SSL VPN Onboarding Developer’s Guide

IP Office Release 9.1

Issue 0.1

November 2014

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

“Documentation” means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya may generally make available to users of its products and Hosted Services. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original Published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on Avaya hardware and software. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya’s standard warranty language, as well as information regarding support for this product while under warranty is available to Avaya customers and other parties through the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Please note that if you acquired the product(s) from an authorized Avaya Channel

Partner outside of the United States and Canada, the warranty is provided to you by said Avaya Channel Partner and not by Avaya.

Licenses

THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYAWEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO

OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AVAYA CHANNEL PARTNER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA CHANNEL PARTNER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS “YOU” AND “END USER”), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE

AVAYA AFFILIATE (“AVAYA”).

Avaya grants you a license within the scope of the license types described below, with the exception of Heritage Nortel Software, for which the scope of the license is detailed below. Where the order documentation does not expressly identify a license type, the applicable license will be a Designated System License. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation or other materials available to you. “Designated Processor” means a single stand-alone computing device. “Server” means a Designated Processor that hosts a software application to be accessed by multiple users.

License type(s)

Named User License (NU). You may: (i) install and use the Software on a single Designated Processor or Server per authorized Named User (defined below); or (ii) install and use the Software on a Server so long as only authorized Named Users access and use the Software. “Named User”, means a user or device that has been expressly authorized by Avaya to access and use the Software. At Avaya’s sole discretion, a “Named User” may be, without limitation, designated by name, corporate function (e.g., webmaster or helpdesk), an e-mail or voice mail account in the name of a person or corporate function, or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software.

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, Hosted Service, or hardware provided by Avaya. All content on this site, the documentation, Hosted Service, and the Product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may

not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Third Party Components

“Third Party Components” mean certain software programs or portions thereof included in the Software or Hosted Service may contain software (including open source software) distributed under third party agreements (“Third Party Components”), which contain terms regarding the rights to use certain portions of the Software (“Third Party Terms”). As required, information regarding distributed Linux OS source code (for those Products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the Documentation or on Avaya’s website at:

http://support.avaya.com/Copyright or such successor site as designated by Avaya. You agree to the Third Party Terms for any such Third Party Components.

THIS PRODUCT IS LICENSED UNDER THE AVC PATENT PORTFOLIO LICENSE FOR THE PERSONAL USE OF A CONSUMER OR OTHER USES IN WHICH IT DOES NOT RECEIVE REMUNERATION TO (i) ENCODE VIDEO IN COMPLIANCE WITH THE AVC STANDARD (“AVC VIDEO”) AND/OR (ii) DECODE AVC VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSED TO PROVIDE AVC VIDEO. NO LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION MAY BE OBTAINED FROM MPEG LA, L.L.C. SEE HTTP://WWW.MPEGLA.COM.

Note to Service Provider

The Product or Hosted Service may use Third Party Components subject to Third Party Terms that do not allow hosting and require a Service Provider to be independently licensed for such purpose. It is your responsibility to obtain such licensing.

Preventing Toll Fraud

“Toll Fraud” is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Toll Fraud intervention

If you suspect that you are being victimized by Toll Fraud and you need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Suspected security vulnerabilities with Avaya products should be reported to Avaya by sending mail to: securityalerts@avaya.com.

Trademarks

The trademarks, logos and service marks (“Marks”) displayed in this site, the Documentation, Hosted Service(s), and Product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation, Hosted Service(s) and Product(s) should be construed as granting, by implication, estoppel, or otherwise, any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.

Avaya is a registered trademark of Avaya Inc.

All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

All non-Avaya trademarks are the property of their respective owners, and “Linux” is a registered trademark of Linus Torvalds.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya.

Contact Avaya Support

See the Avaya Support website: http://support.avaya.com for Product or Hosted Service notices and articles, or to report a problem with your Avaya Product or Hosted Service. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: http://support.avaya.com (or such successor site as designated by Avaya), scroll to the bottom of the page, and select Contact Avaya Support.

