Create New Certificates

To create new certificates:

1. Create a clean environment by deleting the junk files in the folders. You can create the directories as required.
2. Open command prompt, then enter `mkdir` to create a directory called `newcerts`.

   For example: `C:\>mkdir newcerts`
3. In command prompt, type `cd newcerts` The current working directory will be changed to newcerts.
4. Run the following commands:

```
openssl genrsa 2048 > ca-key.pem
```

```
openssl req -new -x509 -nodes -days 1000 -key ca-key.pem -out ca-c
ert.pem
```

```
openssl req -newkey rsa:2048 -days 1000 -nodes -keyout server-key.
pem -out serverreqpem
```

```
openssl rsa -in server-key.pem -out server-key.pem
```

```
openssl x509 -req -in server-req.pem -days 1000 -CA ca-cert.pem -C
Akey ca-key.pem -set_serial 01 -out server-cert.pem
```

```
openssl req -newkey rsa:2048 -days 1000 -nodes -keyout client-key.
pem -out clientreq.pem
```

```
openssl rsa -in client-key.pem -out client-key.pem
```

```
openssl x509 -req -in client-req.pem -days 1000 -CA ca-cert.pem -C
Akey ca-key.pem -set_serial 01 -out client-cert.pem
```

5. Add the above generated files 4 (b), 4 (g)), and 4(h) to `my.ini` file for Client and 4 (g), 4 (d), and 4 (e) files for Server sections as mentioned below.

   The MySQL configuration file for Windows is `my.ini`.
   [Client]

   #SSL Client side files

```
ssl-ca="C:/newcerts/ca-cert.pem"
ssl-cert="C:/newcerts/client-cert.pem"
ssl-key="C:/newcerts/client-key.pem"
ssl-cipher=DHE-RSA-AES256-SHA
```

[mysqld]

SSL Server side files

```
ssl-ca="C:/newcerts/ca-cert.pem"
ssl-cert="C:/newcerts/server-cert.pem"
ssl-key="C:/newcerts/server-key.pem"
ssl-cipher=DHE-RSA-AES256-SHA
```

6. Run the following commands in MySQL prompt:

```
GRANT USAGE ON agiliance.* TO 'agiliance''@'' REQUIRE SSL;
```

```
Example: GRANT USAGE ON agiliance.* TO 'agiliance'@'server_name' R
EQUIRE SSL;
```

```
FLUSH PRIVILEGES;
```

7. Restart the MySQL service and execute the query: show variables like 'have_%ssl%'

Variable Name	Value
have_openssl	YES
have_ssl	YES

MySQL is enabled for SSL connection.

8. Open a command prompt window an execute the following commands:

```
openssl pkcs12 -export -inkey client-key.pem -in client-cert.pem -
out client.packet
```

```
%JAVA_HOME%\bin\keytool.exe -importkeystore -deststorepass  -destk
eypass
 -destkeystore myKS.jks -srckeystore client.packet -srcstoretype P
KCS12 -
srcstorepass  -alias 1
```

```
%JAVA_HOME%\bin\keytool.exe -importcert -alias mysqlCA -trustcacer
ts -file
ca-cert.pem -keystore myKS.jks
```

9. Create a folder `sslStore` under the `\config` directory .

10. Copy the `myKS.jks` file and paste it in the `\config\sslStore` folder.

11. For JDBC URL, when you enable SSL, append the following string:
    `verifyServerCertificate=true&useSSL=true&requireSSL=true`

12. By default, if you enter the server name as `localhost` , you may face errors. Instead, enter the actual host and/or server
    name which is referred in 6(a).

13. Refer to step 6 (b) and run the query.

For more information on properties, please refer to the `agiliance.properties` document.