reporter.jar`), when you deploy the client JAR, the MANIFEST file for that JAR overwrites the MANIFEST file for the client JAR.

Solution

None.

Dynamic Content Technology Such as CGI-bin and SHTML Functionality Not Supported (ID 6373043)

Dynamic content technologies, such as CGI-bin and SHTML, are no longer supported.

Solution

Use JSP and Web service technologies instead.

Bundled Sun JDBC Drivers

This section describes known bundled Sun JDBC driver issues and associated solutions.

Applications Using the `TRANSACTION_SERIALIZABLE` Isolation Level With the Bundled Sun Driver For Microsoft SQL Server Hangs(ID 6165970)

You could experience this problem if you are using a prepared update statement while two parallel transactions are running and one of them is rolled back.

Solution

Set a isolation level for a connection, create the corresponding connection pool at the same isolation level. For more information about configuring connection pools, see the *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide*.

`PreparedStatement` Errors (ID 6170432)

Description 1

If an application generates more than 3000 `PreparedStatement` objects in one transaction, the following error might occur with DB2:

```
[sunm][DB2 JDBC Driver] No more available statements.Please recreate your
package with a larger dynamicSections value.
```

Solution 1

Add following properties to the connection pool definition to ensure that the driver rebinds DB2 packages with a larger dynamic sections value:

`createDefaultPackage=true replacePackage=true dynamicSections=1000`

See the *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide* for details about configuring connection pools.

Description 2

Related to the `PreparedStatement` error , another error message that might be thrown:

```
[sunm][DB2 JDBC Driver][DB2]Virtual storage or database resource is not
available.
```

Solution 2

Increase the DB2 server configuration parameter *APPLHEAPSZ*. For example, use `4096`.

Description 3

Isolation level `TRANSACTION_SERIALIZABLE`. If your application uses isolation level `TRANSACTION_SERIALIZABLE` and uses one of the parameters suggested previously, it might hang while obtaining a connection.

Solution 3

To set an isolation level for a connection, the corresponding connection pool has to be created at that isolation level. See the *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide* for instructions.

Connectors

This section describes known J2EE connector architecture issues and associated solutions.

After Restarting a DAS Instance, Undeploying he Connector Module Fails When Cascade is Set to `False` (ID 6188343)

In this scenario, a standalone or embedded connector module is deployed in DAS and connector connection pools, and resources are created for the deployed module. After restarting the DAS instance, undeploying the connector module fails when cascade is set to false with the following exception:

```
[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterprise.
system .core|_ThreadID=14;|CORE5023: Error while unloading application [foo]|#]
```

Solution

Restart the DAS instance. Use cascaded undeploy (set the cascade option to true) for undeploying standalone and embedded connectors..

JMS `create-jms-resource`: CLI Sets the Default Values Incorrectly (ID 6294018)

Because you cannot specify the minimum pool size and maximum pool size when creating a new JMS resource from the command line with the `asadmin create-jms-resource` command, the `asadmin` command should create the resource by using the default pool size values (minimum `8`, maximum `32`). Instead, creating the resource from the command line results in default minimum and maximum pool sizes of `1` and `250`, respectively.

Solution

After creating a JMS resource from the command line, use the admin console to modify the minimum and maximum pool size values.

Documentation

This section describes known documentation issues and associated solutions.

Javadoc Inconsistencies

The Javadoc for several AMX interfaces and methods is either missing or incorrect:

- Getter methods for `NumConnAcquired` and `NumConnReleased` statistics are missing from `ConnectorConnectionPoolStats` and `AltJDBCConnectionPoolStats`.
- Calling the following methods in `EJBCacheStats` throws an exception: `getPassivationSuccesses()`, `getExpiredSessionsRemoved()`, `getPassivationErrors()`, `getPassivations()`.
- The AMX MBeans might require several seconds after server startup before they are all registered and available for use.
- The constant `XTypes.CONNNECTOR_CONNECTION_POOL_MONITOR` is misspelled (`"NNN"`).

Bundled ANT Throws `java.lang.NoClassDefFoundError` Exception (ID 6265624)

The following exception is thrown in thread "main" `java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher`.

Solution

Using the bundled ANT for performing the tasks outside the Application Server is not recommended.