Contents

Chapter 1: Introduction

Purpose

This document describes the ‘sslvpnOnboarding’ JAVA tool and APIs that can be used to facilitate the creation of onboarding xml files that are installed in IP Office for SSL VPN remote support and monitoring.

Intended audience

This document is designed for Business Partners and Avaya DevConnect Members who wish to use the ‘sslvpnOnboarding’ JAVA directly as a DOS batch file or reference the JAVA APIs for inclusion in a Web Server.

Related resources

Documentation

The following table lists the related documents to SSL VPN and Onboarding on IP Office. Download the documents from the Avaya Support website at http://support.avaya.com.

Title	Description
Avaya IP Office SSL VPN Solutions Guide	Describes the SSL VPN solution for IP Office including the installation of Avaya VPN Gateway. This document also describes the onboarding process and contains configuration information that help detail the contents of the onboarding xml template sample files in the SDK.
SSL VPN Onboarding SDK KT	Knowledge Transfer slides (consult Avaya DevConnect)
Avaya Mentor - IP Office R8.1 SSL/VPN On-Boarding	YouTube video explaining the IP Office Onboarding process in conjunction with the Global Registration Tool (GRT).

Training

The following courses are available on the Avaya Learning website at www.avaya-learning.com. After logging into the website, enter the course code or the course title in the Search field and click Go to search for the course.

Course code	Course title

Avaya Mentor videos

Avaya Mentor videos are available to provide technical content on how to install, configure, and troubleshoot Avaya products.

Videos are available on the Avaya support site, listed under the video document type, and on the Avaya-run channel on YouTube.

To find videos on the Avaya support site, select the product name, and check the videos check box to see a list of available videos.

Note: Videos are not available for all products.

To find the Avaya Mentor videos on YouTube, go to http://www.youtube.com/AvayaMentor and perform one of the following actions:

- Enter a key word or key words in the Search Channel to search for a specific product or topic.
- Scroll down Playlists, and click the name of a topic to see the available list of videos posted on the site.

Support

Visit the Avaya Support website at http://support.avaya.com for the most up-to-date documentation, product notices, and knowledge articles. You can also search for release notes, downloads, and resolutions to issues. Use the online service request system to create a service request. Chat with live agents to get answers to questions, or request an agent to connect you to a support team if an issue requires additional expertise.

Chapter 2: Overview

Introduction

‘**slvpnOnboarding’** is a JAVA based tool that is used to generate a signed onboarding xml file that can be uploaded to IP Office to create an SSL VPN service including the configuration of other relevant components including the installation of the AVG self-signed certificate or the CA certificate that signed the AVG identity certificate. The tool takes a properties file, a template xml file and a key file as input and generates a signed xml file. The key file can be either the PEM encoded certificate data of IP Office or an IP Office inventory xml file which contains the PEM encoded certificate data.

Installation

The tool is delivered as a zip file (SSLVPN_Onboarding_SDK.zip). It requires java 1.6 or newer to be installed on the PC. Unzip the file in a directory on the PC. The unzipped file will contain the following:

- sslvpnOnboarding.bat (DOS batch script for invoking the JAVA tool),
- sslvpnOnboarding.jar (JAVA executable),
- SSLVPNOnboardingDevelopersGuide.pdf (this document),
- templates (directory containing xml template references and a properties file)
- examples (directory containing an inventory xml file and a how-to JAVA example)

Chapter 3: Usage

DOS Batch Script

Place the **SSLVPN_Onboarding_SDK.zip** file in a new folder and unzip. Open a DOS command prompt window. Navigate to the directory that you have unzipped the **SSLVPN_Onboarding_SDK.zip** contents. Execute the tool by entering the name of the batch file with the desired parameters.

sslvpnOnboarding

```
Usage Format-a:

      sslvpnOnboarding -t <template file> -i <inventory file> -o <output file>
      [-p <properties file>]
```

```
Usage Format-b:

      sslvpnOnboarding -t <template file> -k <key PEM file> -o <output file> [-p
      <properties file>]
```

There are two allowed usage formats: Usage format (a) and usage format (b). The difference between the two formats is how the public key is specified. In usage format (a) the public key is specified in the IP Office Inventory xml file. In usage format (b) the IP Office public key is explicitly specified as a text file in PEM format.

