README

Notary Seal Tools Smart Service Plug-in

A Notary Seal used for digitally signing a document involves a digital representation of an official notary seal or stamp, applied electronically to authenticate the document's authenticity and validate its content. This digital seal serves a similar purpose to a physical notary seal by confirming the integrity and origin of the document in a digital format, ensuring its legal validity and providing assurance of its authenticity.

**Input Parameters**

| **Destination folder id** | Id of the destination folder where the output signed file is saved. |
| --- | --- |
| **Document to sign** | Id of the document to sign.  Uploaded as a document into the Appian environment. |
| **Keystore alias** | Alias of Keystore certificate. |
| **Keystore password** | Password of Keystore certificate. |
| **Keystore type** | Type of Keystore(eg: PKCS12). |
| **List of signature field names** | List of field names where the signature is stamped (Eg. 15 Judge's Signature). |
| **Signature Logo** | Logo image that appears on the signature field. |
| **Output file name** | Filename of the signed output document without extension. |
| **Location** | Location that appears on the signature field. |
| **Signer Name** | Name that appears on the signature field. |
| **Reason** | Reason that appears on the signature field. |
| **Character length to wrap** | Character length to wrap |

**Output Parameters**

| **Signed document Id** | Id of the signed document(output file). |
| --- | --- |
| **Success flag** | Shows if the smart service is processed successfully. |
| **ErrorMessage** | Error Message in case of error |

**SSL certificate generation**

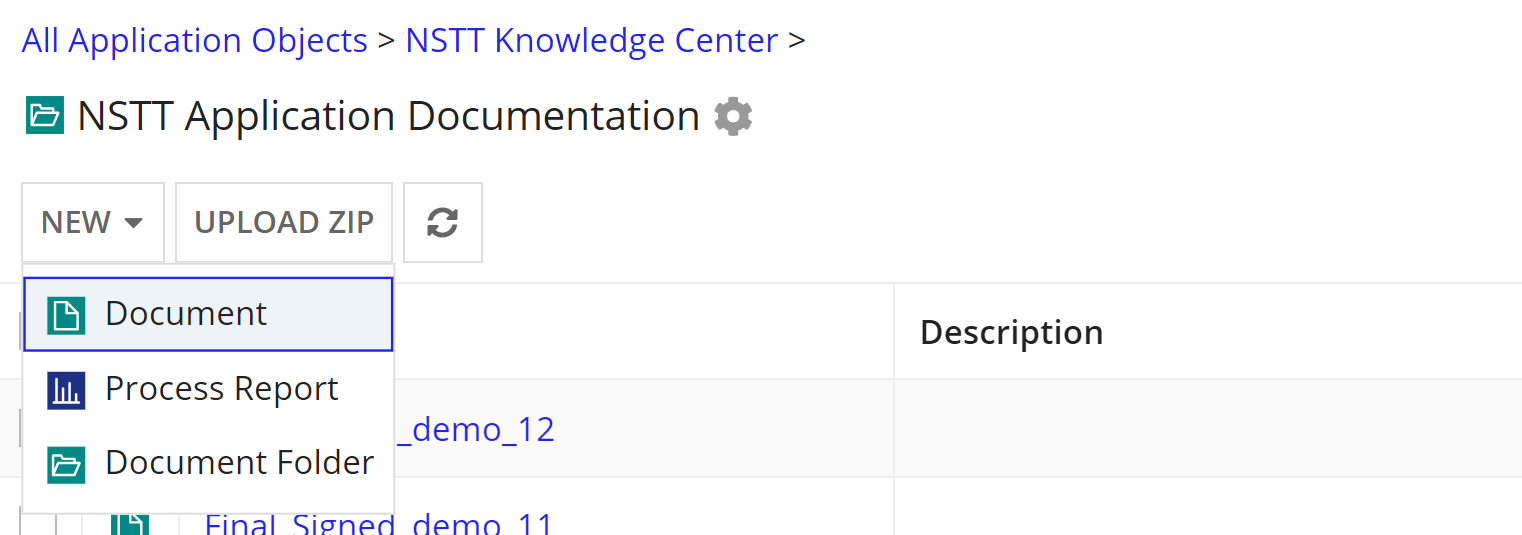
1. Follow the client specific process to get the CA Signed Certificate which includes .pem and .key file.
2. Generate P12 certificate using OpenSSL

