Software Upgrade

Table of Contents

How to upgrade existing installation of the ADSS Server to the latest version?

Click here to find out the high level steps and some aspects to consider when upgrading ADSS Server. Also see the section 4.1.3 **Upgrading the existing ADSS Server instance** in the **ADSS Server installation guide**.

Special steps for upgrading ADSS Server with Azure SQL

1. Go to location: **[ADSS-Server-Installation-Dir]\tomcat\bin** and edit these files:

For Linux

- Edit **catalina.sh** file in a text editor and search for the parameter **JAVA_OPTS** and add parameter **-Dcom.sun.net.ssl.enableECC=false** at the end and save the changes as shown below:

catalina.sh

```
JAVA_OPTS="$JAVA_OPTS
-Dorg.apache.catalina.connector.CoyoteAdapter.ALLOW_BACKSLASH=true
-Dorg.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true
-Dcom.sun.net.ssl.enableECC=false"
```

For Windows

- Edit **catalina.bat** file in a text editor and search for the strings **%JAVA_OPTS% %CATALINA_OPTS%** and add parameter **-Dcom.sun.net.ssl.enableECC=false** at the end of each string and save the changes as shown below:

catalina.bat

```
%_EXECJAVA% %JAVA_OPTS% %CATALINA_OPTS% %DEBUG_OPTS%
-Dcom.sun.net.ssl.enableECC=false
-Djava.endorsed.dirs="%JAVA_ENDORSED_DIRS%" -classpath "%CLASSPATH%"
-Dcatalina.base="%CATALINA_BASE%" -Dcatalina.home="%CATALINA_HOME%"
-Djava.io.tmpdir="%CATALINA_TMPDIR%" %MAINCLASS% %CMD_LINE_ARGS%
%ACTION%
goto end
:doSecurity

%_EXECJAVA% %JAVA_OPTS% %CATALINA_OPTS% %DEBUG_OPTS%
-Dcom.sun.net.ssl.enableECC=false
-Djava.endorsed.dirs="%JAVA_ENDORSED_DIRS%" -classpath "%CLASSPATH%"
-Djava.security.manager
-Djava.security.policy=="%SECURITY_POLICY_FILE%"
-Dcatalina.base="%CATALINA_BASE%" -Dcatalina.home="%CATALINA_HOME%"
-Djava.io.tmpdir="%CATALINA_TMPDIR%" %MAINCLASS% %CMD_LINE_ARGS%
%ACTION%
goto end
:doJpda

if not "%SECURITY_POLICY_FILE%" == "" goto doSecurityJpda
%_EXECJAVA% %JAVA_OPTS% %CATALINA_OPTS% %JPDA_OPTS% %DEBUG_OPTS%
-Dcom.sun.net.ssl.enableECC=false
-Djava.endorsed.dirs="%JAVA_ENDORSED_DIRS%" -classpath "%CLASSPATH%"
-Dcatalina.base="%CATALINA_BASE%" -Dcatalina.home="%CATALINA_HOME%"
-Djava.io.tmpdir="%CATALINA_TMPDIR%" %MAINCLASS% %CMD_LINE_ARGS%
%ACTION%
goto end
:doSecurityJpda

%_EXECJAVA% %JAVA_OPTS% %CATALINA_OPTS% %JPDA_OPTS% %DEBUG_OPTS%
-Dcom.sun.net.ssl.enableECC=false
-Djava.endorsed.dirs="%JAVA_ENDORSED_DIRS%" -classpath "%CLASSPATH%"
-Djava.security.manager
-Djava.security.policy=="%SECURITY_POLICY_FILE%"
-Dcatalina.base="%CATALINA_BASE%" -Dcatalina.home="%CATALINA_HOME%"
-Djava.io.tmpdir="%CATALINA_TMPDIR%" %MAINCLASS% %CMD_LINE_ARGS%
%ACTION%
goto end
:end
```

- Edit the **service.bat** file in a text editor and search for the parameter **--JvmOptions** and **++JvmOptions** one by one, add parameter **;-Dcom.sun.net.ssl.enableECC=false** at the following location for both of them and save the changes

service.bat

```
"%EXECUTABLE%" //US//%SERVICE_NAME% --JvmOptions
"-Dorg.apache.catalina.connector.CoyoteAdapter.ALLOW_BACKSLASH=true;-D
org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true;-Dcatalin
a.base=%CATALINA_BASE%;-Dcatalina.home=%CATALINA_HOME%;-Djava.endorsed
.dirs=%CATALINA_HOME%\endorsed;-Dcom.sun.net.ssl.enableECC=false"
--StartMode jvm --StopMode jvm

"%EXECUTABLE%" //US//%SERVICE_NAME% ++JvmOptions
"-Dorg.apache.catalina.connector.CoyoteAdapter.ALLOW_BACKSLASH=true;-D
org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true;-Djava.io
.tmpdir=%CATALINA_BASE%\temp;-Djava.util.logging.manager=org.apache.ju
li.ClassLoaderLogManager;-Djava.util.logging.config.file=%CATALINA_BAS
E%\conf\logging.properties;-Dcom.sun.net.ssl.enableECC=false" --JvmMs
%6 --JvmMx %7
```

How to retain the ADSS Server keystore password after ADSS Server upgrade?

