How to setup SSL with the VSE Script Connector

Java-based connector
VSE Script connector
VSE Script Client on workstation and VSE

Last formatted on: Monday, January 11, 2010

Joerg Schmidbauer
jschmidb@de.ibm.com

Dept. 3229
VSE Development
IBM Lab Böblingen
Schönaicherstr. 220

D-71032 Böblingen
Germany

Disclaimer

This publication is intended to help VSE system programmers setting up infrastructure for their operating environment. The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk. Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment. Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of other companies:

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.
Microsoft® Office Excel 2003 is a trademark of Microsoft Corporation in the United States and/or other countries.
Microsoft, Windows, Windows XP, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

Contents

Changes:

Jan 2010 – initial version.

1 Introduction

This paper describes the setup of SSL with the VSE Script connector. The VSE Script connector consists of the VSE Script Server, a Java application, and the VSE Script Client that can be used on any platform, including non-Java platforms. Usage scenarios for the VSE Script connector are for example including VSE data into spreadsheet applications like Microsoft Excel or Lotus 1-2-3. We didn't try OpenOffice so far.

The following software has been used in the test setup.

- z/VSE 4.2.1
- TCP/IP for VSE/ESA 1.5F as part of z/VSE 4.2
- VSE Connector Server as part of z/VSE 4.2 (job STARTVCS)
- VSE Connector Client as part of z/VSE 4.2
- VSE Script Server 4.2.0 with APAR PK71659
- VSE Script Client DLL as part of the VSE Script Server installation
- Java 1.6.0_05 from Sun Microsystems
- Microsoft® Office Excel 2003 SP3
- Keyman/VSE, update from 07/2009 with support for Java key stores (JKS)

2 Installing the prerequisite programs

How to download and install the prerequisite programs is described in detail in the "z/VSE e-business Connectors User's Guide".

- VSE Connector Client: chapter 5
- VSE Script Server: chapters 7 and 21.

In addition to these two components, you need:

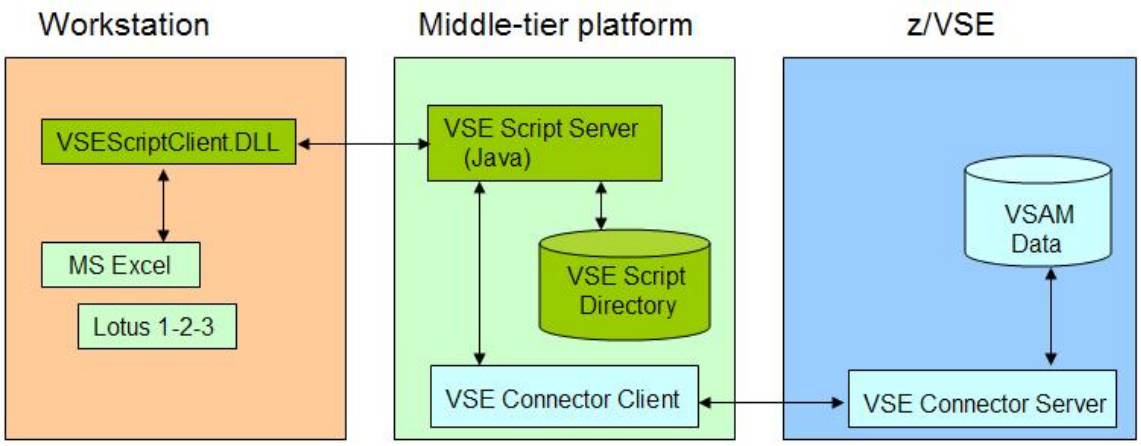
- Keyman/VSE: download and install the latest version from the VSE homepage.

The VSE Script Client DLL is not installed separately. It is part of the VSE Script server installation.

In the next chapters we show two scenarios with the VSE Script connector where SSL is not used. Later, these setups are modified in order to use SSL. In the first scenario the VSE Script Client runs on a workstation, in the second scenario it runs on z/VSE. The VSE Script Server always runs on a Java-enabled platform.

3 Non-SSL setup with client on workstation

The following picture shows a three-tier scenario with a *non-Java* VSE Script Client. This can be for example a Microsoft or Lotus office application. The VSE Script Client communicates with the VSE Script Server running on a Java-enabled middle-tier platform. The VSE Script Server uses the VSE Connector Client to transfer data to and from z/VSE via the VSE Connector Server.



3.1 Setup VSE Script Server

After installing the VSE Script server, you have to setup two properties files:

1. The server's properties file (VSEScriptServer.properties), and
2. The server's connection properties file (Connections.properties).

Normally, no changes are necessary in the server properties file.

In the connection properties file you have to define the IP address of your VSE system and the port number where the VSE Connector Server listens.

```
connection.1.name=z9_LPAR4
connection.1.ip=9.152.85.58
connection.1.port=2893
connection.1.userid=JSCH
connection.1.password=mypasswd

connection.timeout=100
```

Note: During the initial startup of the VSE Script Server the password is encrypted and stored using the property `connection.n.encpassword`.

3.2 Setup VSE Script Client

There are several steps necessary to enable a non-Java application running as a VSE Script Client.

1. Make the VSE Script Client DLL accessible by the client application
2. Define the VSE script in the client application

3.2.1 VSE Script Client DLL

VSEScriptClient.DLL must be accessible by the office application. In our test setup we copied the DLL to `c:\windows`.

Note: the VSEScriptClient.DLL is not SSL enabled.

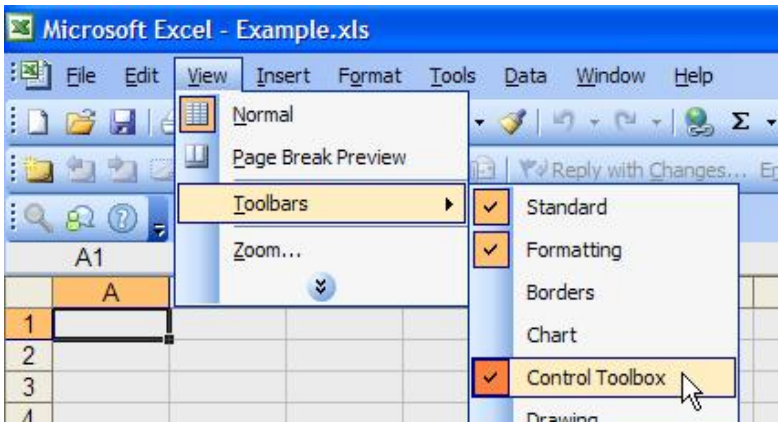
3.2.2 Setup client script in MS Excel

In our setup we use Microsoft Excel as the client application. The following steps show how to include the IBM provided sample script into MS Excel.

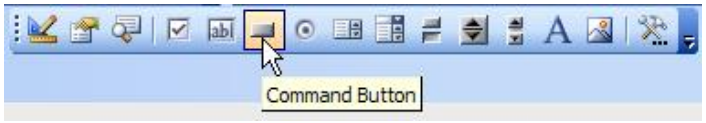
Note: there is an example with Lotus 1-2-3 in the "z/VSE e-business Connectors User's Guide".

As a first step create a new MS Excel worksheet and open it.

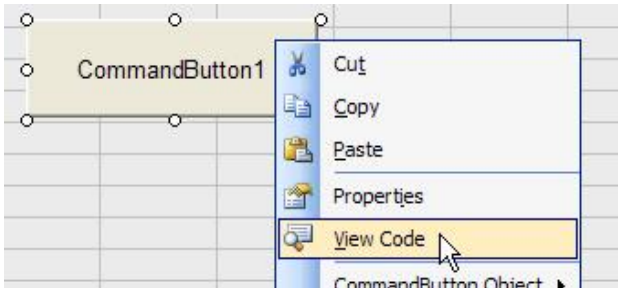
Make sure the Control Toolbox is visible.



