



Java Applet Integration Document for E-Lock Client

Version 3.5

E-Lock Technologies

Contents

<u>1</u>	<u>INTRODUCTION</u>	<u>3</u>
<u>2</u>	<u>CLIENT COMPONENT DETECTION & FLOW</u>	<u>3</u>
<u>3</u>	<u>FILE LIST</u>	<u>4</u>
<u>4</u>	<u>APPLET PARAMETER DESCRIPTION</u>	<u>4</u>
4.1	<PARAM NAME=NDocType VALUE="0">	4
4.2	<PARAM NAME=FormName VALUE="">	4
4.3	<PARAM NAME=BrowserType VALUE="">	4
4.4	<PARAM NAME=ZELOCKCLPage VALUE="">	5
4.5	<PARAM NAME=ZELOCKRCode VALUE="">	5
4.6	<PARAM NAME=ZELOCKOpType VALUE="">	5
4.7	<PARAM NAME=DataFiles VALUE="">	5
4.8	<PARAM NAME=ZELOCKOUTPUTFile VALUE="">	5
4.9	<PARAM NAME=ZELOCKLASTSig VALUE="">	5
4.10	<PARAM NAME=ZELOCKCLPolicy VALUE="">	5
4.11	<PARAM NAME=SigType VALUE="">	5
4.12	<PARAM NAME=ZELOCKBUUpload VALUE="">	5
4.13	<PARAM NAME=ZELOCKSUBMITURL VALUE="">	5
4.14	<PARAM NAME=OS VALUE="UNIX">	5
4.15	<PARAM NAME=ZELOCKINMEMORYMode VALUE="1">	6
<u>5</u>	<u>JAVASCRIPT FUNCTIONS ASSOCIATED WITH THE APPLETS</u>	<u>6</u>

1 INTRODUCTION

The encryption, signing, verification and decryption in non-windows platform with E-Lock Client are done with the help of java applets. These applet take some argument and operate accordingly.

The Applet is compitable with FireFox 3.0 & above.

2 CLIENT COMPONENT DETECTION & FLOW

The client components are detected using DetectVMScript.js, which launches an applet to check weather the client components are present on the machine. If they are not present those will be installed using firefox plugin.

Once the components are intalled properly you can use signer.js to perform the operations, the signer applet will use SignNEncrApplet to perform the PKI operations with provided options.

The DetectVMScript also configures your current firefox profile as default provider for signing. The firefox user profiles are generally stored at USER_HOME/.mozilla/firefox

NOTE:

If a user is first time visiting the page, the DetectVMScript will configure his profile as signing provider & installs the components for him.

When client components are installed the browser needs to be restarted two times at follwing steps:

1. When the xpi is installed (FireFox makes this madatory)
2. When the E-Lock web signing components are installed (This is required for client components to take effect).

Here browser will show restart message when restart is required.

If you are **non admin/root** user make sure you have write permission on JAVA_HOME & its subfolders.

JAVA_HOME is path of installed jre which is configured with your browser.

If you are using e-Token you need to manually configure the e-Token provider, in provider.properties file which is present in **USER_HOME/tellustech/properties** folder.

e.g.

```
tellus.provider.count=<total provider count>
tellus.provider.<provider no>=PKCS11
tellus.provider.<provider no>.count=1
tellus.provider.<provider no>.parameter.1.name=<Aladdin-eToken>
tellus.provider.<provider no>.parameter.1.location=<full path to the PKCS#11
driver, typically .so in case of Linux>
```

For more details on configuring signing provider look in to **E-Lock Java SDK Installation Guide**.

3 FILE LIST

The following files are required to perform the above operations in web-based environment.

1. DetectVMScript.js

This file is responsible for downloading and configuring E-Lock Client components in client's machine. You must include these two files to perform any operation.

2. Signer.js

This file is responsible for invoking signing, & communicating with applet.

3. ConnectClient.jsp

This file is responsible for sending the client keys to the client's machine from the server. The complete URL to this page is passed to the Applet.

NOTE:

1. You can modify provided javascript file as per your needs.
2. During installation of client components the browser will restart two times:
 - a. When browser plugin get installed, which is by default forced by firefox. Here the browser plugin will take effect after the restart.
 - b. When client component gets installed after browser plugin is installed. Here the component will take effect after browser restart.

4 APPLET PARAMETER DESCRIPTION

All the operation with E-Lock Client in web environment is done through a Java Applet named [SignNEncrApplet](#). This applet takes several parameters. They are described below –

4.1 <PARAM NAME=nDocType VALUE="0">

This paramet takes always a hardcoded value 0. It is used for calling javascript function for conveying the result in the same window

4.2 <PARAM NAME=FormName VALUE="">

FormName - name of the form, which is going to get submitted to the server.

4.3 <PARAM NAME=browserType VALUE="">

BrowserType. The valid values are IE, NS6

4.4 <PARAM NAME=ZElockCIPage VALUE="">

URL of the [ConnectClient.jsp](#) file in the server.

4.5 <PARAM NAME=ZElockRCode VALUE="">

The Reseller code of the product. In case of live version it is to be obtained from E-Lock.

4.6 <PARAM NAME=ZElockOpType VALUE="">

OperationType - 0 Signing, 1 Encryption, 2 Decryption

4.7 <PARAM NAME=DataFiles VALUE="">

DataFiles = file paths of the documents to be signed. Each file path is separated by ','
OR
DataFiles= data to be signed.

4.8 <PARAM NAME=ZElockOutputFile VALUE="">

OutFile = output file name. If this parameter is null, output file will be created in the same folder as that of datafile.

4.9 <PARAM NAME=ZElockLastSig VALUE="">

LastSig = BASE64 encoded PKCS#7 structure containing earlier signature in the workflow. It will be used in multiple signature scenarios.

4.10 <PARAM NAME=ZElockCIPolicy VALUE="">

client side policy mandating settings for encryption/signing. Parameter is optional.

4.11 <PARAM NAME=SigType VALUE="">

SigType = 1 attached, 0 detached

4.12 <PARAM NAME=ZElockbUpload VALUE="">

Checks wheather to upload the signed / encrypted file or not. To stop uploading the value should be 1.

4.13 <PARAM NAME=ZElockSubmitUrl VALUE="">

If the Upload parameter is not equal to 1, then the file is uploaded to the URL specified in this parameter.

4.14 <PARAM NAME=OS VALUE="UNIX">

This parameter indicates the operating system. It should be UNIX always.

4.15 <PARAM NAME=ZElockinMemoryMode VALUE="1">

This parameter decides wheather the data passed in <PARAM NAME=DataFiles VALUE=""> is a file or an in-memory data that is to be signed. Remember is field is mandatory. Value="1" indicates it is an in-memory data and value="0" indicated file(s).

5 JAVASCRIPT FUNCTIONS ASSOCIATED WITH THE APPLLET

submitformNS6 : Performs the submission of the Form from NS browser

Note: Please refer to the Javascript files for the definition of the above functions.