The attributes are defined as follows:

-t <template file> - Mandatory file. Specify the path of the SSL VPN template xml file. This file describes all of the SSL VPN services and related attributes that must be configured on the IP Office system.

-i <inventory file> - Specifies the IP Office system's public key inside the XML tags. This inventory XML data must be downloaded from the IP Office system you wish to onboard.

-k<key PEM file> - Specifies the IP Office system's public key in a PEM formatted text file (used as an alternative to the '-i <inventory file>' option.)

-o <output file> - Mandatory file. Specifies the path of the output file. This filename will be used to generate the signed onboarding xml file

-p <properties file> - Optional file. Specify the path of the properties file. Properties from this file will be used to overlay the values of corresponding attributes in the template xml file.

The program will print 0 if the operation was successful.

If an error occurs during processing, an error code and text explanation will be displayed

The following errors can be generated.

1 - Template xml file not specified
2 - Either a Key file (-k) or an Inventory file (-i) must be specified
3 - Output file not specified
4 - Unable to open properties file
5 - Unable to open template file
6 - Unable to read key file
7 - Unable to create output file
8 - Invalid properties file. Invalid field value detected
9 - Invalid properties file. Service mismatch – the number of SSL VPN services defined in template xml file does not match the services defined in the properties file
10 - Mandatory field missing in the properties file
11 - Invalid template xml file. Invalid field value detected
12 - Template xml file does not define a service
13 - Invalid key file
14 - Internal error
15 - Identity certificate cannot be found in inventory file
16 - Invalid identity certificate
17 - Unable to open inventory file

JAVA APIs

The JAVA tool is implemented as a JAVA class with two public static methods that are used to read the public key extracted from the IP Office certificate PEM file and to generate the onboarding xml file. The class is com.avaya.sslvpn.sslvpnGen.

API: initPubKey

The following method is used to read the public key from either the inventory xml file or from a PEM file containing the IP Office X.509 identity certificate. The two parameters are mutually exclusive so one parameter should be null.

public static PublicKey initPubKey(String inventoryFile, String keyFile)

Parameters:

Field Name	Description
String inventoryFile	Specifies the path of the inventory xml file previously downloaded from IP Office. The inventory xml file contains the IP Office certificate from which the public key is extracted. If both parameters are provided, then the inventory xml file path is processed and the keyFile parameter is ignored.
String keyFile	Specifies the path of the PEM encoded certificate of IP Office. Note: The provided file must have its EOF on the last line of the “-----END CERTIFICATE-----” string. This parameter is only processed if a null inventory file string is provided.

Exceptions:

Field Name	Description
sslvpnException	Thrown to indicate an error. See previous section for error number definitions. The exception message will give a text message for the error.
Exception	Thrown if an unexpected error occurs.

API: generateSSLVPN

The following method is used to generate the file.

public static void generateSSLVPN(PublicKey publicKey, String sslTemplateFile, String outputFile, String propFile, boolean checkDefaultServerName)

Parameters:

Field Name	Description
PublicKey publicKey	The public key extracted from the identity certificate of IP Office.
String sslTemplateFile	Specifies the path of the template xml file. This file contains the SSL VPN settings for the system. See the two template examples in the SDK.
String outputFile	Specifies the path of the output file. This filename will be used to generate the signed onboarding xml file.
String propFile	Specifies the path of the properties file. This file contains the SSL VPN service parameters that are specific to an IP Office.
boolean checkDefaultServerName	When set to true forces extra checking to make sure the “ServerAddress” field value is not empty in the properties file (when provided) and in the xml template.

Exceptions:

Field Name	Description
sslvpnException	Thrown to indicate an error. See previous section for error number definitions. The exception message will give a text message for the error.
Exception	Thrown if an unexpected error occurs.