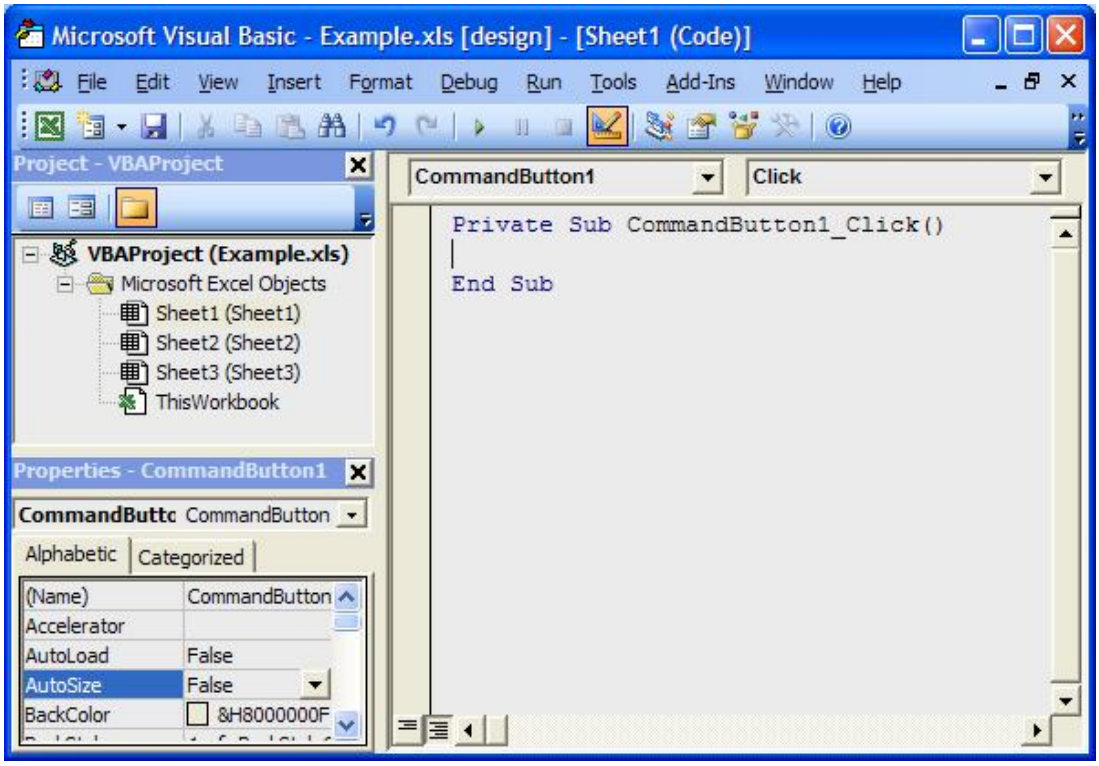
Insert a new Command Button.



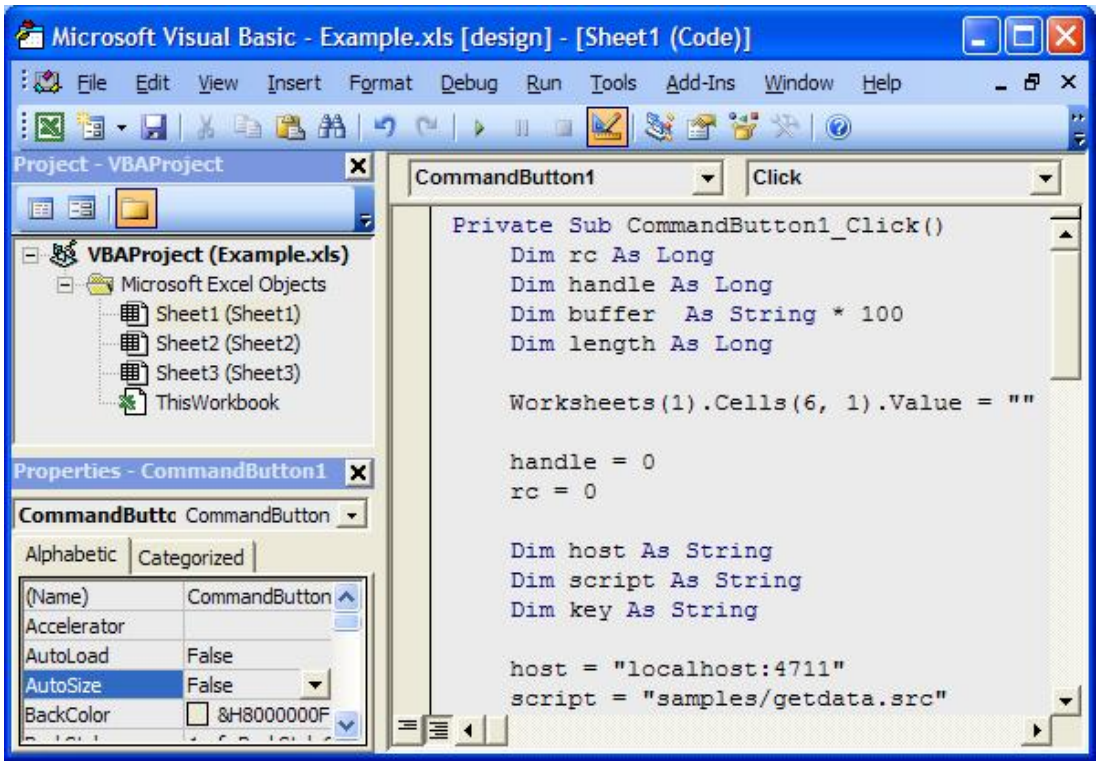
Then right-click the button and select View Code.



The Visual Basic design sheet opens. Make sure that the two drop-down list boxes show **CommandButton1** and **Click.**



Now copy/paste the body of the Visual Basic sample script VSEScriptClient.bas that is located in the VSE Script Server install directory into the text area.



This code assumes that you are running the VSE Script server on the same workstation as the VSE Script Client. If this is not the case, you have to change the following line accordingly.

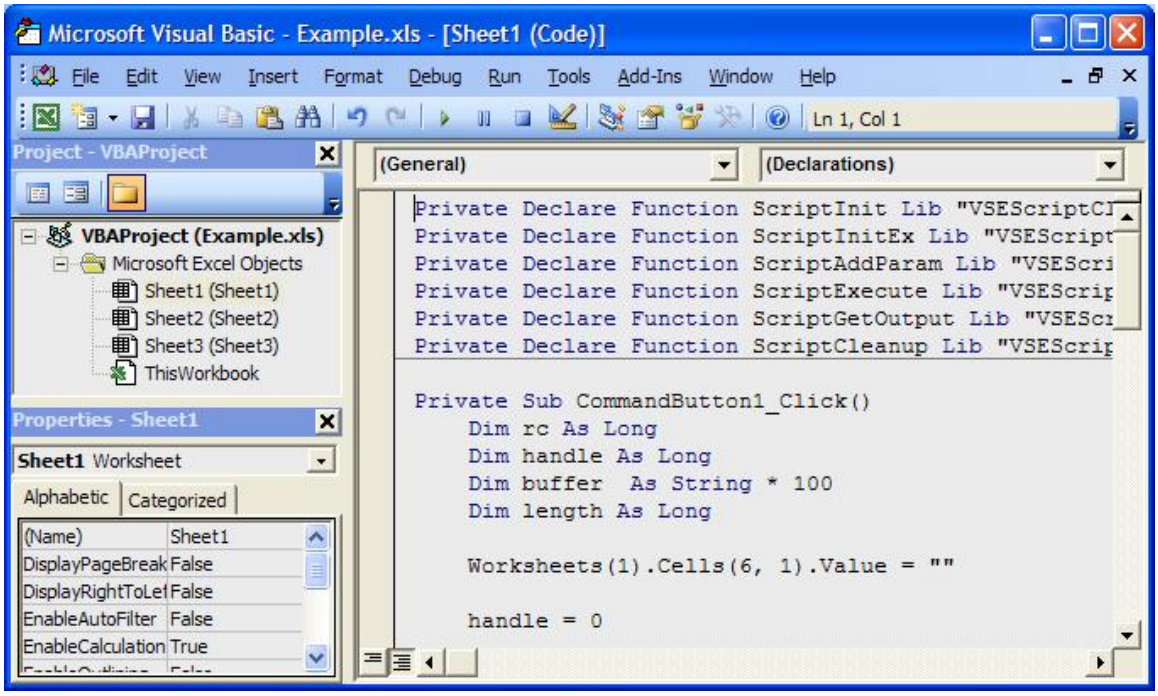
```
host = "localhost:4711"
```

It also assumes that the server script is located in the scripts/samples directory.

```
script = "samples/getdata.src"
```

If this is not the case, enter the correct pathname here.

Now select **(General)** and **(Declarations)** in the two drop-down list boxes.