C:\OpenSSL-Win64\bin\openssl.exe pkcs12 -export -inkey [keyfilename].key -in

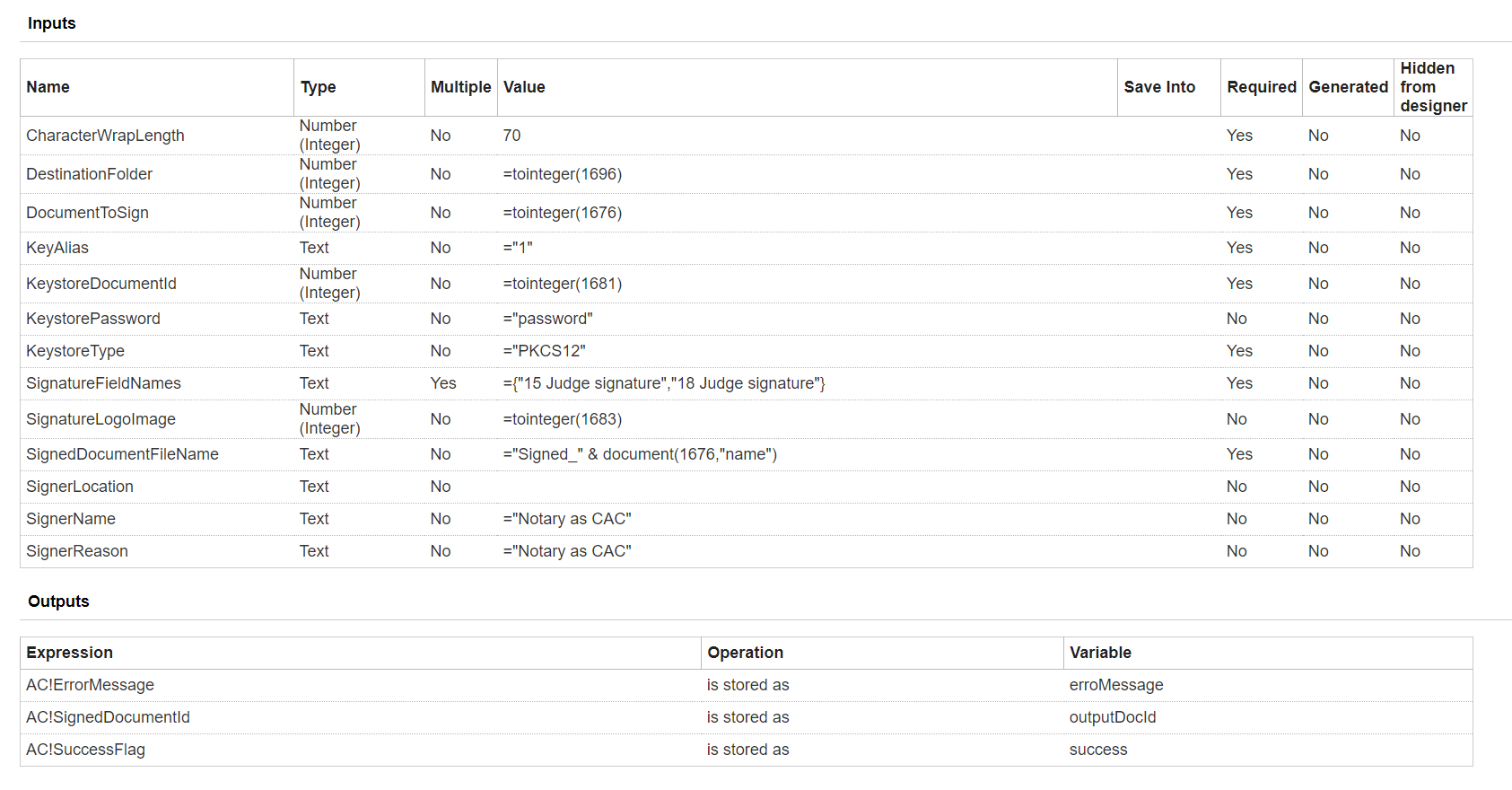
[pemfilename].pem -out [p12outputfilename].p12