Code Snippet

The SDK contains a Hello World JAVA example along with a set of example files, templates and properties. The SDK also contains a test_onboarding.bat DOS executable batch script you can invoke from a Windows DOS command (or double-clicking on the batch file) to test the Hello World example that's been compiled. For convenience, this JAVA example follows.

```java
// Adjust your package name to fit your environment.
package com.avaya.sslvpn;

// Add this import:
//import com.avaya.sslvpn;

import java.security.PublicKey;

public class OnboardingHelloWorld
{
    private static String INVENTORY_FILE = "examples/inventory.xml";
    private static String PUBLIC_KEY_FILE = "examples/public_key.txt";
    private static String PROPERTIES_FILE = "templates/sslvpn.properties";
    private static String XML_TEMPLATE_FILE = "templates/sslvpn_template.xml";
    private static String HELLO_WORLD_FILE1 = "examples/hello_world_onboarding1.xml";
    private static String HELLO_WORLD_FILE2 = "examples/hello_world_onboarding2.xml";

    public static void main(String[] args)
    {
        try
        {
            PublicKey publicKey = sslvpnOnboarding.initPubKey(INVENTORY_FILE, null);

            if (publicKey != null)
            {
                sslvpnOnboarding
                        .generateSSLVPN(publicKey, XML_TEMPLATE_FILE, HELLO_WORLD_FILE1,
PROPERTIES_FILE, false);
                System.out.println(SSLVPN_RESULT.SUCCESS);
            }
            else
            {
                System.out.println("The public key could not be parsed.");
            }

            publicKey = sslvpnOnboarding.initPubKey(null, PUBLIC_KEY_FILE);

            if (publicKey != null)
            {
                sslvpnOnboarding
                        .generateSSLVPN(publicKey, XML_TEMPLATE_FILE, HELLO_WORLD_FILE2,
PROPERTIES_FILE, false);
                System.out.println(SSLVPN_RESULT.SUCCESS);
            }
            else
            {
                System.out.println("The public key could not be parsed.");
            }
        }
        catch (sslvpnException e)
        {
            System.out.println(e.getMessage());
            System.out.println(e.getError());
        }
        catch (Exception e)
        {
            System.out.println(e.getMessage());
        }
    }
}
```

Properties File

The properties file contains the SSL VPN parameters that are specific to an IP Office. If a field is specified in the file, the corresponding element in the template xml file will be updated.

The properties file has the following format:

```
#Mandatory fields
soldto=45

#Service Name = VPN 1
# Mandatory fields
VPN_1.AccountName=acct1
VPN_1.Password=acc1pass

# Optional fields
VPN_1.SessionMode=
VPN_1.InFallBack=
VPN_1.ServerAddress=10.136.66.58

#Service Name = VPN_2
# Mandatory fields
VPN_2.AccountName=acc2
VPN_2.Password=acc2pass

# Optional fields
VPN_2.SessionMode=
VPN_2.InFallBack=
VPN-2.ServerAddress=
```

Soldto – Mandatory field; It gives the customer reference number (e.g. SOLD-TO) of the IP Office system. This is a numeric value up to 10 digits. It is used as a reference number.

Each SSL VPN service defined in the template xml file can have a service section in the properties. The nomenclature used for each property is

<ServiceName>.<AttributeName>

For example, in the example property “VPN_1.AccountName=sa1”, the “VPN_1” is the name of the service that appears in the TEMPLATE file. The “AccountName=sa1” indicates that VPN_1’s AccountName is to be overwritten with the value “sa1”.

The first section under services **#Mandatory fields**, lists the fields that are required. If these fields are missing or there is no value supplied then it will cause an error.

<ServiceName>.AccountName – The service account name. This field will be encrypted in the generated xml file.

<ServiceName>.Password – The service password. This field will be encrypted in the generated xml file.

The second section under services **#Optional fields**, lists the fields that are optional.

SessionMode – The session mode for the service. Valid values are “always_on”, “event_based” and “periodic”. Note that “always_on” is the only supported value.