Logging Options Documentation Incorrect (ID 6463965)

The *Sun Java System Application Server Enterprise Edition 8.2 Performance Tuning Guide* incorrectly states the following about Log Options:

> The Administration GUI provides the following two logging options:
>
> - *Option 1* – Log `stdout` (`System.out.print`) content to the event log
> - *Option 2* – Log `stderr` (`System.err.print`) content to the event log

These log options no longer exist in Application Server Enterprise Edition 8.2.

Conflicting Information About HTTP File Cache Feature in Application Server 8.2 (ID 6474799)

The Application Server Enterprise Edition 8.2 documentation discusses an HTTP file caching feature, in "HTTP File Cache" in *Sun Java System Application Server Enterprise Edition 8.2 Performance Tuning Guide*. However, this feature was not included in Application Server Enterprise Edition 8.2. Note that this feature has been reintroduced in Application Server 9.0.

High Availability

This section describes known high-availability database (HADB) issues and associated solutions.

`hadbm set` Does Not Check Resource Availability (Disk and Memory Space) (ID 5091280)

When you increase the device or buffer sizes by using `hadbm set`, the management system checks resource availability when creating databases or adding nodes. However, the system does not check if sufficient resources are available when device or main-memory buffer sizes are changed.

Solution

Verify that enough free disk or memory space is available on all hosts before increasing any of the `devicesize` or `buffersize` configuration attributes.

Heterogeneous Paths for `packagepath` Not Supported (ID 5091349)

You cannot register the same software package with the same name with different locations at different hosts. For example:

```
hadbm registerpackage test --packagepath=/var/install1 --hosts europa11
Package successfully registered.
hadbm registerpackage test --packagepath=/var/install2 --hosts europa12
hadbm:Error 22171: A software package has already been registered with
the package name test.
```

Solution

HADB does not support heterogeneous paths across nodes in a database cluster. Make sure that the HADB server installation directory (`--packagepath`) is the same across all participating hosts.

`createdomain` Might Fail (IDs 6173886, 6253132)

If running the management agent on a host with multiple network interfaces, the `createdomain` command might fail if not all network interfaces are on the same subnet:

```
hadbm:Error 22020: The management agents could not establish a
domain, please check that the hosts can communicate with UDP multicast.
```

If not configured, the management agents will (use the "first" interface for UDP multicasts. "First" is defined by the result from `java.net.NetworkInterface.getNetworkInterfaces()`.

Solution

The best solution is to tell the management agent which subnet to use (set `ma.server.mainternal.interfaces` in the configuration file. For example, `ma.server.mainternal.interfaces=10.11.100.0`). Alternatively you might configure the router between the subnets to route multicast packets (the management agent uses multicast address 228.8.8.8).

Before retrying with a new configuration of the management agents, you might have to clean up the management agent repository. Stop all agents in the domain, and delete all files and directories in the repository directory, identified by `repository.dr.path` in the management agent configuration file. This cleanup must be performed on all hosts before restarting the agents with a new configuration file.

Directories Must Be Cleaned Up After Deleting an HADB Instance (ID 6190878)

After deleting an HADB instance, subsequent attempts to create new instances with the `configure-ha-cluster` command fail. The problem is that old directories are left from the original HADB instance in `ha_install_dir/rep/*` and `ha_install_dir/config/hadb/instance_name`.

Solution

Be sure to manually delete these directories after deleting an HADB instance.

`clu_trans_srv` Cannot Be Interrupted (ID 6249685)

A bug in the 64-bit version of Red Hat Enterprise Linux 3.0 forces the `clu_trans_srv` process into an uninterruptible mode when performing asynchronous I/O. This means that `kill -9` command does not work and the operating system must be rebooted.

Solution

Use a 32-bit version of Red Hat Enterprise Linux 3.0.

`hadbm` Does Not Support Passwords Containing Capital Letters (ID 6262824)

Capital letters in passwords are converted to lowercase when the password is stored in `hadb`.

Solution

Do not use passwords containing capital letters.

`hadbm/ma` Produces Faulty Error Message When a Session Object Times Out and Deletes at Management Agent (ID 6275103)

Sometimes a resource contention problem on the server may cause a management client to become disconnected. When reconnecting, the following misleading error message `"hadbm:Error 22184: A password is required to connect to the management agent"` might be returned.

Solution

Check if a resource problem has occurred on the server, take proper action (for example, , add more resources), and retry the operation.