Note: You can use any other supported tool to generate a P12 certificate. Plugin uses P12 certificate only.

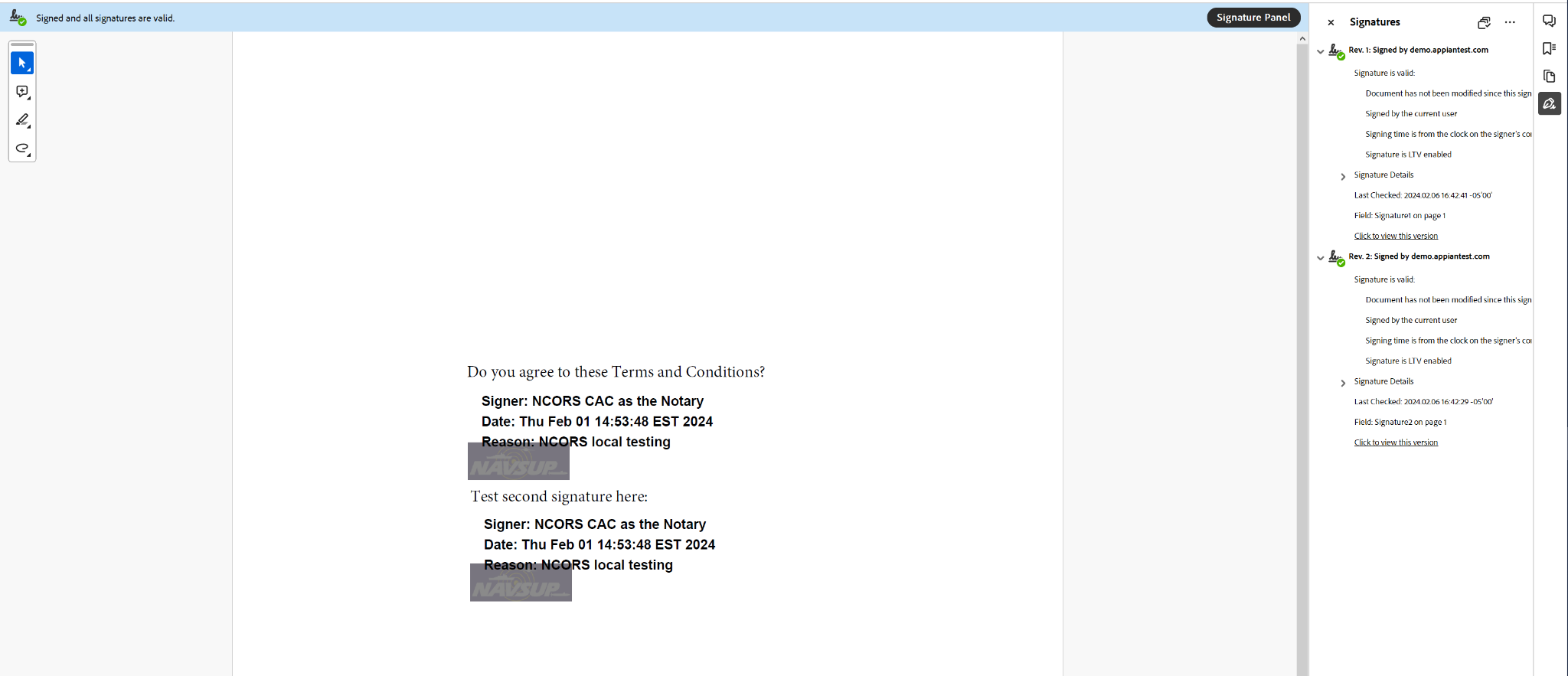
1. Import the P12 certificate into the Appian environment as a document in the application Knowledge Center.