InFallBack – The fallback setting for the service. Valid values are “true” and “false”.

ServerAddress – The Avaya VPN Gateway (AVG) server address for the service which can be either an IPv4 address or Fully Qualified Domain Name (FQDN).

Template XML File

The template xml file contains the IP Office configuration settings for SSL VPN. It has the default settings for a service provider. This file is used with the properties file to generate an SSL VPN xml file for a specific IP Office. The parameters in the properties field will override the field values of the template xml file. For this reason, some of the fields in the template xml file are better left empty such as AccountName and Password since they will be specified in the properties file.

Please consult the two sslvpn_template.xml and advanced_sslvpn_template.xml files included in the SDK. To better understand the relationship of the configuration components inside the xml files, please consult the SSL VPN Solutions Guide in reference.

Key File

The key file is used to contain the encryption key that will be used by the tool or API to encrypt the service account name and password. It will also be used in the signing of the generated onboarding xml file. The public key is either contained in the inventory xml file as part of the identity certificate tag or in a PEM encoded certificate file.

Inventory XML File

The inventory file contains a listing of all of the hardware and software components on the IP Office. It also contains the public key. The inventory file is downloaded from the IP Office.

When specified using the “-i” option the tool will parse the inventory file, extract the public key and use the key for encrypting and signing.

Example format of an inventory file:

```
<?xml version="1.0" encoding="UTF-8"?>
<response status="1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="sslvpn_inventory_9_1_0.xsd">
<header>
<definition>IP OFFICE_HW_INVENTORY</definition>
<version>9.1</version>
</header>
<data>
<inventory>
…
<IdentityCertificates>
<IdentityCertificate type="DefaultIdentityCertificate">
-----BEGIN CERTIFICATE-----
MIIEWjCCA0KgAwIBAgIUZr0mgXsvNYKA3eJXk8/rahGr/5swDQYJKoZIhvcN
AQELBQAwgagxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpOZXcgSmVyc2V5MRYw
FAYDVQQHEw1CYXNraW5nIFJpZGdlMRIwEAYDVQQKEwlBdmF5YSBJbmMxDDAK
BgNVBAsTA0dDUzEoMCYGA1UEAxMfaXBvZmZpY2UtMDBlMDA3MDVlYjc0LmF2
YXlhLmNvbTEgMB4GCSqGSIb3DQEJARYRc3VwcG9ydEBhdmF5YS5jb20wHhcN
MTQxMDI2MjEzMTQ2WhcNMjExMDI2MjEzMTQ2WjCBqDELMAkGA1UEBhMCVVMx
EzARBgNVBAgTCk5ldyBKZXJzZXkxFjAUBgNVBAcTDUJhc2tpbmcgUmlkZ2Ux
```

```
EjAQBgNVBAoTCUF2YXlhIEluYzEMMAoGA1UECxMDR0NTMSgwJgYDVQQDEx9p
cG9mZmljZS0wMGUwMDcwNWViNzQuYXZheWEuY29tMSAwHgYJKoZIhvcNAQkB
FhFzdXBwb3J0QGF2YXlhLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBALXw8eblL5nquzPQ9ccBdcweu2jkJyUs5fyGvG68a4cxZrrwOUV3
a/VId4u3MlJw3VHFS/X4Opnxic3xyL0tKEo69U3gqHrfCgLe4sVJtH3WvEBZ
iNGFNJjTCCWHZVVXHZvkNTsTnli4PRq0TznymUTsPeuthZwUn9gmBK1h2e/a
5DMOsrHFxOOOkp8H9ilGzF0c8xKG0rd1xKu4Ny+ipI/3YtVsPF7hQb/qgGAw
8w2g35d5lTUzSXl6kPm7bho5ZfxOXmqjpiBQRsq1oV0NkspPt589M4NIHPko
XeQpNoXYCF7h+xIdOFKOqBYS2dzAyw5hd3R52ZF+VDDHOgxJRn8CAwEAAaN6
MHgwEgYDVR0TAQH/BAgwBgEB/wIBADALBgNVHQ8EBAMCAvwwHQYDVR0lBBYw
FAYIKwYBBQUHAwEGCCsGAQUFBwMCMDYGA1UdEQQvMC2CH2lwb2ZmaWNlLTAw
ZTAwNzA1ZWI3NC5hdmF5YS5jb22HBMCoKgGHBAqITDcwDQYJKoZIhvcNAQEL
BQADggEBAJ/eQTSIir10Z6oF7lakz1gtAafbbRMIuUDBuCkgGgB3GoOsNftc
DpYj8rIk3Af+G2J00kzGmsK4CHlnqNraon1L9i9OsGmjD5REi/yPV4c4/ES8
nkJeIhi3pTiECRTK0twIGigkVGDWOqxCj7CMKW4U7WJiFV87FMvhRV6Dk+MS
sIaqH16sKLm+00zgwrMrmjDxUk1cmJTvcj7EQGaTQYNPBE8AW0YifMGrA8+G
TlMs52A2i3NDUvAIl5AndbjcuDfAfK+4kvQinBWZcmxwHLxD5zEdNfQ1l6u1
lPCH/PBCOTdqKPDYAr4UDK2WY7hfSt1L6OcFwzYCM/gc4nzLJsE=
-----END CERTIFICATE-----
</IdentityCertificate>
</IdentityCertificates>
</inventory>
</data>
</response>
```

