Configuring SSL communication between IBM Security Identity Manager server and IBM DB2 database server

Secure socket layer (SSL) communication can be used between an IBM® DB2 database server and IBM Security Identity Manager to secure database communication. You must configure the IBM® DB2 database server to use SSL for secure communications.

If you are using IBM® DB2 database to store IBM Security Identity Manager information, you must first set the server to use SSL. Then you must configure the SSL certificates that you want to use. This task can be done only after installing IBM Security Identity Manager. You cannot configure database through an SSL connection while installing IBM Security Identity Manager.

Following sections describes various steps required to configure SSL communication between IBM Security Identity Manager server and IBM® DB2 database server.

Enabling SSL on IBM DB2 Database Server

To have secure socket layer (SSL) communication between IBM DB2 Database Server and IBM Security Identity Manager server, you must configure IBM DB2 Database Server to listen on a port with a defined certificate. The certificate authority must be in the signer certificate database on the SSL client.

Use GSKit to create the key database file and certificates. Make sure to extract the server certificate (the one created for the database server) for client use. The certificate must be copied to the system where IBM Security Identity Manager server is running. The location of the server certificate is required to set up a trusted certificate for IBM Security Identity Manager in a later task.

For more information about activating SSL on database for IBM DB2 Database Server, see the documentation available in the IBM DB2 Information Center.

Configuration of the SSL client to trust the database server certificate

The IBM Security Identity Manager Server does not operate as an embedded part of WebSphere® Application Server. It operates as a Java™ application and uses Java secure socket extension (JSSE) to implement SSL support.

SSL certificates and CA certificates are retrieved in a standard Java truststore or keystore format. The truststore and keystore use the same file formats that the Java virtual machine and WebSphere Application Server use for other certificate configuration. You can use standard Java tools to maintain the truststores and keystores, like the IBM® Key Management tool or the Java Keytool command-line utility.

To configure the SSL connection between the IBM Security Identity Manager Server and DB2 Database Server, you must import the self-signed certificate or CA certificate created for the Database Server into the truststore. This truststore is used by the IBM JSSE, which is part of WebSphere Application Server. Additionally, you must first configure IBM Security Identity Manager to use SSL when communicating with the Database Server.

Following sections describe the various tasks one has to perform to install a self-signed certificate and configure ISIM to use it and establish SSL connection with DB2 database.

1. Configure Websphere Application Server to use SSL communication when communicating with the IBM DB2 database server

In order for IBM Security Identity Manager Server application to establish and validate this SSL connection, the signer of the DB2 certificate needs to be available to the application Server.

Before you begin

Ensure that the WebSphere Application Server is running and that you start the WebSphere administrative console. You also need WebSphere Application Server administrative user ID and password.

About this task

Import the signer certificate for the DB2 database into the WebSphere Application Server certificate store file.

Procedure

1. Log on to the WAS admin console.

2. Click Security -> SSL certificate and key management; under “Related items”, click “Key stores and certificates”.

3. For the single severe environment, click NodeDefaultTrustStore, otherwise click on CellDefaultTrustStore in case of clustered environment.

4. On the right, under “Additional Properties”, click Signer certificates.

5. Click “Retrieve from port” to contact the port and request the signer.

6. Enter the host name of the DB2 server, the SSL port number, and an alias for the certificate that you are importing.

7. Click “Retrieve signer information”; the signer information is displayed;

8. Click OK to import the certificate.

9. Click Save.

This is one of the many alternative ways to make the signer certificate available to application server.

What to do next

Restart WebSphere Application Server to use this certificate.

2. Configuring IBM Security Identity Manager to use SSL when communicating with the IBM DB2 Database server

In order for IBM Security Identity Manager Server application to establish SSL connection, you must configure IBM Security Identity Manager Server to use SSL connection.

Before you begin

Ensure that the IBM Security Identity Manager installation process is completed.

About this task

Edit ISIM_HOME/data/enRoleDatabase.properties file. In case of clustered environment, the configuration steps need to be performed on IBM Security Identity Manager instances deployed on deployment manager and each member of the cluster.

Procedure

1. Take a backup of ISIM_HOME/data/enRoleDatabase.properties file.
2. Edit the ISIM_HOME/data/enRoleDatabase.properties using any text editor and make the following changes:

   a. Change the port number in the database url for the database.jdbc.driverUrl property to the SSL port number configured on database server. Set sslConnection=true property in the database url as shown below. Please make sure that ssl property is the first property set to database url.

   For example,

   ```
   database.jdbc.driverUrl=jdbc:db2://localhost:50000/ISIMDB:sslConnection=true;
   ```

   b. Set the value of the database.db.security.protocol property to ssl. This setting indicates to the Security Identity Manager Server to use SSL to communicate to database when running DBConfig utility.