The Management Agent Should Not Use Special-Use Interfaces. (ID 6293912)

Special-use interfaces with IP addresses like `0.0.0.0` should not be registered as valid interfaces to be used for HADB nodes in the Management Agent. Registering such interfaces can cause problems if HADB nodes are set up on these interfaces by means of a user issuing a `hadbm create` command that uses host names instead of IP addresses. The nodes will then be unable to communicate, causing the `create` command to hang.

Solution

When using `hadbm create` on hosts with multiple interfaces, always specify the IP addresses explicitly by using DDN notation.

Reassembly Failures on Windows (ID 6291562)

On the Windows platform, with certain configurations and loads, a large number of reassembly failures might occur in the operating system. The problem has been seen with configurations of more than 20 nodes when running several table scans (`select *`) in parallel. Possible symptoms include transactions aborting frequently, repair or recovery taking a long time to complete, and frequent timeouts occurring in various parts of the system.

Solution

To fix the problem, the Windows registry variable `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` can be set to a value higher than the default `100`. For best results, increase this value to `0x1000` (`4096`). For more information, see article 811003 (`http://support.microsoft.com/default.aspx?scid=kb;en-us;811003`) from the Microsoft support pages.

When Running `hadbm start` *db_name*, Part of Typed Password Displayed Without Being Masked (IDs 6303581, 6346059, 6307497)

When a machine is overloaded, the masking mechanism fails and some characters from the typed password can be exposed. This exposition poses a minor security risk. The password should always be masked.

Solution

Put the passwords in their own password files (the method recommended since Application Server 8.1) and refer to these files with either the `--adminpassword` or `--dbpasswordfile` options.

Installation

This section describes known installation issues and associated solutions.

Java Enterprise System 5 Installer for Minimal Application Server 8.x Load Balancer Does Not Install Properly (ID 6478047)

Apache and IIS cannot be configured through Java ES 5 Installer. You need to manually configure Apache and IIS on the Windows platform.

Solution

To configure Load Balancer Apache or IIS, follow these steps.

To Configure Apache 2.x:

1. Install Apache 2.x.

   Apache is installed in the `APDIR=C:\Apache2\Apache2` directory
2. Install JES5 with minimal installation.

   Deselect all components but loadbalancer. Java ES 5 is installed in the `JES5DIR=C:\Program Files\Sun\JavaES5` directory.

   The
3. Create the `resource` and `errorpages` directories in the `Apache2` directory.

   ```
   mkdir %APDIR%\modules\resource
   ```

   ```
   mkdir %APDIR%\modules\errorpages
   ```
4. Copy the resource file to the `resource` directory.

   ```
   cd %APDIR%\modules\resource
   ```

   ```
   copy %JES5DIR%\appserver\lib\webserver-plugin\windows\apache2\LBPlugin*.res
   .
   ```
5. Copy the load balancer DLL to the `modules` directory.

   ```
   cd %APDIR%\modules
   ```

   ```
   copy
   %JES5DIR%\appserver\lib\webserver-plugin\windows\apache2\mod_loadbalancer.dll
   .
   ```
6. Copy the template `errorpages` to the `errorpages` directory.

   ```
   cd %APDIR%\modules\errprpages
   ```

   ```
   copy %JES5DIR%appserver\lib\webserver-plugin\windows\iws\errorpages .
   ```

7. Copy the loadbalancer template and the other DTD to the Apache `config` directory.

   ```
   cd %APDIR%\config
   copy %JES5DIR%\appserver\lib\install\templates\loadbalancer.xml.template .
   copy %JES5DIR%\appserver\lib\dtds\sun-loadbalancer* .
   ```

8. Create a backup of the `httpd.conf` file.

   ```
   cd %APDIR%\config
   copy httpd.conf httpd.conf.orig
   ```

9. Edit the `httpd.conf` file.

   Append the following lines to the `httpd.conf` file:

   ```
   ##BEGIN EE LB Plugin Parameters
   LoadModule apachelbplugin_module modules/mod_loadbalancer.dll
   <IfModule mod_apache2lbplugin.cpp>
           config-file "C:\Apache2\Apache2/conf/loadbalancer.xml"
           locale en
   </IfModule>
   <VirtualHost 10.12.8.107>
   DocumentRoot "C:\Apache2\Apache2/htdocs"
   ServerName vm07
   </VirtualHost>
   ##END EE LB Plugin Parameters
   ```

10. Replace `C:\Apache2\Apache2` with the actual `%APDIR%` directory.

    Also replace the IP, ServerName, and DocumentRoot directory.