Key File

This is just a text file that contains the IP Office’s public key in a PEM format.

Important note: The provided file must have its EOF on the last line of the “-----END CERTIFICATE-----” string. If not the case, the file parser will throw an error.

An example format of the key PEM file is as follows:

```
-----BEGIN CERTIFICATE-----
MIIEWjCCA0KgAwIBAgIUZr0mgXsvNYKA3eJXk8/rahGr/5swDQYJKoZIhvcN
```

```
AQELBQAwgagxCzAJBgNVBAYTAlVTMRMwEQYDVQQIEwpOZXcgSmVyc2V5MRYw
FAYDVQQHEw1CYXNraW5nIFJpZGdlMRIwEAYDVQQKEwlBdmF5YSBJbmMxDDAK
BgNVBAsTA0dDUzEoMCYGA1UEAxMfaXBvZmZpY2UtMDBlMDA3MDVlYjc0LmF2
YXlhLmNvbTEgMB4GCSqGSIb3DQEJARYRc3VwcG9ydEBhdmF5YS5jb20wHhcN
MTQxMDI2MjEzMTQ2WhcNMjExMDI2MjEzMTQ2WjCBqDELMAkGA1UEBhMCVVMx
EzARBgNVBAgTCk5ldyBKZXJzZXkxFjAUBgNVBAcTDUJhc2tpbmcgUmlkZ2Ux
EjAQBgNVBAoTCUF2YXlhIEluYzEMMAoGA1UECxMDR0NTMSgwJgYDVQQDEx9p
cG9mZmljZS0wMGUwMDcwNWViNzQuYXZheWEuY29tMSAwHgYJKoZIhvcNAQkB
FhFzdXBwb3J0QGF2YXlhLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBALXw8eblL5nquzPQ9ccBdcweu2jkJyUs5fyGvG68a4cxZrrwOUV3
a/VId4u3MlJw3VHFS/X4Opnxic3xyL0tKEo69U3gqHrfCgLe4sVJtH3WvEBZ
iNGFNJjTCCWHZVVXHZvkNTsTnli4PRq0TznymUTsPeuthZwUn9gmBK1h2e/a
5DMOsrHFxOOOkp8H9ilGzF0c8xKG0rd1xKu4Ny+ipI/3YtVsPF7hQb/qgGAw
8w2g35d5lTUzSXl6kPm7bho5ZfxOXmqjpiBQRsq1oV0NkspPt589M4NIHPko
XeQpNoXYCF7h+xIdOFKOqBYS2dzAyw5hd3R52ZF+VDDHOgxJRn8CAwEAAaN6
MHgwEgYDVR0TAQH/BAgwBgEB/wIBADALBgNVHQ8EBAMCAvwwHQYDVR0lBBYw
FAYIKwYBBQUHAwEGCCsGAQUFBwMCMDYGA1UdEQQvMC2CH2lwb2ZmaWNlLTAw
ZTAwNzA1ZWI3NC5hdmF5YS5jb22HBMCoKgGHBAqlTDcwDQYJKoZIhvcNAQEL
BQADggEBAJ/eQTSIir10Z6oF7lakz1gtAafbbRMIuUDBuCkgGgB3GoOsNftc
DpYj8rIk3Af+G2J00kzGmsK4CHlnqNraon1L9i9OsGmjD5REi/yPV4c4/ES8
nkJeIhi3pTiECRTK0twIGigkVGDWOqxCj7CMKW4U7WJiFV87FMvhRV6Dk+MS
sIaqH16sKLm+00zgwrMrmjDxUk1cmJTvcj7EQGaTQYNPBE8AW0YifMGrA8+G
TlMs52A2i3NDUvAIl5AndbjcuDfAfK+4kvQinBWZcmxwHLxD5zEdNfQ1l6u1
lPCH/PBCOTdqKPDYAr4UDK2WY7hfSt1L6OcFwzYCM/gc4nzLJsE=
-----END CERTIFICATE-----
```