   ```
   database.db.security.protocol=ssl
   ```
3. Save the changes.

What to do next

Configure the ISIM_HOME/bin/DBUpgrade.lax file as per the steps given in the 'Running DBUpgrade with SSL enabled' section. This configuration is required to execute the DBUpgrade utility during IBM Security Identity Manager fix pack installation.

3. Configure IBM Security Identity Manager Data Source to use SSL when communicating with the Database server

After configuring WebSphere application server and IBM Security Identity Manager to use SSL connection, you must configure IBM Security Identity Manager Data Source.

Before you begin

You must make the signer certificate for the database server available to Websphere Application Server. You must finish configuring IBM Security Identity Manager to use SSL communication with database server.

Ensure that the WebSphere Application Server is running and that you start the WebSphere administrative console. You also need WebSphere Application Server administrative user ID and password.

About this task

Edit IBM Security Identity Manager data sources in the WebSphere Application Server. In the WebSphere Application Server console, click Resources > JDBC > Data sources. Perform the following steps for all the IBM Security Identity Manager application related data source such as "ITIM Data Source", "ITIM Bus DataSource ", and "ITIM Bus Shared DataSource" (in case of clustered environment).

Procedure

1. Click on the data source to be configured.

2. Click on "Custom properties" under the heading "Additional Properties".

3. Click New to create the "sslConnection" property with the following values:

    Scope: Use default
    Name: sslConnection
    Value: true
    Type: java.lang.Boolean

4. Click on "Apply" button. Save the changes.

5. Change the port number in "Port number" field to the SSL port number configured on database server.

6. Click on the "Apply". Save the changes.

7. Verify that other properties, database server, database name, database user name and password are properly set.

8. Verify whether the connection to database is successful using "Test connection".

What to do next

Restart WebSphere Application Server.

4. Installing the self-signed certificate in the JSSE truststore

Use this procedure to install the self-signed certificate and to add it to the certificate store. This certificate store is used to execute the configuration and stand alone utilities in IBM Security Identity Manager.

Before you begin

For this task, the default truststore that is present in the JRE of the WebSphere® Application Server is used. Alternately you can create your own certificate store location and use it to specify the JSSE System Properties in the configuration files. The ikeyman utility is used to configure the certificates.

About this task

Install the signed certificate in the JSSE truststore. In case of the clustered environment, the certificate store is required for IBM Security Identity Manager instance deployed on deployment manager and each member of the cluster.

Procedure

1. Start the ikeyman utility. The utility (ikeyman.bat or ikeyman.sh) is in the WAS_HOME/bin.
2. From the Key Database File menu, select Open.
3. In the key database type, select JKS.
4. In the File Name field, type cacerts.
5. In the Location field, type WAS_HOME/java/jre/lib/security/.
6. In the password prompt window, type the password in the “Password” and “Confirm Password” field. The default password is changeit.
7. Click OK.
8. Add the certificate you created for the database server into this certificate store by executing the following steps.
    1. In the main window, in the Key database content area, select Signer Certificates from the list.
    2. Click Add.
    3. In the Certificate file name field, browse and locate the server certificate file that was created for the database server, which is in Base64 encoded ASCII data. Verify that the appropriate directory is displayed in the Location field.
    4. Click OK.
    5. In the prompt, type a label for this certificate. For example, type DATABASECA.
    6. Click OK.

    Note: If you are not able to locate the server certificate file as previously described, extract it from the server certificate store. For IBM® DB2 Database Server, use the ikeyman utility to extract the certificate.

    The certificate is added in the certificate store. You can now close the ikeyman utility.

What to do next

Use the certificate store to specify the JSSE system properties in the configuration files.

Running DBConfig with SSL enabled

If database is configured to use SSL only with IBM Security Identity Manager, follow these steps to manually run the DBConfig utility.

Before you begin

Ensure that the IBM Security Identity Manager installation process is completed. All the steps required to configure IBM Security Identity Manager to use SSL when communicating with the IBM DB2 Database server are executed.

About this task

Edit ISIM_HOME/bin/DBConfig.lax file to specify JSSE system properties. In case of the clustered environment, this configuration should be performed on IBM Security Identity Manager instance deployed on deployment manager only.

Procedure

1. Verify that enRoleDatabase.properties has database.db.security.protocol set to ssl.
2. Take a backup of ISIM_HOME/bin/DBConfig.lax file.
3. Edit ISIM_HOME/bin/DBConfig.lax file using any text editor. Add this property, which is one line:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=/opt/IBM/WebSphere/AppServer/java/jre/lib/security/cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/java/jre/lib/ext:/opt/IBM/WebSphere/AppServer/plugins:/
   opt/IBM/WebSphere/AppServer/lib:/opt/IBM/WebSphere/AppServer/lib/ext
   ```

   For example, on the Windows operating system:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\security\cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\ext;C:\Progra~1\IBM\WebSphere\Ap
   pServer\plugins;C:\Progra~1\IBM\WebSphere\AppServer\lib;C:\Progra~1\IBM\WebSphere\AppServer\lib
   \ext
   ```

   Save the changes.

