Integration of Symantec Protection Engine with IBM Sterling B2B Integrator and File Gateway

Authors:	Mehmet Cambaz
Version:	1.0
Date Created:	12.06.2015
Last Updated:	

Disclaimer:

All of the information is given "as-is" basis and author do not give any kind of guarantee of any sort, use it as your own risk.

Document Revision History

Revision	Date	Description	Author
1.0	12.06.2015	First Draft	Mehmet Cambaz

Document Approval History

Name	Signature	Date	Version Approved/Comments

Reference Documents

Document Name	Description	Owner	Location

Table of Contents

1. Introduction
2. Symantec Protection Engine Integration Details
3. Business Process Implementation on Sterling B2B Integrator and File Gateway

1. Introduction

This document’s purpose is to give helpful information on how to integrate with Symantec Protection Engine.

- Symantec Protection Engine is a cloud based antivirus application which checks the file sent for harmful information. Used version is 7.5
- IBM Sterling B2B Integrator, IBM Sterling File Gateway are products of IBM which are used for file-based-integration middleware which have adapters FTP/SFTP/FTPS/C:D etc. and have capabilities of business processes developed by BPML. Used version is 5.2.5

2. Symantec Protection Engine Integration Details

Symantec Protection Engine can be integrated via C/C#/Java programming languages with using the SDK provided.

This is the Java sample, other languages are similar.

Symantec(TM) Protection Engine Software Development Kit - Java

Version: 7.5

The SDK requires Java Runtime Environment (JRE) 1.6 Update 25 or later.

Compiling example code

Execute the below command form the path which contains

JavaAPICheck.java

On Windows/Linux/Solaris

```
javac -cp <library path> JavaAPICheck.java
```

Running example code

Usage on Windows

```
java -cp .;<library path> JavaAPICheck.java -file:<file to be scanned>
```

Usage on Linux/Solaris

```
java -cp .:<library path> JavaAPICheck.java -file:<file to be scanned>
```

Notes

1. <library path> is the path to the API library, SymJavaAPI.jar.

2. <file to be scanned> is the name of the file to be scanned.

3. The JAVA_HOME needs to be set to <Java install dir>

Sample usage

Red Hat Linux

```
/jdk/ibm-java-x86_64-70/bin/java -cp .:SymJavaAPI.jar JavaAPICheck -
file:1.txt -streambased:1 server:10.230.200.159:1344 -streamFileLocal:1
```

Windows

```
java -cp .;SymJavaAPI.jar JavaAPICheck -file:1.txt
```

Clean response

```
-----------------------------------------------------------------------
Scanning file .........................................................
-----------------------------------------------------------------------
Results ...............................................................
-----------------------------------------------------------------------
File Scanned            : 1.txt
Scan Policy             : DEFAULT
File Status             : CLEAN
Total Infection         : 0
Virus Def Date          : Mon May 25 00:00:00 EEST 2015
Virus Def Revision No   : 001
Symantec Protection Engine IP           : 10.230.200.159
Symantec Protection Engine Port : 1344
Symantec Protection Engine Port : Able to connect
```

Harmful response

```
-----------------------------------------------------------------------
Scanning file .........................................................
-----------------------------------------------------------------------
Results ...............................................................
-----------------------------------------------------------------------
File Scanned            : /data1/SOURCE/eicar.com
Scan Policy             : DEFAULT
File Status             : INFECTED_REPLACED
Total Infection         : 1
Virus Def Date          : Mon May 25 00:00:00 EEST 2015
Virus Def Revision No   : 001
File Name               : index.html
Violation Name          : EICAR Test String
Non Viral Threat Category :
Violation Id            : 11101
Disposition             : 0
File Unscannable                : false
Symantec Protection Engine IP           : 10.230.200.159
Symantec Protection Engine Port : 1344
Symantec Protection Engine Port : Able to connect
```

3. Business Process Implementation on Sterling B2B Integrator and File Gateway

This is a sample business process which checks the filesize and decides whether to send the file for antivirus inspection, if the file is infected stops the business process.