Then copy/paste the variable declarations from the provided sample script.

> **Note:** the Visual Basic code provided with the VSE Script Server installation did not work with MS Excel 2003. We removed the first line of the sample code:
>
> ```
> Attribute VB_Name = "Module1"
> ```
>
> We also changed the declarations from `Public` to `Private`. Refer to section Visual Basic script syntax error on page 24 for more details about the error.

Now save your changes and exit.

3.2.3 Setup server script

In our setup we just use the provided sample server script `GetData.src`, which is located in scripts/samples.

Here we have to change some parameters according to our test environment. Open the script with a text editor and make sure you have the right host and file name defined.

```
String host = "z9_LPAR4";
String file = "VSESP.USER.CATALOG\\VSAM.CONN.SAMPLE.DATA\\USEDCARS";
```

Refer to section Setup sample VSAM data on page 9 for how to define the sample VSAM cluster and how to fill it with some sample data.

Save your changes and exit.

3.2.4 Start VSE Connector Server on VSE

Now make sure that the VSE Connector Server is started on VSE in non-SSL mode (job STARTVCS).

3.2.5 Setup sample VSAM data

Some sample VSAM data is provided with job SKVSSAMP in ICCF library 59. Job SKVSSAMP defines a new VSAM file `VSAM.CONN.SAMPLE.DATA` in the `VSESP.USER.CATALOG`. It automatically uses the IDCAMS RECMAP command to define a VSAM map that contains the field names of the sample data.

The VSAM file is then filled with some sample data (within job SKVSSAMP):

```
// LIBDEF *,SEARCH=(PRD1.BASE)
// EXEC VSAMSMPD,SIZE=AUTO
```

VSAMSMPD is a small IBM provided program that is used to fill the VSAM file with some sample records. The VSAM record layout matches the structure that is used in the client script.

3.2.6 Start the VSE Script server locally

Now start the VSE Script server on your workstation. Remember that the client script contains the line.

```
host = "localhost:4711"
```

This assumes the script server running locally on your workstation.

Double-click the `runserver.bat` file in the script connector install directory or start the server from the **Start – Programs** menu.

The server displays some output similar to the following.

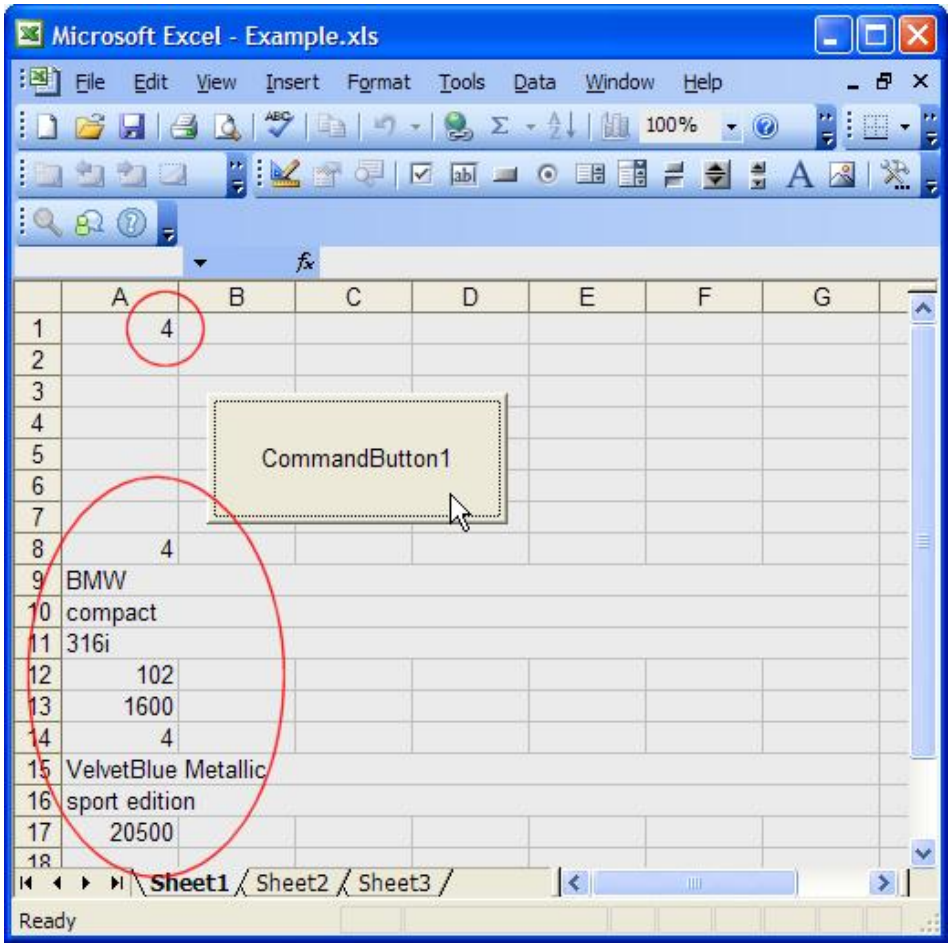
```
VSEScriptServer starting...
Dec 8, 2009 2:29:08 PM (1) - Listening socket created on port 4711
Dec 8, 2009 2:29:08 PM (1) - Starting ConnectionManager
Dec 8, 2009 2:29:08 PM (1) - Connection 'Z9_LPAR4' has been bound to
'9.152.85.58'
Dec 8, 2009 2:29:08 PM (1) - Connection 'DEMO2' has been bound to '9.12.34.78'
Enter 'quit' to stop the server
Dec 8, 2009 2:29:08 PM (8) - Waiting for connections...
```

3.2.7 Running the client script

You can now run the script by pressing the command button in the Excel worksheet. The following line in the script reads the key of the VSAM data from cell (1,1).

```
key = Worksheets(1).Cells(1, 1).Value
```

So just enter a valid key in cell (1,1) and press the command button.



The transferred VSAM data is now shown in the spreadsheet.

The scripts server shows the status of the VSE connection.

```
Dec 8, 2009 4:07:18 PM (8) - Waiting for connections...
Dec 8, 2009 4:23:24 PM (8) - Client connection request from 127.0.0.1
Dec 8, 2009 4:23:24 PM (22) - Client has been accepted.
Dec 8, 2009 4:23:24 PM (22) - Connection has been accepted from 127.0.0.1
Dec 8, 2009 4:23:24 PM (22) - Using default system codepage.
Dec 8, 2009 4:23:24 PM (22) - Executing script 'samples/getdata.src'
Dec 8, 2009 4:23:24 PM (22) - Connection has been terminated from 127.0.0.1
Dec 8, 2009 4:23:24 PM (22) - Client has been disconnected.
```

You can enter the STATUS command to see the connection status.

```
status
```

```
Status Information:

Clients:

  0 clients connected.

ConnectionManager:

  1. Connection: Z9_LPAR4 (unused, current timeout=96)

  Number of used Connections:   0
  Number of unused Connections: 1
```

The output shows that the server caches the VSE connection for further reuse, i.e. when pressing the button again, the already established connection is used again.

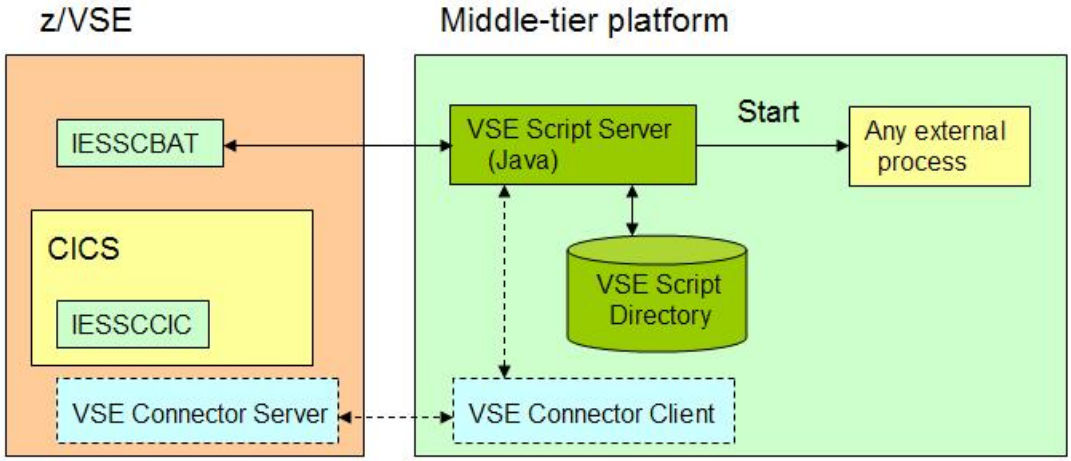
In the next chapter the same script is run on VSE using the VSE-based script client IESSCBAT.