   Note: On UNIX systems, the delimiter for the list of directories in java.ext.dirs must be a colon. On the Windows systems, the delimiter for these directories must be a semi-colon. Also, on Windows systems, use 8.3 notation for the directory names as there can be no spaces in the list. Depending on the version of WebSphere Application server and the JSSE configuration you might have to specify additional jars in the classpath.

What to do next

Run the DBConfig utility.

Running SAConfig with SSL enabled

If database is configured to use SSL only with IBM Security Identity Manager, follow these steps to run the SAConfig utility.

Before you begin

Ensure that the IBM Security Identity Manager installation process is completed. All the steps required to configure IBM Security Identity Manager to use SSL when communicating with the IBM DB2 Database server are executed.

About this task

Edit ISIM_HOME/bin/SAConfig.lax file to specify JSSE system properties. In case of the clustered environment, this configuration should be performed on IBM Security Identity Manager instance deployed on deployment manager and each member of the cluster.

Procedure

1. Before running the SAConfig utility, verify that the properties database.jdbc.driverUrl and database.db.security.protocol in ISIM_HOME/data/enRoleDatabase.properties file are set to use SSL communication with database server.

2. Take a backup of ISIM_HOME/bin/SAConfig.lax file.

3. Edit ISIM_HOME/bin/SAConfig.lax file in any text editor.

   Add this property, which is one line:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=/opt/IBM/WebSphere/AppServer/java/jre/lib/security/cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/java/jre/lib/ext:/opt/IBM/WebSphere/AppServer/plugins:/
   opt/IBM/WebSphere/AppServer/lib:/opt/IBM/WebSphere/AppServer/lib/ext
   ```

   For example, on the Windows operating system:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\security\cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\ext;C:\Progra~1\IBM\WebSphere\Ap
   pServer\plugins;C:\Progra~1\IBM\WebSphere\AppServer\lib;C:\Progra~1\IBM\WebSphere\AppServer\lib
   \ext
   ```

   Save the changes.

   Note: On UNIX systems, the delimiter for the list of directories in java.ext.dirs must be a colon. On the Windows systems, the delimiter for these directories must be a semi-colon. Also, on Windows systems, use 8.3 notation for the directory names as there can be no spaces in the list. Depending on the version of WebSphere Application server and the JSSE configuration you might have to specify additional jars in the classpath.

What to do next

Run the SAConfig utility.

Running runConfig with SSL enabled

If database is configured to use SSL only with IBM Security Identity Manager, follow these steps to manually run the runConfig utility.

Before you begin

All the steps required to configure IBM Security Identity Manager to use SSL when communicating with the IBM DB2 Database server are executed.

About this task

Edit ISIM_HOME/bin/runConfig.lax file to specify JSSE system properties. In case of the clustered environment, this configuration should be performed on IBM Security Identity Manager instance deployed on deployment manager and each member of the cluster.

Procedure

1. Before running the runConfig utility, verify that the properties database.jdbc.driverUrl and database.db.security.protocol in ISIM_HOME/data/enRoleDatabase.properties file are set to use SSL communication with database server.

2. Take a backup of ISIM_HOME/bin/runConfig.lax file.

3. Edit ISIM_HOME/bin/runConfig.lax file in any text editor.

   Add this property, which is one line:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=/opt/IBM/WebSphere/AppServer/java/jre/lib/security/cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/java/jre/lib/ext:/opt/IBM/WebSphere/AppServer/plugins:/
   opt/IBM/WebSphere/AppServer/lib:/opt/IBM/WebSphere/AppServer/lib/ext
   ```

   For example, on the Windows operating system:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\security\cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\ext;C:\Progra~1\IBM\WebSphere\Ap
   pServer\plugins;C:\Progra~1\IBM\WebSphere\AppServer\lib;C:\Progra~1\IBM\WebSphere\AppServer\lib
   \ext
   ```

   Save the changes.

   Note: On UNIX systems, the delimiter for the list of directories in java.ext.dirs must be a colon. On the Windows systems, the delimiter for these directories must be a semi-colon. Also, on Windows systems, use 8.3 notation for the directory names as there can be no spaces in the list. Depending on the version of WebSphere Application server and the JSSE configuration you might have to specify additional jars in the classpath.

What to do next

Run the runConfig utility.

Running DBUpgrade with SSL enabled

If database is configured to use SSL only with IBM Security Identity Manager, follow these steps to run the DBUpgrade utility manually or during a fix pack installation.