Example

Two example SSL VPN template XML files and their respective properties files have been provided. They are located in the **templates** folder. Those two examples illustrate a basic and an advanced onboarding example.

To be useable, the example template must be updated with the AVG self-signed certificate or the CA certificate that signed the AVG identity certificate. Please edit the template XML and replace the example certificate with the actual value. PEM format is expected.

In addition, the respective properties file will need to be modified to reflect the correct AVG Account Name, Password and AVG Server address. The AVG Server address may be an FQDN or an IP ADDRESS string. SOLDTO should be the number associated with the purchase or used as a reference number.

Note: If the advanced template is used, then more modifications may be required to achieve the desired settings.

In the following example the required changes to the properties & basic template are shown in bold-italics font.

```
# Example sslvpn.properties
#

soldto=1234567890
VPN_Service.Name=BP_SUPPORT
VPN_Service.AccountName=SA1
VPN_Service.Password=sa1
VPN_Service.ServerAddress=example_avg_server_FQDN_name_or_ip_address.com
```

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<data>
<on_boarding type="SSL_VPN">
<soldto></soldto>
..
<ws_object>
 <Service>
  <Name>VPN_Service</Name>
  <AccountName></AccountName>
  <Password></Password>
   ..
  <ServerType>avg</ServerType>
```

```
    <ServerPortNumber>443</ServerPortNumber>
...
    <InFallBack>false</InFallBack>
  </Service>
</ws_object>
<ws_object>
  <ShortCode>
    <Code>*77520</Code>
    <TelephoneNumber>"VPN_Service"</TelephoneNumber>
    <Feature>SetHuntGroupNightService</Feature>
  </ShortCode>
</ws_object>
<ws_object>
   <Certificates>
      <TrustedCertificateStore>
         <CertificateData>
-----BEGIN CERTIFICATE-----
MIIDHzCCAoigAwIBAgIBBTANBg ... etc ...oDELMAkGA1UEBhMCVVMx
-----END CERTIFICATE-----
          </CertificateData>
      </TrustedCertificateStore>
   </Certificates>
</ws_object>
</on_boarding>
```

Chapter 4: Appendix A: IP Office 9.1.0.0 Release Notes

This is the first release for the introduction of the SDK.

To query the version of the JAVA tool simply invoke the DOS script.

For example:

```
C:\temp_onb>sslvpnOnboarding.bat

C:\temp_onb>java -cp sslvpnOnboarding.jar com.avaya.sslvpn/sslvpnOnboarding
sslvpnOnboarding - 9.1.0.0 build 79
```

Index

No index entries found.