11. Create a new `sec_db_files`directory in the `%APDIR%`.

    ```
    cd %APDIR%
    mkdir sec_db_files
    ```

12. Copy NSS key store to the `%APDIR%\sec_db_files` directory.

    ```
    cd %APDIR%\sec_db_files
    copy %JES5DIR%\appserver\lib\webserver-plugin\windows\iis\*.db .
    ```

13. Set the PATH to include the required libraries.

    Prepend the following extra path:

    ```
    PATH %JES5DIR%\share\lib;%JES5DIR%\appserver\lib;%JES5DIR%\appserver\bin
    ```

14. Replace `%JES5DIR%` with the actual Java ES 5 directory.
15. Add the `NSPR_NATIVE_THREADS_ONLY` variable with value 1 in the system environment.
16. Reboot and test Apache 2 (after configuring `loadbalancer.xml`).

To configure IIS LBPlugin:

1. Create the `sun-passthrough` directory in the `c:\inetpub\wwwroot` directory.

   ```
   cd c:\inetpub\wwwroot
   mkdir sun-passthrough
   ```

2. Create`errorpages`, `resource` and `sec_db_files` directories in the `c:\inetpub\wwwroot\sun-passthrough` directory.

   ```
   cd c:\inetpub\wwwroot\sun-passthrough
   mkdir errorpages
   mkdir resources
   mkdir sec_db_files
   ```

3. Copy DLL files to the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/webserver-plugin/iis/*.dll
   c:\inetpub\wwwroot\sun-passthrough\
   ```

4. Copy DTDs to the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/dtds/sun-loadbalancer*.dtd
   c:\inetpub\wwwroot\sun-passthrough\
   ```

5. Copy the `sun-passthrough.properties` fileto the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/webserver-plugin/iis
   c:\inetpub\wwwroot\sun-passthrough\
   ```

6. Copy security DB files to the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/webserver-plugin/iis/*.db
   c:\inetpub\wwwroot\sun-passthrough\sec_db_files\
   ```

7. Copy resource files to the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/webserver-plugin/iws/*.res
   c:\inetpub\wwwroot\sun-passthrough\resource\
   ```

8. Copy error pages to the `sun-passthrough` directory.

   ```
   copy <as_install_dir>/appserver/lib/webserver-plugin/iws/errorpages/*.html
   c:\inetpub\wwwroot\sun-passthrough\errorpages\
   ```

9. Copy the `loadbalancer.xml.example` template to the `sun-passthrough` directory.

   ```
   copy
   <as_install_dir>/appserver/lib/install/templates/loadbalancer.xml.example
   c:\inetpub\wwwroot\sun-passthrough\
   ```

10. Edit the `sun-passthrough.properties` file.

```
##BEGIN EE LB Plugin Parameters
log-file = C:\InetPub\wwwroot\sun-passthrough\lb.log
### The valid options for different logging levels are FATAL, SEVERE, WARNING, INFO and FINE.
```

```
log-level = INFO
lb-config-file = C:\InetPub\wwwroot\sun-passthrough\loadbalancer.xml
##END EE LB Plugin Parameters
```

Note – If you are configuring IIS6, ensure that you set the permissions and perform additional steps as described in the AS82 documentation. You might also need to set the IIS6 isolation mode to IIS5 compatible.

`imq` Directory Needs to Be Created During Installation. (ID 6199697)

On the Windows platform, immediately after the installation of Application Server Enterprise Edition, the Message Queue broker fails on startup. A error message is displayed stating that the directory `drive:\as\domains\domain1\imq` does not exist.

Note that if the broker is started after starting `domain1`, the directory is created by the Application Server and the problem does not occur.

Solution

1. Create the `var_home_dir_location` before creating the broker:

   ```
   $imqbrokerd -varhome var_home_dir_location
   ```

   For example:

   ```
   $imqbrokerd -varhome D:\as\domains\domain1\imq
   ```

J2EE Tutorial

To run the J2EE 1.4 Tutorial on the Sun Java System Application Server Enterprise Edition 8.2, perform these tasks:

- When you edit the file examples`/common/build.properties` as described in the "About the Examples" section of the "About this Tutorial" chapter, also change port 4848 to 4849.
- When using Deploytool, add the server `localhost:4849` before deploying an example.
- When using the Administration Console to create any resource, use the Targets tab to specify the server as the target. If you use the command line or an `asant` target, the server is the default target, and no further action is required.

Lifecycle Management

This section describes known lifecycle management issues and associated solutions.

Changing the `ejb-timer-service` Property Generates an Error (ID 6193449)

After setting the `ejb-timer-service` property `minimum-delivery-interval` to `9000`, an attempt to set the `ejb-timer-service` property `redelivery-interval-in-mills` to `7000` causes the `set` command to fail. The following error message is displayed:

```
[echo] Doing admin task set
[exec] [Attribute(id=redelivery-interval-internal-in-millis) : Redelivery-
Interval (7,000) should be greater than or equal to Minimum-delivery-
interval-in-millis (9,000)]
[exec] CLI137 Command set failed.
```

- `minimum-delivery-interval` is the minimal interval duration between deliveries of the same periodic timer.
- `redelivery-interval-in-mills` is the time the timer service waits after a failed `ejbTimeout` before attempting redelivery.

  The problem is that the logic that relates the redelivery interval property to the minimum delivery property is incorrect and prevents you from using the GUI or the CLI to set any value where the minimum delivery interval is greater than the redelivery interval.

  The `minimum-delivery-interval-in-millis` must always be set equal to or higher than the `ejb-timer-service` property `redelivery-interval-in-millis`. The problem is that there is an erroneous validation check in the Application Server to verify that the value for `redelivery-interval-in-millis` is greater than the value for `minimum-delivery-interval-in-millis`.

Solution

Use the default values for these properties, as follows:

```
minimum-delivery-interval(default)=7000
redelivery-interval-in-millis(default)=5000
```

Values other than these defaults will generate an error.

Logging

This section describes known logging issues and solutions.

Setting Debug Statement for `access.failure` Causes Hanging in Application Server Startup (ID 6180095)

Setting the `java.security.debug` option for the JVM will cause the server instance startup to freeze with a deadlock; for example, setting the following in `domain.xml` causes the problem:

```
<jvm-options\>-Djava.security.debug=access,failure</jvm-options\>
```

Solution

None. Avoid setting this flag.

Logging Location and Instance Location Have Changed for Java ES3 Application Server (ID 6189409)

Default logging and server instance locations have changed in Sun Java System 8.2 compared to 7 version and compatible versions.

For more information, refer to the *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide* or the *Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide*.

Message Queue

This section describes known Java message queue issues and associated solutions.

JMS Reconnection Not Successfully Completed in Timing-Dependent Cases(IDs 6173308, 6189645, 6198481, 6199510, 6208728)

Failures to reconnect in timing-dependent scenarios can be caused by several problems.

Solution

You can work around these problems by:

- Restarting the brokers involved
- Restarting the Application Server instances involved

Asynchronous Message Listener Behavior Changed in `appclient` From 8.0 to 8.1 Update 2 (ID 6198465)

Due to a recent change, when an asynchronous message listener is the only live thread in the `app-client` container, the remaining `appclient` virtual machine exists as a daemon. This behavior is a regression for older applications that perform asynchronous receives in ACC. This problem affects application clients that set a JMS message listener and exit the main thread.

Solution

Do not exit the main thread. Wait for the message listener to notify the main thread before terminating the main thread.

Monitoring

This section describes known monitoring issues and associated solutions.

Monitoring Framework Integration With Application Server (6469302)

In Application Server beta release, monitoring framework is not supported by default.

Solution

To integrate Monitoring Framework with Application Server, follow these steps:

1. Edit the `<Install_dir>\appserver\lib\install\templates\ee\com.sun.cmm.as.xml` file.

   Update ${InstalledDate} with Application Server installation location and ${InstalledDate} with current date.
2. Copy the `<Install_dir>\appserver\lib\install\templates\ee\com.sun.cmm.as.xml` file to `<Install_dir>\appserver\lib`.
3. Execute the `<MFWK_Install_location>\bin\mfwksetup.bat -r <Install_dir>\appserver\lib\com.sun.cmm.as.xml` command.

Note – ${InstalledLocation} value is the Application Server installation location, `c:\Sun\JavaES5\appserver`. For $InstalledDate you need to put time in milliseconds by calculating the current time in millisecond from 1970.

Samples

This section describes known and associated solutions related to the sample code included with the Application Server 8.2 product.

`setup-one-machine-cluster` Hangs (ID 6195092)

On the Windows platform, the `mqfailover` command requires pressing Control+C keys to exit the hung process. You must rerun the `setup-one-machine-cluster` process.

From *install_dir*`\samples\ee-samples\failover\apps\mqfailover\docs\index.html`, run the following commands:

- Console 1