Before you begin

All the steps required to configuring IBM Security Identity Manager to use SSL when communicating with the IBM DB2 Database server are executed.

About this task

Edit ISIM_HOME/bin/DBUpgrade.lax file to specify JSSE system properties. In case of the clustered environment, this configuration should be performed on IBM Security Identity Manager instance deployed on deployment manager only.

Procedure

1. Verify that the properties database.jdbc.driverUrl and database.db.security.protocol in ISIM_HOME/data/enRoleDatabase.properties file are set to use SSL communication with the database server.

2. Take a backup of ISIM_HOME/bin/DBUpgrade.lax file.

3. Edit ISIM_HOME/bin/DBUpgrade.lax file in any text editor.

   Add this property, which is one line:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=/opt/IBM/WebSphere/AppServer/java/jre/lib/security/cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/java/jre/lib/ext:/opt/IBM/WebSphere/AppServer/plugins:/
   opt/IBM/WebSphere/AppServer/lib:/opt/IBM/WebSphere/AppServer/lib/ext
   ```

   For example, on the Windows operating system:

   ```
   lax.nl.java.option.additional=-Djavax.net.ssl.trustStoreType=jks
   -Djavax.net.ssl.trustStore=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\security\cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=C:\Progra~1\IBM\WebSphere\AppServer\java\jre\lib\ext;C:\Progra~1\IBM\WebSphere\Ap
   pServer\plugins;C:\Progra~1\IBM\WebSphere\AppServer\lib;C:\Progra~1\IBM\WebSphere\AppServer\lib
   \ext
   ```

   Note: On UNIX systems, the delimiter for the list of directories in java.ext.dirs must be a colon. On the Windows systems, the delimiter for these directories must be a semi-colon. Also, on Windows systems, use 8.3 notation for the directory names as there can be no spaces in the list. Depending on the version of WebSphere Application server and the JSSE configuration you might have to specify additional jars in the classpath.

   Save the changes.

4. Test if this property is set correctly.
   1. Make the above changes to the ISIM_HOME/bin/runConfig.lax file too.
   2. Click Test on the database screen. If the test returns a success message, the property is set correctly.
   3. Click on cancel and quit runConfig. Do not click on Ok or Apply button.

What to do next

Run the DBUpgrade utility.

Running the utilities that access the Database server with SSL

You must add a Java™ runtime property to utilities to that access the database server with SSL.

Before you begin

Ensure that the IBM Security Identity Manager installation is completed and IBM Security Identity Manager is configured with database server to use ssl.

About this task

The following utilities present in the ISIM_HOME/bin/<platform> directly access database:

config_remote_services
CrystalConfigWAS
CrystalUpgradeWAS
DBPurge
remove_service_profiles
startIncrementalSynchronizerCMD_WAS
startIncrementalSynchronizerUI_WAS
itim_report_data_sync_utility

To successfully run the above utilities when SSL is configured, you must complete these additional steps:

Procedure

1. Verify that the properties database.jdbc.driverUrl and database.db.security.protocol in ISIM_HOME/data/enRoleDatabase.properties file are set to use SSL communication with the database server.

2. Take a backup of the utility file before editing it.

3. Open the utility file (for example, DBPurge.sh or DBPurge.cmd) in a text editor.

   Add this property as a Java runtime property. The property is one line.

   ```
   -Djavax.net.ssl.trustStoreType=type_of_truststore -Djavax.net.ssl.trustStore=truststore_location
   -Djavax.net.ssl.trustStorePassword=truststore_password
   -Djava.ext.dirs=WAS_HOME/java/jre/lib/ext:WAS_HOME/plugins:WAS_HOME/lib:WAS_HOME/lib/ext
   ```

   For example, DBPurge.sh modified for SSL looks like this example:

   ```
   $JAVA -Djavax.net.ssl.trustStoreType=jks -Djavax.net.ssl.trustStore=/opt/ibm/cacerts
   -Djavax.net.ssl.trustStorePassword=changeit
   -Djava.ext.dirs=/opt/IBM/WebSphere/AppServer/java/jre/lib/ext:/opt/IBM/WebSphere/AppServer/plugins:/
   opt/IBM/WebSphere/AppServer/lib/ext -Xms64m -Xmx256m -classpath $CLASSPATH
   com.ibm.itim.systemConfig.cleanup.DBPurgeMain $*
   ```

   Save the changes to the utility file.

   Note: On UNIX systems, the delimiter for the list of directories in java.ext.dirs must be a colon. On the Windows systems, the delimiter for these directories must be a semi-colon. Also, on Windows systems, use 8.3 notation for the directory names as there can be no spaces in the list. Depending on the version of WebSphere Application server and the JSSE configuration you might have to specify additional jars in the classpath.

What to do next

Use the utilities.