4 Non-SSL setup with client on z/VSE

The following picture shows our second scenario, where the script client runs on VSE. This scenario is intended for controlling any processes on the platform where the script server runs, for example access to a local database and so on. The point is that you can initiate the process from z/VSE, similar to the REXEC function provided by TCP/IP. On the other hand, it's even possible to retrieve VSE data via the Java-based connector.

Basically, there are two VSE clients provided:

- IESSCBAT – runs in batch.
- IESSCCIC – runs in CICS, not considered here.



In the following sections we assume that the basic setup is already done, i.e. the VSAM file is set up, and the VSE Connector Server and VSE Script Server are running.

4.1 Setup IESSCBAT

You can find various examples of how to use IESSCBAT in the "z/VSE e-business Connectors User's Guide". We used the following JCL in our test environment.

```
* $$ JOB JNM=IESSCBAT,CLASS=A,DISP=D
// JOB IESSCBAT
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES'
9.152.222.37:4711
SAMPLES/GETDATA.SRC
7
/*
/&
* $$ EOJ
```

The above used IP address is the IP address of the workstation running the VSE Script Server. The IP address is followed by the name of the server script to be invoked. The script name is followed by the input argument (key for VSAM file).

Note that the server script `GETDATA.SRC` is still located on the workstation, although the script gets invoked from VSE.

The job output shows the retrieved data.

```
// JOB IESSCBAT
// LIBDEF *,SEARCH=PRD1.BASE
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES'
1S54I  PHASE IESSCBAT IS TO BE FETCHED FROM PRD1.BASE
7
Ford
Escort
ZX2 2 Door Coupe
150
2000
6
White
Sp.Seats,Zetec Eng.
15715
1S55I  LAST RETURN CODE WAS 0000
EOJ IESSCBAT  MAX.RETURN CODE=0000
```

4.2 Setup IESSCCIC

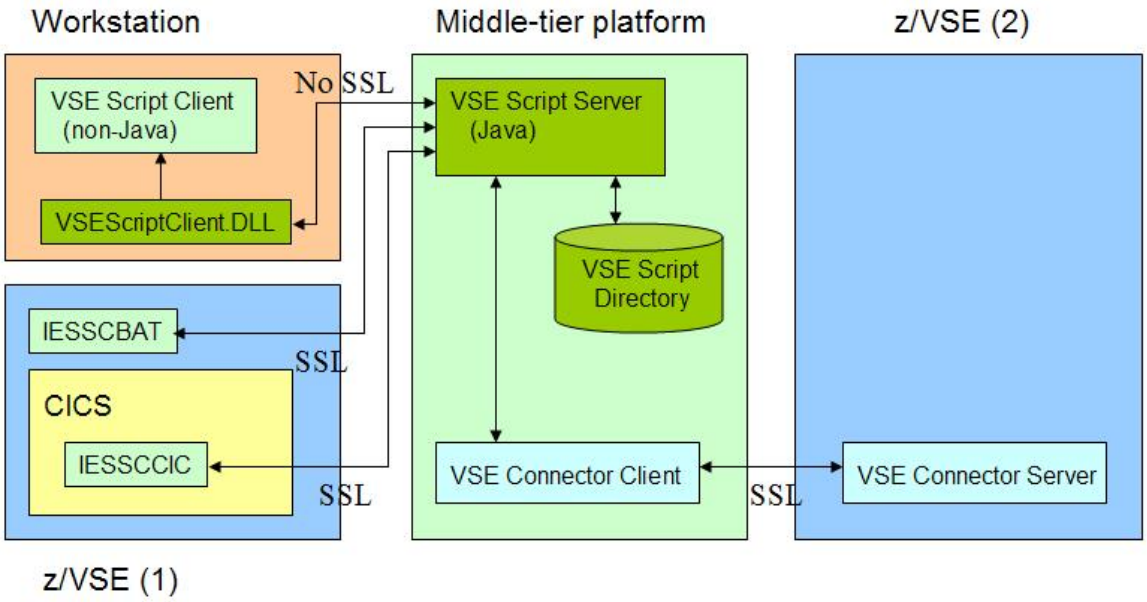
Refer to the "z/VSE e-business Connectors User's Guide" for how to use the IBM provided CICS client IESSCCIC.

5 General SSL setup

The following sections describe the basic SSL setup for the VSE Script connector with server and client authentication.

5.1 Overview of SSL connections

The following picture shows which connections can be secured with SSL. The picture shows two VSE systems, where one VSE runs the script client and the other VSE is used to retrieve data via the VSE Connector Client. Of course, these two VSE systems could be the same system.



The connection between the VSEScriptClient.DLL and the VSE Script Server cannot be SSL-enabled.

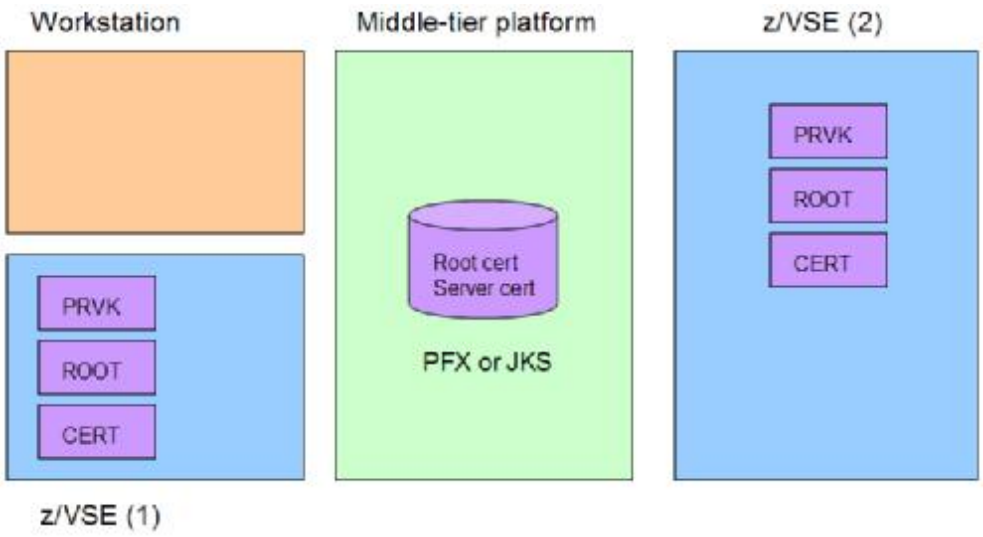
For a detailed description of the SSL setup between VSE Connector Client and VSE Connector Server refer to IBM Redbook “Security on IBM z/VSE”, SG24-7691 or document “How to setup SSL with VSE” on

http://www.ibm.com/servers/eserver/zseries/zvse/documentation/security.html#howto

In the following, we focus on the SSL setup between IESSCBAT and the VSE Script Server.

5.2 Overview of keys and certificates

The following picture shows how the keys and certificates must be stored on the various platforms.



- There are no keys and certificates necessary on the client platform, because the VSEScriptClient.DLL is not SSL-enabled
- A keyring file (PFX or JKS) is created with the Keyman/VSE tool on the middle-tier platform where the script server runs. After uploading the RSA key to the VSE systems, it is no longer needed in the keyring file.
- The RSA key, the VSE server certificate and the root certificate are transferred to the VSE systems.

The following section describes how to create the local keyring file (PFX or JKS) and how to upload the certificate items to VSE.