BPML

```
<process name="AntiVirus">

  <rule name="checkCleanFile">

     <condition>scanStatusFlag/text()=scanCleanFlag/text()</condition>

  </rule>


  <rule name="fileSizeOkForScan">

     <condition> permittedFileSize/text() &gt; originalFileSize/text()</condition>

  </rule>


<sequence name="Main Process">

<assign to="permittedFileSize">1048576</assign>

<assign to="scanDirectory">/data1/SOURCE/scanEngine</assign>

<assign to="javaCommPath">/data1/SOURCE/sfg/B2Bi_5020500/jdk/ibm-java-x86_64-
70/bin/java</assign>


<assign to="scancomm1"> -cp .:SymJavaAPI.jar JavaAPICheck -file:</assign>

<assign to="scanServer"> -streambased:1 server:10.230.200.159:1344 -streamFileLocal:1 -
output:</assign>


<operation name="GetFileSizeFromDocument">

<participant name="GetDocumentInfoService"/>

<output message="xout">

<assign to="." from="*"/>

</output>

<input message="xin">

<assign to="originalFileSize" from="DocumentBodyLength/text()"/>

<assign to="fileName" from="DocumentName/text()"/>

</input>

</operation>


<choice name="CheckTheFileSize">
```

```
  <select>
  <case ref="fileSizeOkForScan" activity="proceedForScanning"/>
</select>
<sequence name="proceedForScanning">

<operation name="File System Adapter">
<participant name="fsa_loca"/>
<output message="FileSystemInputMessage">
<assign to="Action">FS_EXTRACT</assign>
<assign to="extractionFolder" from="scanDirectory/text()"/>
<assign to="fileModTimeThreshold">5</assign>
<assign to="." from="*"/>
</output>
<input message="inmsg">
<assign to="." from="*"/>
</input>
</operation>

<assign to="fileNameOut" from="concat(All_Res/DocumentName/text(),'_out')"/>

<assign to="commnd1"
from="concat(javaCommPath/text(),scancomm1/text(),fileName/text(),scanServer/text(),fileNameOu
t/text())"/>

<operation name="CommandLine Operation">
<participant name="CommandLineAdapter2"/>
<output message="CmdLine2InputMessage">
  <assign to="." from="*"/>
  <assign to="workingDir" from="scanDirectory/text()"/>
  <assign to="cmdLine" from="commnd1/text()"/>
</output>
<input message="inmsg">
<assign to="." from="*"/>
<assign to="comndStatus" from="Status_Rpt('StatusReport')"></assign>
</input>
</operation>

<assign to="scanStatus" from="substring-after(comndStatus/StatusReport/text(),&quot;File
Status&quot;)"/>
<assign to="scanStatusFlag" from="substring(scanStatus,5,5)"/>
```

```
<assign to="scanCleanFlag">CLEAN</assign>

<choice name="CheckTheVirus">
  <select>
  <case ref="checkCleanFile" activity="cleanFile_Process"/>
  <case ref="checkCleanFile" activity="infectedFile_StopProcess" negative="true"/>
</select>
<sequence name="cleanFile_Process">
<assign to="commnd2" from="concat('rm -f ',fileName/text(),' ',fileNameOut/text())"/>

<operation name="CommandLine Operation">
<participant name="CommandLineAdapter2"/>
<output message="CmdLine2InputMessage">
  <assign to="." from="*"/>
  <assign to="workingDir" from="scanDirectory/text()"/>
  <assign to="cmdLine" from="commnd2/text()"/>
</output>
<input message="inmsg">
<assign to="." from="*"/>
</input>
</operation>
</sequence>

<sequence name="infectedFile_StopProcess">
<operation name="generateException">
   <participant name="BPExceptionService"/>
        <output message="Xout">
        <assign to="exceptionCode">Virus Check Failed</assign>
        <assign to="." from="*"/>
     </output>
        <input message="Xin">
         <assign to="." from="*"/>
         </input>
</operation>
</sequence>
</choice>

<assign to="processStatus">InsideFileCheck</assign>
</sequence>
```

```
</choice>

<assign to="processStatus">endOfProcess</assign>
</sequence>
</process>
```