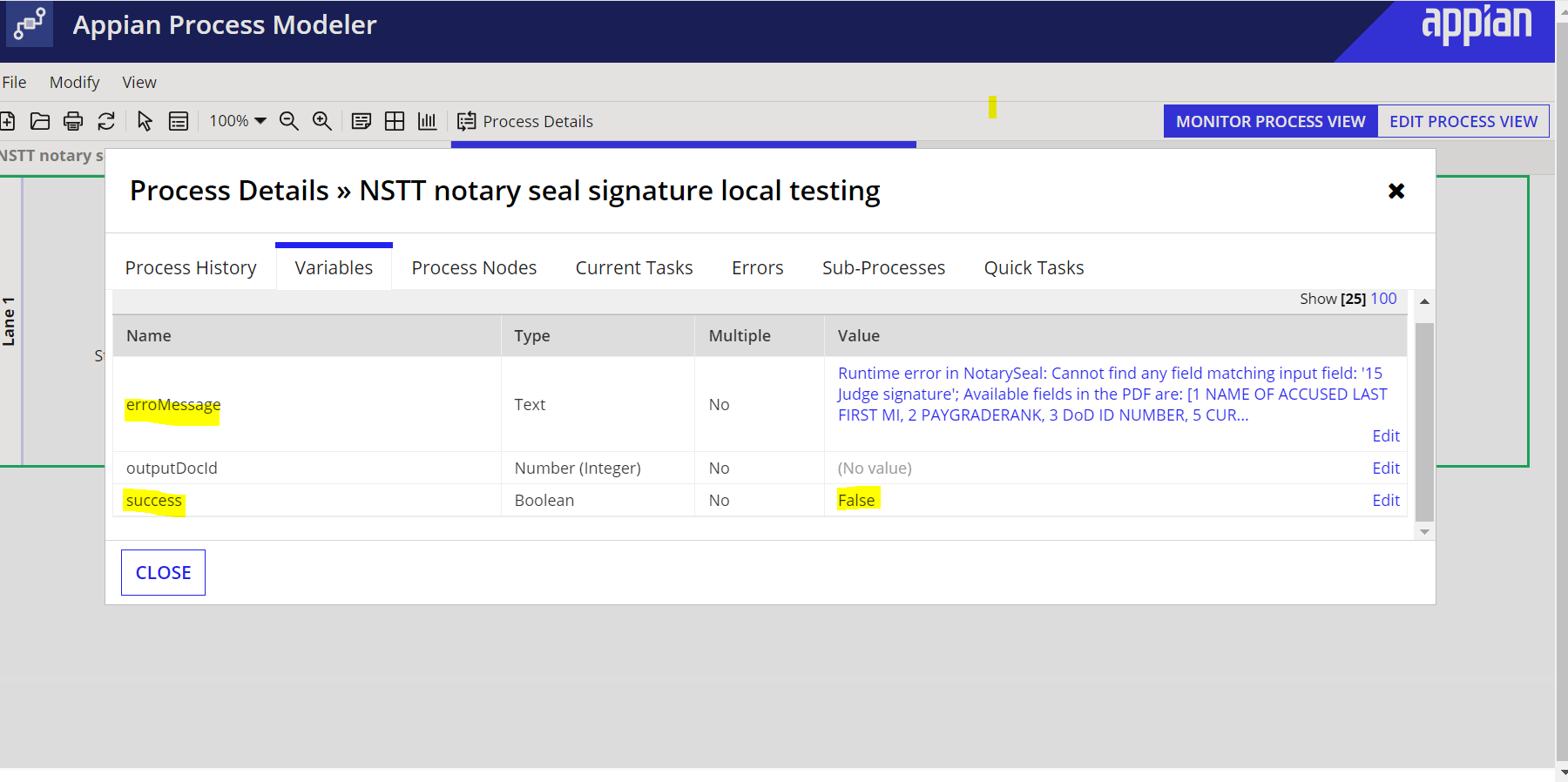
**Notary seal tools smart service configuration example**