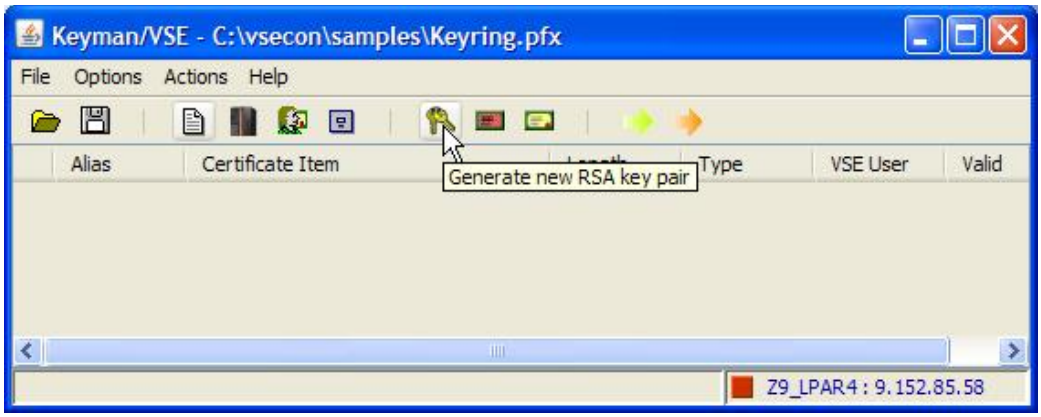
5.3 Generate the server key and certificates

These tasks are meanwhile described many times in different papers on the VSE homepage. Just have a look at:

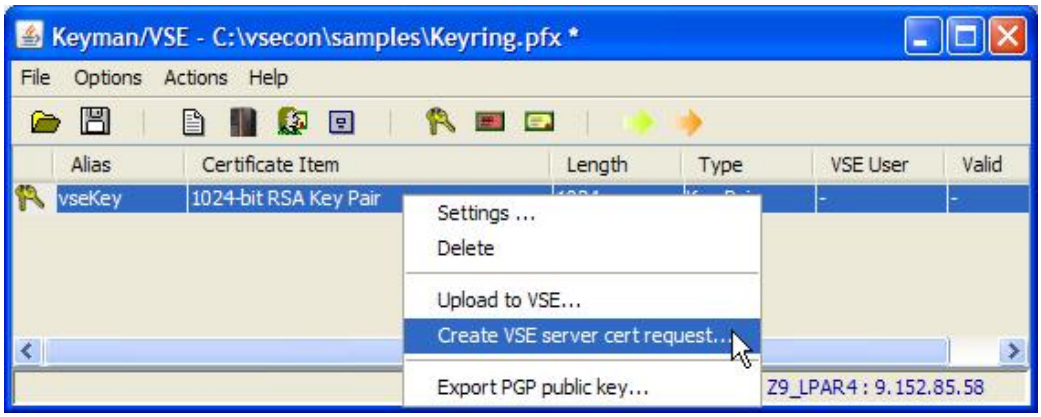
http://www.ibm.com/servers/eserver/zseries/zvse/documentation/security.html#howto

Here is a version not using the Wizard dialogs.

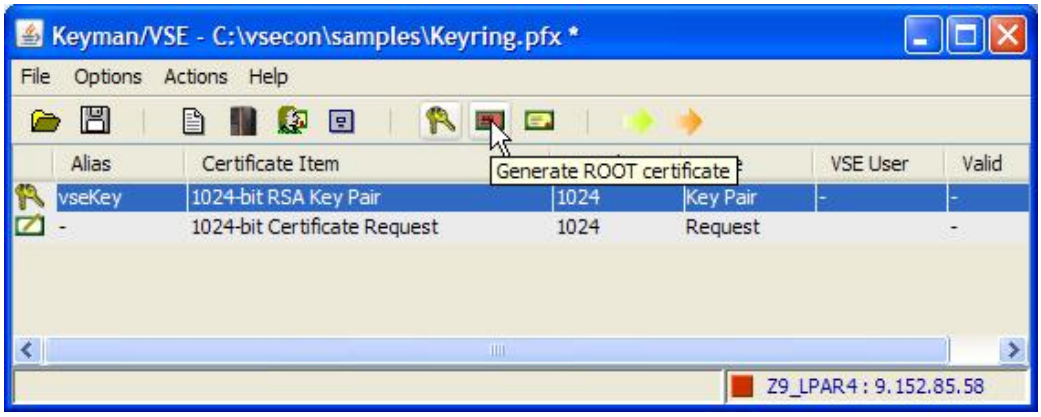
Start the Keyman/VSE tool and generate an RSA key.



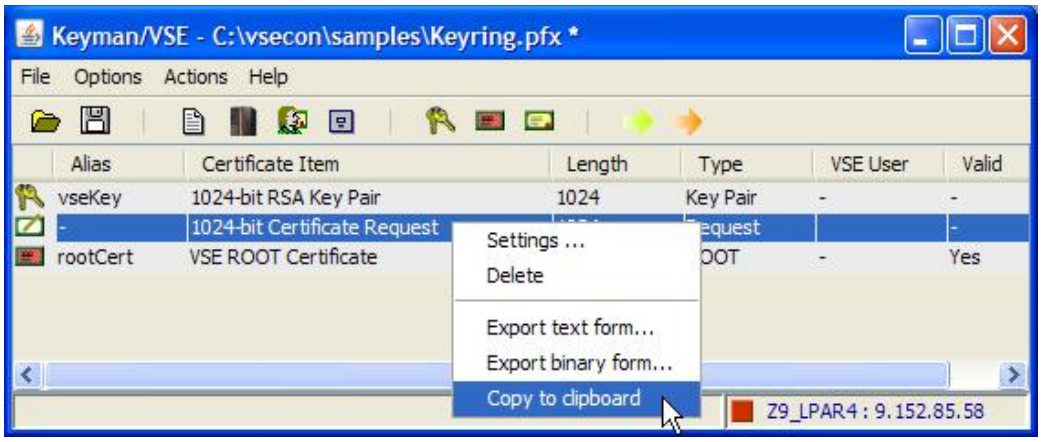
Then create a certificate request from this RSA key.



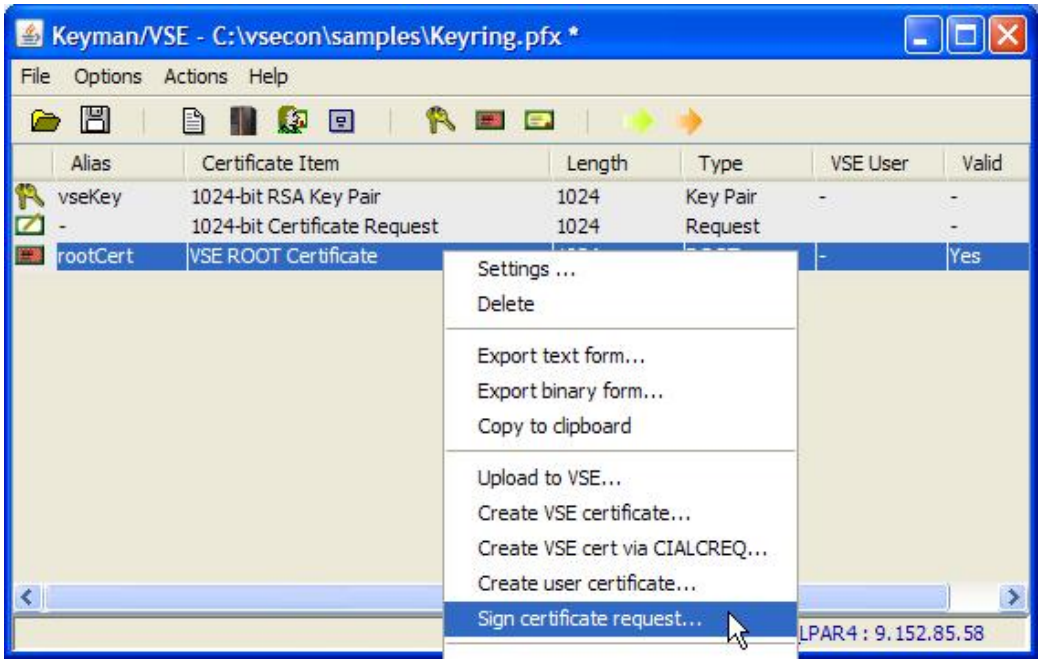
Then create a new self-signed root certificate.