  ```
  cd install_dir\samples\ee-samples asant start-mq-master-broker1
  ```

- Console 2

  ```
  cd install_dir\samples\ee-samples asant start-mq-cluster-broker1
  ```

- Console 3

  ```
  cd install_dir\samples\ee-samples asant start-mq-cluster-broker2
  ```

- Console 4

  ```
  cd install_dir\samples\ee-samples asadmin start-domain domain1
  ```

If you have already executed `asant setup-one-machine-cluster-without-ha` or `asant setup-one-machine-cluster-with-ha` for any other Enterprise Edition sample, execute `asant configure-mq`. Otherwise, execute `asant setup-one-machine-cluster-and-configure-mq`. In this case, the command appears to succeed:

```
start_nodeagent: [echo] Start the node agent cluster1-nodeagent
[exec] Command start-node-agent executed successfully.
```

But then the system hangs indefinitely.

Solution

None. This problem similarly affects all Enterprise Edition samples that use this ant target on Windows. A workaround is to press Control+C to exit the hung process and then rerun it.

Documentation Not Explicitly States the Need to Create JMS Resources Before Running the Message Queue Failover Sample Application (ID 6198003)

After you complete the `asadmin` deploy instructions and run the Message Queue failover sample application, the following error message is dilsplayed:

```
/opt/SUNWappserver/domains/domain1/config/sun-acc.xml -name
MQFailoverTestClient -textauth -user j2ee -password j2ee
Nov 18, 2004 10:50:17 PM com.sun.enterprise.naming.NamingManagerImpl
bindObjects
SEVERE: NAM0006: JMS Destination object not found: jms/durable/TopicA
Nov 18, 2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl
bindObjects
SEVERE: javax.naming.NameNotFoundException
javax.naming.NameNotFoundException
```

The documentation does not explicitly state that JMS resources must be manually created if manual deployment is performed by using the `asadmin deploy` commands. The documentation also fails to mention that that the provided ant targets should be used to deploy the sample application .

Solution

Use the `asant deploy` target for the `build.xml` script, which creates the required JMS resources to run the application.

Security

This section describes known issues and associated solutions related to Application Server and web application security and certificates.

Cannot Run `WebServiceSecurity` Applications on Enterprise Edition With J2SE 5.0 (ID 6183318)

`WebServiceSecurity` applications cannot run with J2SE 5.0 for the following reasons:

- J2SE 5.0 PKCS11 does not support UNWRAP mode
- J2SE 5.0 PKCS11 does not support `RSA/ECB/OAEPWithSHA1AndMGF1Padding` with PKCS11

Solution

Use J2SE 1.4.2 with any other JCE provider (not the one included by default). Note that hardware accelerator support will not be present in this configuration.

SSL Termination Not Working (ID 6269102)

When Load Balancer (Hardware) is configured for SSL termination, the Application Server changes the protocol from `https` to `http` during redirection.

Solution

Add a software load balancer between the hardware load balancer and the Application Server.

Upgrade Utility

This section describes known upgrade utility issues and associated solutions.

Derby Database Used by Samples Script Created in Wrong Location (ID 6377804)

There are two aspects to this bug:

1. When sample application setup scripts that use the Derby database are run, the Derby database is created in the current directory or in the `<install_root>/bin`.
2. The sample `build` Ant script creates a `password.txt` file that stores the admin password file in the current directory, which will not be writable in nonroot and sparse zones scenarios.

Solution

1. *Derby database location* – Use the `--dbhome` option with the `start-database` command to create the database at the value specified for `--dbhome`. For example, the following is the `asadmin` command syntax for `start-database`.