If you have changed the ADSS Server Keystore password in the old installation then follow these instructions:

1. Go to the location: **[Old ADSS Server Installation Directory]/service/server/conf/server.xml**
2. Find the instances of the **8778** or **8779** connector tags and copy the value of **keystorePass** or **truststorePass** attribute
3. Go to the location: **[New ADSS Server Installation Directory]/util/bin**
    a. Execute the **encrypt_password.bat/sh** file accordingly
    b. Enter the copied password to encrypt it. The utility will show the encrypted password e.g. **Encrypted Password: p/aZouGB6w4vL9lmu7AKsw==**
4. Go to the location: **[New ADSS Server installation directory]/console/server/conf**
    a. Edit the **server.xml** in a text editor
    b. Find the instance of the **8774** connector tag and replace the values of **keystorePass** and **truststorePass** attributes with the newly encrypted password

server.xml

```
<Connector port="8774"
             ....
                   ....
                   keystoreFile="../../conf/adss.keystore"
                   keystorePass="8cktUaFLGOP2YVttsBTRKg=="
                   truststoreFile="../../conf/adss.keystore"
                   truststorePass="8cktUaFLGOP2YVttsBTRKg==" />
```

    c. Save the changes
5. Go to the location: **[ADSS Server installation directory]/service/server/conf**
    a. Edit the **server.xml** in a text editor
    b. Find the instance of the **8778** connector tag and replace the value of **keystorePass** attribute with the newly encrypted password

server.xml

```
<Connector port="8778"
             ....
                   ....
                   keystoreFile="../../conf/adss.keystore"
                   keystorePass="8cktUaFLGOP2YVttsBTRKg==" />
```

c. Find the instance of the **8779** connector tag and replace the values of **keystorePass** and **truststorePass** attributes with the newly encrypted password

server.xml

```
<Connector port="8779"
             ....
                   ....
                   keystoreFile="../../conf/adss.keystore"
                   keystorePass="8cktUaFLGOP2YVttsBTRKg=="
                   truststoreFile="../../conf/adss.keystore"
                   truststorePass="8cktUaFLGOP2YVttsBTRKg==" />
```

d. Save the changes

6. Restart the ADSS Server Console and Service instances from Windows NT Services panel or UNIX daemon in order to take the password change into effect

Are all configurations and transaction logs retained after upgrading?

ADSS Server on upgrade does not delete any configurations or transactional logs. On upgrade, the existing tables may be edited only to have new columns added as a result of DB schema change or their HMAC value being re-calculated. In short all existing configurations or transactional data remain in the database unless archived by the operator (or auto-archived based on configuration).

Why does the database size increase after an upgrade?

Background:

During an upgrade of ADSS Server the installation process ensures that the database structure is also upgraded to allow the latest version to run properly. This includes but is not limited to the creation of new tables, insertion of new records or updating of old records. The database size increase is actually negligible due to these structural changes, but the size of database transactional logs increases creating the main reason for the database growth. The database transaction logs store full information on any insertion, deletion and updating operations on a table.

In particular the database transaction log size increases because of the re-calculation of the HMAC values when upgrading. ADSS Server uses Hash Message Authentication Codes (HMAC) for protecting the integrity of its database records. These HMAC values are calculated automatically by ADSS Server whenever a database record is written, and these values can later be verified either manually at the request of the ADSS operator or automatically by ADSS Server if configured. Each HMAC value acts like a fingerprint for the input data it is protecting. Therefore HMAC values depend on the database scheme (i.e. which table column values are included within the HMAC).

When upgrading from a previous ADSS Server release the HMAC values need to be recomputed for any database tables for which the database schema has been changed. Depending upon the number of records in the database this may take some time. While updating HMACs, the DBMS transactional log may become very large because they store the information relating to the re-calculation of the HMACs.

Impact on System:
None (other than the use of extra space on the database machine).

Workaround:
Truncate the DBMS transactional log after an upgrade to reclaim space OR set your DBMS not to increase the transactional log above a limit. Consult the relevant database documentation on how to truncate transactional logs.

How to fix the error "Number of locks exceeds the lock table size" when upgrading ADSS Server using MySQL database?

When ADSS Server setup executes the upgrade scripts, it acquires lock on the database tables. If table sizes in the ADSS Server database are very large then MySQL database server may throw this error "Number of locks exceeds the lock table size". This happens when the MySQL database server doesn't have enough space to store all the table rows for which lock is acquired. Before ADSS Server upgrade, adjust **innodb_buffer_pool_size** accordingly and restart MySQL database server. You may consult your DBA for further instructions.

How to fix the "InnoDB The Table is Full"?

The issue that **InnoDB table is full** may arise with MySQL database due to the small log file size of InnoDB due to which transactions are not being written to the log file. The resolution is as follows:

1. In the **my.cnf** [**Linux**] or **my.ini** [**Windows**] configuration file of your MySQL Server, add/modify the following property right below the **mysqld** section in the configuration file: **innodb_data_file_path=ibdata1:1G:autoextend**
2. Restart the MySQL Server for the changes to take effect.