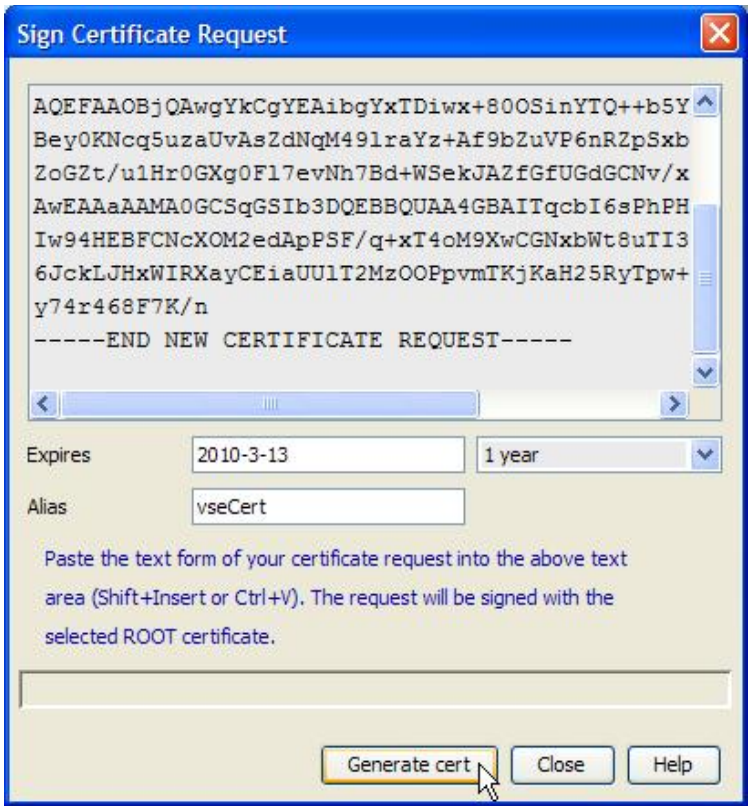
Then sign the certificate request with the self-signed root certificate.



You first copy the certificate request into the clipboard. Then right-click the root certificate and select "Sign certificate request".



Paste the certificate request into the text area of the next dialog box.

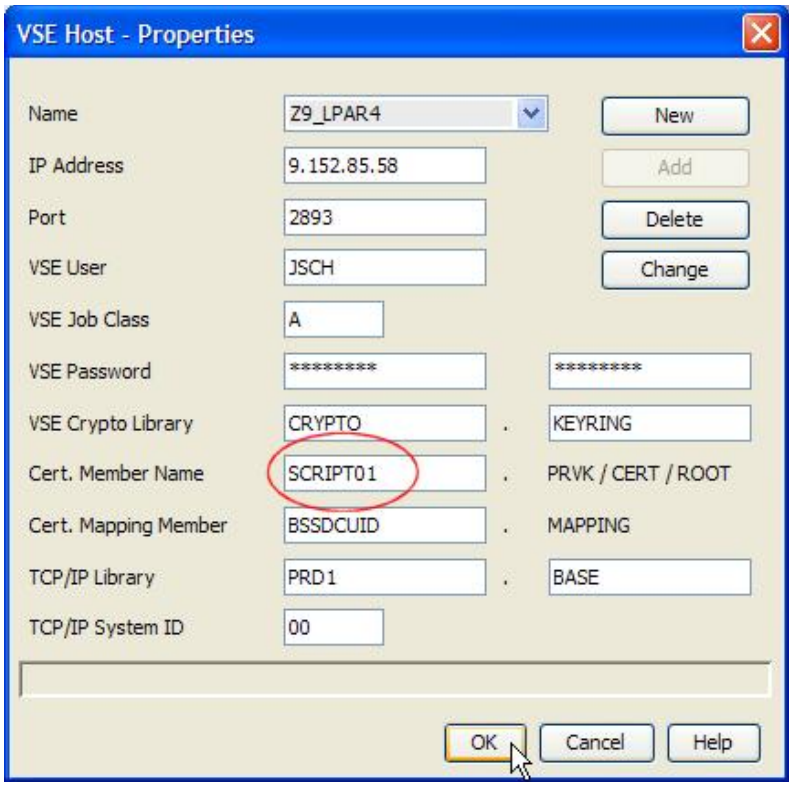


Then press “Generate cert”.

You can now delete the certificate request. Just select the table entry and press the DEL button on your keyboard.

The next steps are uploading the three items to all involved VSE systems. This requires that the VSE Connector Server runs on VSE in non-SSL mode.

You may pre-select the member name of the VSE keyring members in the VSE Host Properties dialog box.



Now upload the items by selecting “Upload to VSE” from their pop-up menus.

Now there are three new members in the VSE keyring library.

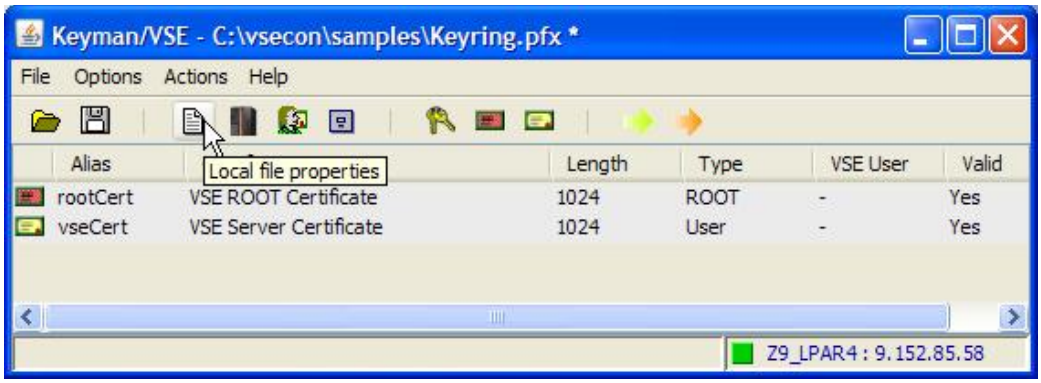
```
LD SCRIPT01.*

DIRECTORY DISPLAY    SUBLIBRARY=CRYPTO.KEYRING      DATE: 2009-12-09
                                                    TIME: 11:48
-------------------------------------------------------------------
 M E M B E R      CREATION   LAST     BYTES    LIBR CONT SVA  A- R-
NAME     TYPE     DATE      UPDATE   RECORDS   BLKS STOR ELIG MODE
-------------------------------------------------------------------
SCRIPT01 CERT     09-12-09   -  -      702 B       1 YES   -   -   -
SCRIPT01 PRVK     09-12-09   -  -     2048 B       3 YES   -   -   -
SCRIPT01 ROOT     09-12-09   -  -      700 B       1 YES   -   -   -
L113I RETURN CODE OF LISTDIR IS  0
```

The RSA key is now no more needed locally and should be deleted from the local keyring file.

Now save the two certificates into a local keyring file with file type JKS.

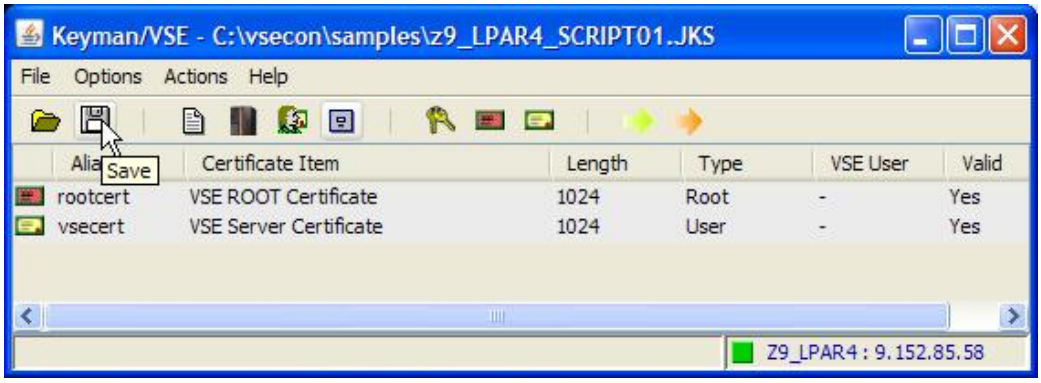
Note: SSL does not work with a PFX file when using a Java runtime environment from Sun! An IBM JRE works with both file types, PFX and JKS.

Open the “Local file properties” box.



Here, specify the file name, file type (JKS), and file password.

Press **OK**.

