

---

# ONESPAN SIGN SIGNATURE IMPORT TOOL

---

User Guide

September 14, 2020  
OneSpan

## Contents

General.....	3
Prerequisites .....	3
Usage.....	3
Command Line .....	3
Integrated Approach.....	4
Questions or Issues? .....	5

## General

This tool allows a Signature to be imported into an OneSpan Sign account from an image stored in a file or in another storage location.

## Prerequisites

- You have an account on an OneSpan Sign instance and an associated API Key
- Java installed
- Signature image file (.jpg, .bmp and .png files have been tested and work well)

## Usage

This tool can be used either via the command line or Integrated into an existing java application.

### Command Line

In the directory where the `esl-signature-import-tool.jar` is located, you can run command lines and directly update sender signature. The jar file provides four parameter patterns:

#### Pattern 1: API Key + Sender Email

```
java -jar esl-signature-import-tool.jar --filename mySignature.png --rgbthreshold 140 --apiurl  
https://sandbox.esignlive.com --apikey xxxxxxxxxx --senderemail onespansign@esignlive.com
```

Where:

“--filename” is the path to the signature file to be imported into the OneSpan Sign account

“--rgbthreshold” is the threshold of average RGB value to tell whether a pixel is black or white ( $0 \leq \text{rgbthreshold} \leq 255$ , suggested to set around 128), adjust this value to get better display effect of your signature depending on the noise of your input image

“--apiurl” is the URL of OneSpan Sign where your account exists. i.e. <https://sandbox.esignlive.com>

“--apikey” is the API key that is specific to your account that allows you to access the API

“--senderemail” is the email of the account for the sender you wish to update the signature for.

#### Pattern 2: Client App ID & Secret + Sender Email

```
java -jar esl-signature-import-tool.jar --filename mySignature.png --rgbthreshold 140 --apiurl  
https://sandbox.esignlive.com --clientappid MyDemoApp --clientappsecret xxxxxxxxxx --  
senderemail onespansign@esignlive.com
```

Where:

“--clientappid” is the ID of your Client Application

“--clientappsecret” is the secret of your Client Application

**Pattern 3: API Key / Client App Token + Sender ID**

```
java -jar esl-signature-import-tool.jar --filename mySignature.png --rgbthreshold 140 --apiurl  
https://sandbox.esignlive.com --apikey xxxxxxxxxx --senderid xxxxxx
```

Where:

“--senderid” is the ID of the account for the sender you wish to update the signature for.

**Note:**

1. “--help” parameter gives you the usage and explanations of all commands.
2. If you were logged in with your dashboard, logout and log back in, go to “My Account” > “Signature” to see the imported signature.

Alternatively, if you wish to just get the base 64 string for the signature, run the following command:

```
java -jar esl-signature-import-tool.jar --filename mySignature.png --rgbthreshold 140
```

The jar file outputs the result in console and you can run below REST API in some other way:

```
POST /api/account/senders/{senderId}
```

The request body should look like:

```
{  
  
  "signature":{
```

```
"handdrawn":"base_64_signature",  
"textual":null  
}  
}
```

## Integrated Approach

The jar file can be used directly in a java class:

```
import java.io.File;  
  
import java.io.FileInputStream;  
  
import com.silanis.esl.signer.SignatureCaptureValueConverter;  
  
SignatureCaptureValueConverter converter = new SignatureCaptureValueConverter();  
  
try (FileInputStream fis = new FileInputStream(new File(filePath))) {  
  
    String signature = converter.convertToVectorFromImageFormat(fis, 140);  
  
}
```

## Questions or Issues?

Visit the OneSpan Sign Developer Community forums:

<https://developer.esignlive.com/forum>