```
start-database [--dbhost 0.0.0.0] [--dbport 1527] [--dbhome db_directory] [--echo=false]
[--verbose=false]
```

2. *Location of the* `password.txt` *file* – By design, the samples directory is expected to be writable because all the build commands include the creation of a `password.txt` file in that directory. Be sure to install a working copy of the samples in a writable location.

Failure to Invalidate Semicolon (;) Character in Admin User Name or Password (ID 6473341)

The Application Server Enterprise Edition 8.2 installation does not allow special characters in the admin user name. Domain creation fails if any special characters are used. Note, however, that the admin password can have special characters.

Solution

When upgrading from Application Server 7 to Application Server 8.2, verify that the admin user name does not contain any special characters.

Web Container

This section describes known web container issues and associated solutions.

No Support For Apache and IIS for Load Balancer Plug-in

Sun Java ES 5 Application Server does not support Apache and IIS (non-Sun web container) for load balancer plug-in. Sun Java ES installs Sun Java System Web Server for load balancer plug-in configuration.

Deployment of Application Using `--precompilejsp=true` Can Lock JAR Files (ID 5004315)

On Windows platform, if you request precompilation of JSPs when you deploy an application on Windows, later attempts to undeploy that application or to redeploy it (or any application with the same module ID) will not work as expected. JSP precompilation opens JAR files in your application but does not close them, and Windows prevents the undeployment from deleting those files or the redeployment from overwriting them.

Note that undeployment succeeds to a point, in that the application is logically removed from the Application Server. Also note that no error message is returned by the `asadmin` utility, but the application's directory and the locked jar files remain on the server. The server's log file will contain messages describing the failure to delete the files and the application's directory.

Attempts to redeploy the application after undeploying fail because the server tries to remove the existing files and directory, and these attempts also fail. This scenario can happen if you try to deploy any application that uses the same module ID as the originally deployed application, because the server uses the module ID in choosing a directory name to hold the application's files.

Attempts to redeploy the application without undeploying it first will fail for the same reasons.

Diagnostics

If you attempt to redeploy the application or deploy it after undeploying it, the `asadmin` utility returns an error similar to the following:

```
An exception occurred while running the command. The exception
message is: CLI171 Command deploy failed : Deploying application in
domain failed; Cannot deploy. Module directory is locked and can't
be deleted.
```

Solution

If you specify `--precompilejsps=false` (the default setting) when you deploy an application, this problem does not occur. Be aware that the first use of the application triggers the JSP compilation, so the response time to the first request is longer than for later requests.

Note also that if you do precompile, you should stop and restart the server before undeploying or redeploying the application. The shutdown frees the locked JAR files so the undeployment or redeployment after the restart can succeed.

Unable to Deploy WAR with Servlet 2.4-Based `web.xml` That Contains Empty `<load-on-startup>` Element (ID 6172006)

The optional load-on-startup servlet element in a `web.xml` file indicates that the associated servlet is to be loaded and initialized as part of the startup of the web application that declares it.

The optional content of this element is an integer indicating the order in which the servlet is to be loaded and initialized with respect to the web application's other servlets. An empty `<load-on-startup>` element indicates that the order is irrelevant, as long as the servlet is loaded and initialized during the startup of its containing web application.

The Servlet 2.4 schema for `web.xml` no longer supports an empty `<load-on-startup>`, meaning that an integer must be specified when using a Servlet 2.4 based `web.xml` file. If specifying an empty `<load-on-startup>`, as in `<load-on-startup/>`, the `web.xml` file fails validation against the Servlet 2.4 schema for `web.xml`, causing deployment of the web application to fail.

Backward compatibility issue. Specifying an empty <load-on-startup> still works with Servlet 2.3 based `web.xml`.

Solution

Specify `<load-on-startup>0</load-on-startup>` when using a Servlet 2.4 based `web.xml` to indicate that servlet load order does not matter.

Unable to Compile JSP Page on Resource-Constrained Servers. (ID 6184122)

The JSP page is accessed but fails to compile, and the server log contains the error message "Unable to execute command" with the following stack trace:

```
at org.apache.tools.ant.taskdefs.Execute$Java13CommandLauncher.
exec(Execute.java:655) at org.apache.tools.ant.taskdefs.Execute.
launch(Execute.java:416)
at org.apache.tools.ant.taskdefs.Execute.execute(Execute.java:427)
at org.apache.tools.ant.taskdefs.compilers.DefaultCompilerAdapter.
executeExternalCompile(DefaultCompilerAdapter.java:448)
at org.apache.tools.ant.taskdefs.compilers.JavacExternal.execute
(JavacExternal.java:81)
at org.apache.tools.ant.taskdefs.Javac.compile(Javac.java:842)
at org.apache.tools.ant.taskdefs.Javac.execute(Javac.java:682)
at org.apache.jasper.compiler.Compiler.generateClass(Compiler.java:396)
```

Solution

Set the JSP compilation switch "fork" to "false."

You can activate this setting in either of two ways:

- Globally, by setting the fork init parameter of the JspServlet in `${S1AS_HOME}/domains/domain1/config/default-web.xml` to false:

  ```
  <servlet> <servlet-name>jsp</servlet-name>
  <servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class>
  .... <init-param>
  <param-name>fork</param-name> <param-value>false</param-value>
  </init-param> .... </servlet>
  ```

- On a per-web application basis, by setting the fork JSP configuration property in `sun-web.xml` to false:

```
<sun-web-app> <jsp-config> <property name="fork" value="false" />
</jsp-config> </sun-web-app>
```

Either setting prevents `Ant` from spawning a new process for `javac` compilation.

Application Server Does Not Support `auth-passthrough` Web Server 6.1 Add-On (ID 6188932)

The Sun Java System Application Server Enterprise Edition 8.2 adds support for the functionality provided by the `auth-passthrough` plug-in function available with Sun Java System Application Server Enterprise Edition 7.1. However, in Application Server Enterprise Edition 8.2, the `auth-passthrough` plug-in feature is configured differently.

The `auth-passthrough` plug-in function in Application Server Enterprise Edition 7.1 has been useful in two-tier deployment scenarios, with the following constraints:

- Application Server instance is protected by a second firewall behind the corporate firewall.
- No client connections are permitted directly to the Application Server instance.

In such network architectures, a client connects to a front-end web server, which has been configured with the `service-passthrough` plugin function and forwards HTTP requests to the proxied Application Server instance for processing. The Application Server instance can receive requests only from the web server proxy, but never directly from any client hosts. As a result, any applications deployed on the proxied Application Server instance that query for client information, such as the client's IP address, will receive the proxy host IP, as that is the actual originating host of the relayed request.

In Application Server Enterprise Edition 7.1, the `auth-passthrough` plug-in function could be configured on the proxied Application Server instance in order to make the remote client's information directly available to any applications deployed on it; as if the proxied Application Server instance had received the request directly, instead of via an intermediate web server running the `service-passthrough` plug-in.

In Application Server Enterprise Edition 8.2, the `auth-passthrough` feature may be enabled by setting the `authPassthroughEnabled` property of the `<http-service>` element in `domain.xml` to TRUE, as follows:

```
<property name="authPassthroughEnabled" value="true"/>
```

The same security considerations of the `auth-passthrough` plug-in function in Application Server Enterprise Edition 7.1 also apply to the `authPassthroughEnabled` property in Application Server Enterprise Edition 8.2. Since `authPassthroughEnabled` enables the overriding of information that may be used for authentication purposes (such as the IP address from which the request originated, or the SSL client certificate). Therefore, only trusted clients or servers should be allowed to connect to an Application Server Enterprise Edition 8.2 instance with `authPassthroughEnabled` set to `TRUE`. As a precautionary measure, configure only servers behind the corporate firewall with `authPassthroughEnabled` set to `TRUE`. A server that is accessible through the Internet must never be configured with `authPassthroughEnabled` set to `TRUE`.

Notice that in the scenario where a proxy web server has been configured with the `service-passthrough` plug-in and forwards requests to an Application Server 8.1 Update 2 instance with `authPassthroughEnabled` set to `TRUE`, SSL client authentication can be enabled on the web server proxy, and disabled on the proxied Application Server 8.1 Update 2 instance. In this case, the proxied Application Server 8.1 Update 2 instance still treats the request as though it was authenticated through SSL, and provide the client's SSL certificate to any deployed applications requesting it.

HTTP Listener Created With `--enabled=false` DoesNot Disable the Listener (ID 6190900)

When an `httplistener` is created with the `--enabled=false` flag, the listener does not become disabled. The flag `--enabled` does not have any effect when used at the same time the listener is created.

Solution

Create the listener in an enabled state, then disable it manually later.