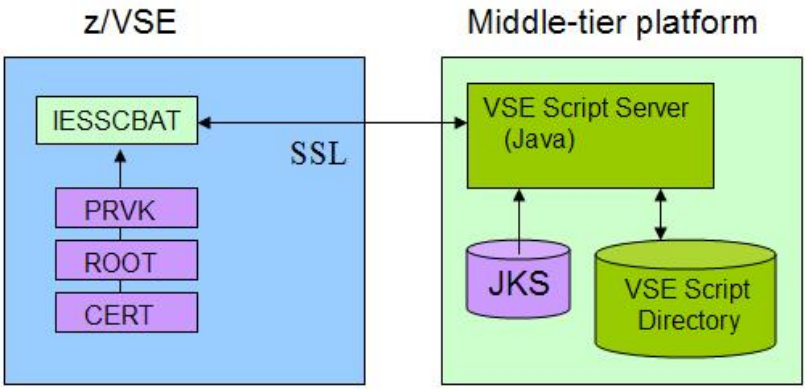


Press **Save**.

The following chapter describes the SSL setup for the VSE batch client IESSCBAT.

6 SSL setup with client on VSE

The following sections describe how to setup the keys and certificates for SSL between IESSCBAT and the VSE Script Server.



6.1 Setting up IESSCBAT for SSL

The SSL parameters for IESSCBAT are directly specified in the JCL. With the below JCL, IESSCBAT assumes the script server running in *SSL Server Authentication* mode. For how to enable IESSCBAT for client authentication refer to Using SSL Client Authentication on page 22.

```
* $$ JOB JNM=IESSCBAT,CLASS=A,DISP=D
// JOB IESSCBAT
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES SSL=YES'
9.152.222.37:4711
SSL30
CRYPTO.KEYRING
SCRIPT01
0A2F
86440
SAMPLES/GETDATA.SRC
7
/*
/&
* $$ EOJ
```

The keyring member name (SCRIPT01) must match the name you used before when uploading the key and certificates to VSE.

6.2 Setting up the VSE Script Server for SSL

To enable SSL for the VSE Script Server you have to modify the `VSEScriptServer.properties` file. Just uncomment the following lines and specify the appropriate values for your environment.

```
sslversion=SSL
keyringfile=c:\\vsecon\\samples\\z9_LPAR4_SCRIPT01.jks
keyringpwd=mypasswd
ciphersuites=SSL_RSA_WITH_3DES_EDE_CBC_SHA,TLS_RSA_WITH_AES_128_CBC_SHA
```

If you leave the comment for `ciphersuites`, all available cipher suites are enabled.
Here is the complete list of supported cipher suites and their meaning.

Hex Code	Cipher Suite	Encryption	Min. TCP/IP
01	SSL_RSA_WITH_NULL_MD5	None	1.5D
02	SSL_RSA_WITH_NULL_SHA	None	1.5D
08	SSL_RSA_EXPORT_WITH_DES40_CBC_SHA	40 bits	1.5D
09	SSL_RSA_WITH_DES_CBC_SHA	56 bits	1.5D
0A	SSL_RSA_WITH_3DES_EDE_CBC_SHA	168 bits	1.5D
2F	TLS_RSA_WITH_AES_128_CBC_SHA	128 bits	1.5E
35	TLS_RSA_WITH_AES_256_CBC_SHA	256 bits	1.5E
62	RSA1024_EXPORT_DESCBC_SHA	56 bits	1.5D

Table 1: list of supported SSL cipher suites

Notes:

- Cipher suites 01, 02, 08, 09, and 62 only provide weak encryption and should therefore no more be used in today's computing environments.
- Java does by default not allow using AES-256. If you want to use cipher suite 35, follow the instructions described in document "*How to setup SSL with VSE (PDF, 810KB)*", section "Using encryption with AES-256" on

  http://www.ibm.com/servers/eserver/zseries/zvse/documentation/security.html#howto

Now restart the VSE Script Server. You can now run IESSCBAT using SSL.

6.3 Testing the connection

When now running IESSCBAT again, output similar to the following should be produced.

```
// JOB IESSCBAT
// LIBDEF *,SEARCH=PRD1.BASE
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES SSL=YES'
1S54I  PHASE IESSCBAT IS TO BE FETCHED FROM PRD1.BASE
7
Ford
Escort
ZX2 2 Door Coupe
150
2000
6
White
Sp.Seats,Zetec Eng.
15715
1S55I  LAST RETURN CODE WAS 0000
```

The VSE Script Server shows:

```
Dec 9, 2009 4:26:10 PM (8) - Waiting for connections...
Dec 9, 2009 4:26:16 PM (8) - Client connection request from 9.152.85.58
Dec 9, 2009 4:26:16 PM (10) - Client has been accepted.
Dec 9, 2009 4:26:16 PM (10) - Connection has been accepted from 9.152.85.58
```

```
Dec 9, 2009 4:26:16 PM (10) - Session established with cipher suite:
TLS_RSA_WITH_AES_128_CBC_SHA
Dec 9, 2009 4:26:16 PM (10) - Using script-requested codepage CP1047
Dec 9, 2009 4:26:16 PM (10) - Executing script 'SAMPLES/GETDATA.SRC'
Dec 9, 2009 4:26:16 PM (10) - Created new Connection for 'Z9_LPAR4 (unused,
current timeout=100)'
Dec 9, 2009 4:26:16 PM (10) - Connection has been terminated from 9.152.85.58
Dec 9, 2009 4:26:16 PM (10) - Client has been disconnected.
```

Note: depending on the specified list of cipher suites in IESSCBAT, some errors occurred. Refer to section SSLHandshakeException: Invalid padding on page 25 for details.

6.4 Using SSL Client Authentication

If you want to use client authentication, you have to modify the server's properties file (VSEScriptServer.properties). Activate following parameter:

```
clientauthentication=true
```

Then change the IESSCBAT job like shown below:

```
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES SSL=YES           X
               ALLOWCLIENTAUTH=YES'
```

Now restart the VSE Script Server.

6.5 Testing the connection with client authentication

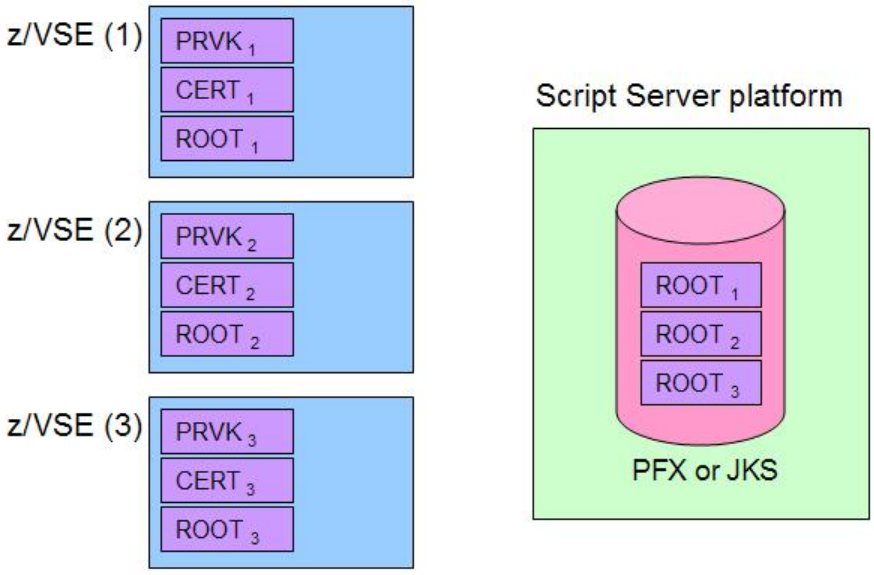
When running IESSCBAT with client authentication, the server now shows following output.

```
Dec 10, 2009 10:35:17 AM (8) - Waiting for connections...
Dec 10, 2009 10:35:35 AM (8) - Client connection request from 9.152.85.58
Dec 10, 2009 10:35:35 AM (10) - Client has been accepted.
Dec 10, 2009 10:35:35 AM (10) - Connection has been accepted from 9.152.85.58
Dec 10, 2009 10:35:36 AM (10) - Session established with cipher suite:
TLS_RSA_WITH_AES_128_CBC_SHA
Dec 10, 2009 10:35:36 AM (10) - Client Certificate:
Dec 10, 2009 10:35:36 AM (10) -   Subject:    CN=VSE Server Certificate,
OU=Development, O=IBM, L=Boeblingen, ST=Baden-Wuerttemberg, C=DE,
EMAILADDRESS=zvse@de.ibm.com
Dec 10, 2009 10:35:36 AM (10) -   Issuer:     CN=VSE ROOT Certificate, OU=IBM
Germany, O=IBM, L=Boeblingen, ST=Baden-Wuerttemberg, C=DE,
EMAILADDRESS=zvse@de.ibm.com
Dec 10, 2009 10:35:36 AM (10) -   Serial No:  1936789053
Dec 10, 2009 10:35:36 AM (10) -   Valid from: Wed Dec 09 13:36:46 CET 2009
Dec 10, 2009 10:35:36 AM (10) -   Valid to:   Sat Mar 13 13:36:46 CET 2010
Dec 10, 2009 10:35:36 AM (10) -   Valid:      true
Dec 10, 2009 10:35:37 AM (10) - Using script-requested codepage CP1047
Dec 10, 2009 10:35:37 AM (10) - Executing script 'SAMPLES/GETDATA.SRC'
Dec 10, 2009 10:35:37 AM (10) - Created new Connection for 'Z9_LPAR4 (unused,
current timeout=100)'
Dec 10, 2009 10:35:37 AM (10) - Connection has been terminated from 9.152.85.58
Dec 10, 2009 10:35:37 AM (10) - Client has been disconnected.
Dec 10, 2009 10:37:17 AM (9) - Destroyed Connection for 'Z9_LPAR4 (unused,
current timeout=0)'
```

6.6 Client authentication with multiple VSE clients

When using multiple VSE clients, you have to create a unique keyring for each involved VSE system. The JKS keystore on the script server side must contain copies of all root certificates stored on the different VSE systems. You can delete all RSA keys and VSE certificates ("yellow" certificates) from the JKS file.

Following picture shows the setup.



You can now run IESSCBAT on each VSE system using its own keyring, and thus, its own unique client certificate. As soon as you remove a root certificate from the JKS file, the related VSE client is no more able to connect.

If a VSE client certificate (stored in the CERT member on VSE) cannot be verified on the server side (because the related root certificate has been deleted from the JKS file), the following error occurs:

```
Dec 23, 2009 3:35:05 PM (11) - Connection has been accepted from 9.152.85.58
Dec 23, 2009 3:35:05 PM (11) - Error during SSL handshake:
javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException:
PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find
 valid certification path to requested target
Dec 23, 2009 3:35:05 PM (11) - Connection has been terminated from 9.152.85.58
Dec 23, 2009 3:35:05 PM (11) - Client has been disconnected.
```

7 Known problems

This chapter describes some errors that showed up in our test setup.

7.1 Visual Basic script syntax error

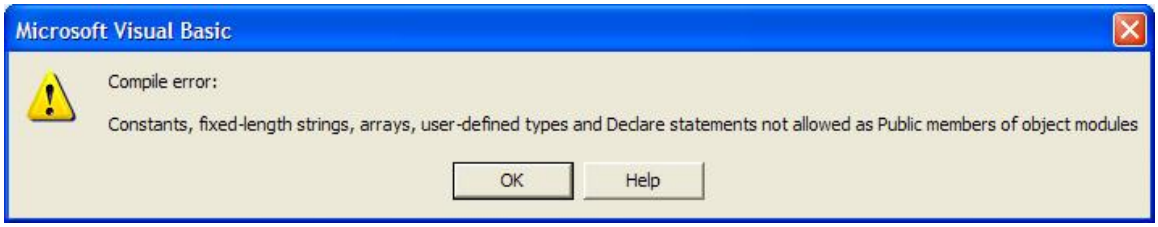
Symptom:

Following error message shows up in MS Excel.

Solution:

The IBM provided Visual Basic client script did not work with the MS Excel version we were using. There was a compile error caused by the first line in the declarations part.
Line `Attribute VB_Name = "Module1"` was not recognized.

After removing this line, the following error occurred.



This error could be resolved by defining the routines as private.

7.2 SSLException when starting VSE Script Server

Symptom:

The VSE Script Server shows following error:

```
Dec 9, 2009 1:37:41 PM (8) - Error: javax.net.ssl.SSLException: No available
certificate or key corresponds to the SSL cipher suites which are enabled.
```

Solution:

The local keyring file was stored in PFX file format. This file format cannot be read correctly by the Java runtime environment from Sun Microsystems. Using the JKS file format instead solves the problem.

7.3 SSLHandshakeException: Invalid padding

Symptom:

The VSE Script Server shows:

```
Dec 9, 2009 4:27:59 PM (10) - Error during SSL handshake:
javax.net.ssl.SSLHandshakeException: Invalid padding
```

Solution:
IESSCBAT used these cipher suites: 08090A2F35. Trying several combinations of cipher suites showed that cipher suite 08 causes the problem. Removing 08 solves the problem. At the moment we don't know the reason why cipher suite 08 fails. Suppressing the use of hardware instructions (CPACF) with a $SOCKOPT phase did not help.

However, as cipher suite 08 uses 40-bit DES, it should no more be used anyway.

7.4 SSL Client Authentication does not work

Symptom:

SSL Client Authentication does not work with either an IBM or Sun Java runtime environment. The VSE Script Server shows:

```
Dec 9, 2009 4:09:24 PM (8) - Error during SSL handshake:
javax.net.ssl.SSLHandshakeException: null cert chain
```

The IESSCBAT job ends with rc = 14.

```
// JOB IESSCBAT
// LIBDEF *,SEARCH=PRD1.BASE
// OPTION DUMP,NOSYSDMP
// UPSI 1
// EXEC IESSCBAT,PARM='CODEPAGE=CP1047 SHOWERROR=YES SSL=YES'
1S54I  PHASE IESSCBAT IS TO BE FETCHED FROM PRD1.BASE
VSEScriptClient SSL Error: gsk_secure_soc_init: -3100
Error: ScriptInit: SSL handshake failed.
1S55I  LAST RETURN CODE WAS 0014
```

Solution:

Parameter `ALLOWCLIENTAUTH=YES` was not specified in IESSCBAT.

8 Debugging hints

In case of an error you can get useful trace information on VSE and on your workstation.

To trace the VSE Script Server, modify the runserver.bat file as follows:

```
%JAVA_EXEC% -Djavax.net.debug=all com.ibm.vse.script.VSEScriptServer %*
```

To trace the IESSCBAT job, you have to catalog a $SOCKDBG phase similar to the following:

```
* $$ JOB JNM=SOCKDBG,CLASS=0,DISP=D
// JOB $SOCKDBG
// OPTION CATAL
// LIBDEF *,CATALOG=PRD2.TCP15G
// EXEC ASMA90,SIZE=ASMA90
         PUNCH    ' PHASE $SOCKDBG,* '
$SOCKDBG CSECT
         SOCKDBG  CSECT,         GENERATE A PHASE                      X
               FL01=$DBGWLST,    MESSAGE TO SYSLST ONLY                X
               FL02=$DBGISON,    DEBUG IS ON                           X
               FL03=$DBGNONE,    NONE                                  X
               MSGT=$DBGALL,     ALL MESSAGES                          X
               DUMP=$DBGSDMP,    YES DIAGNOSTIC SDUMPS FOR IPNRBSDC    X
               SSLD=$DBGSDMP,    YES DIAGNOSTIC SDUMPS FOR IPCRYPTO    X
               CIAL=$DBGSDMP,    YES DIAGNOSTIC SDUMPS FOR IPDSCIAL    X
               CECZ=$DBGNONE     NO  DIAGNOSTIC SDUMPS FOR CIALCECZ
         END   $SOCKDBG
/*
// EXEC LNKEDT,SIZE=512K
/*
/&
* $$ EOJ
```

Then add the following lines to the IESSCBAT job:

```
// OPTION DUMP,NOSYSDMP
// upsi 1
```

9 More information

You can find more information in these books and web sites:

z/VSE e-business Connectors User's Guide
http://www.ibm.com/servers/eserver/zseries/zvse/documentation/#conn

z/VSE Administration
http://www.ibm.com/servers/eserver/zseries/zvse/documentation/#vse

Redbook: Security on IBM z/VSE, SG24-7691
http://www.redbooks.ibm.com/abstracts/sg247691.html?Open

Download connector components from the VSE homepage
http://www.ibm.com/servers/eserver/zseries/zvse/downloads/

Various SSL and crypto related technical papers on the VSE homepage
http://www.ibm.com/servers/eserver/zseries/zvse/documentation/security.html#howto

TCP/IP and SSL related documentation on the CSI website
http://www.csi-international.com/download.htm