****

**Sample output document**



**Sample Error Scenario**

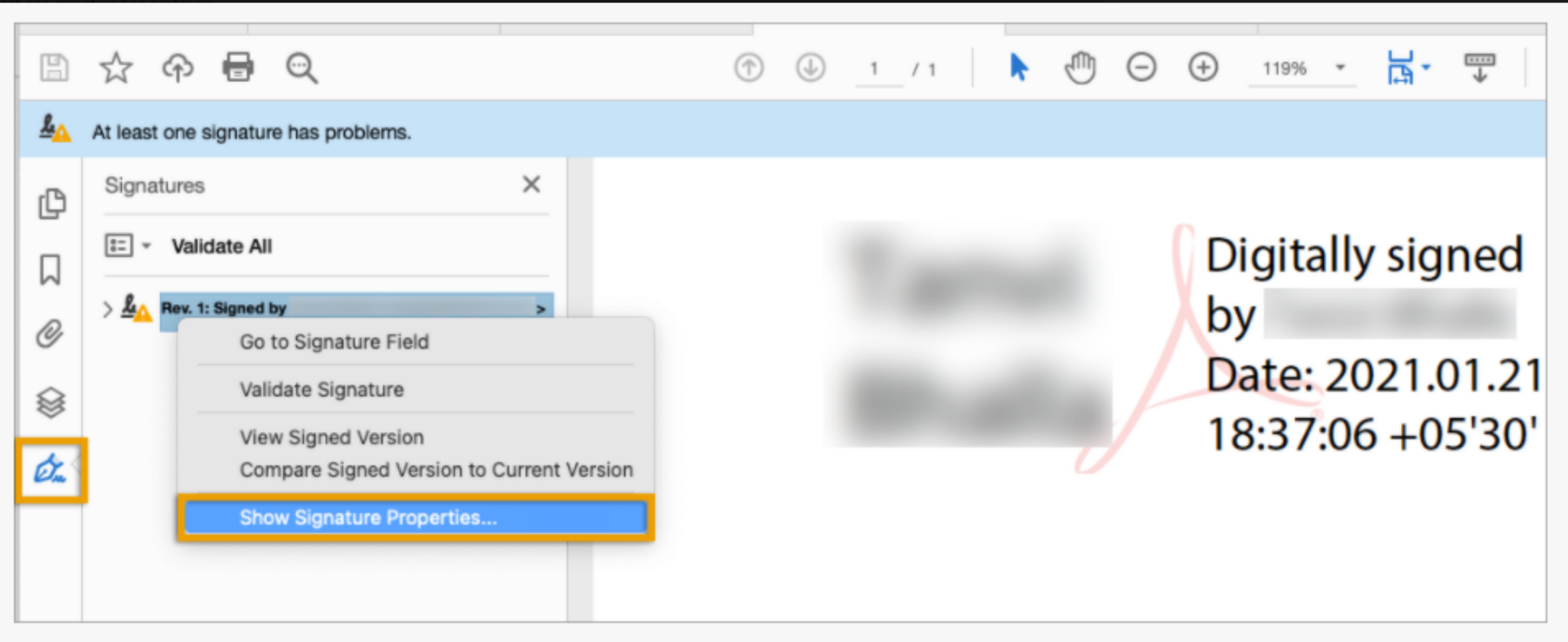
****

**Enable LTV in Adobe**

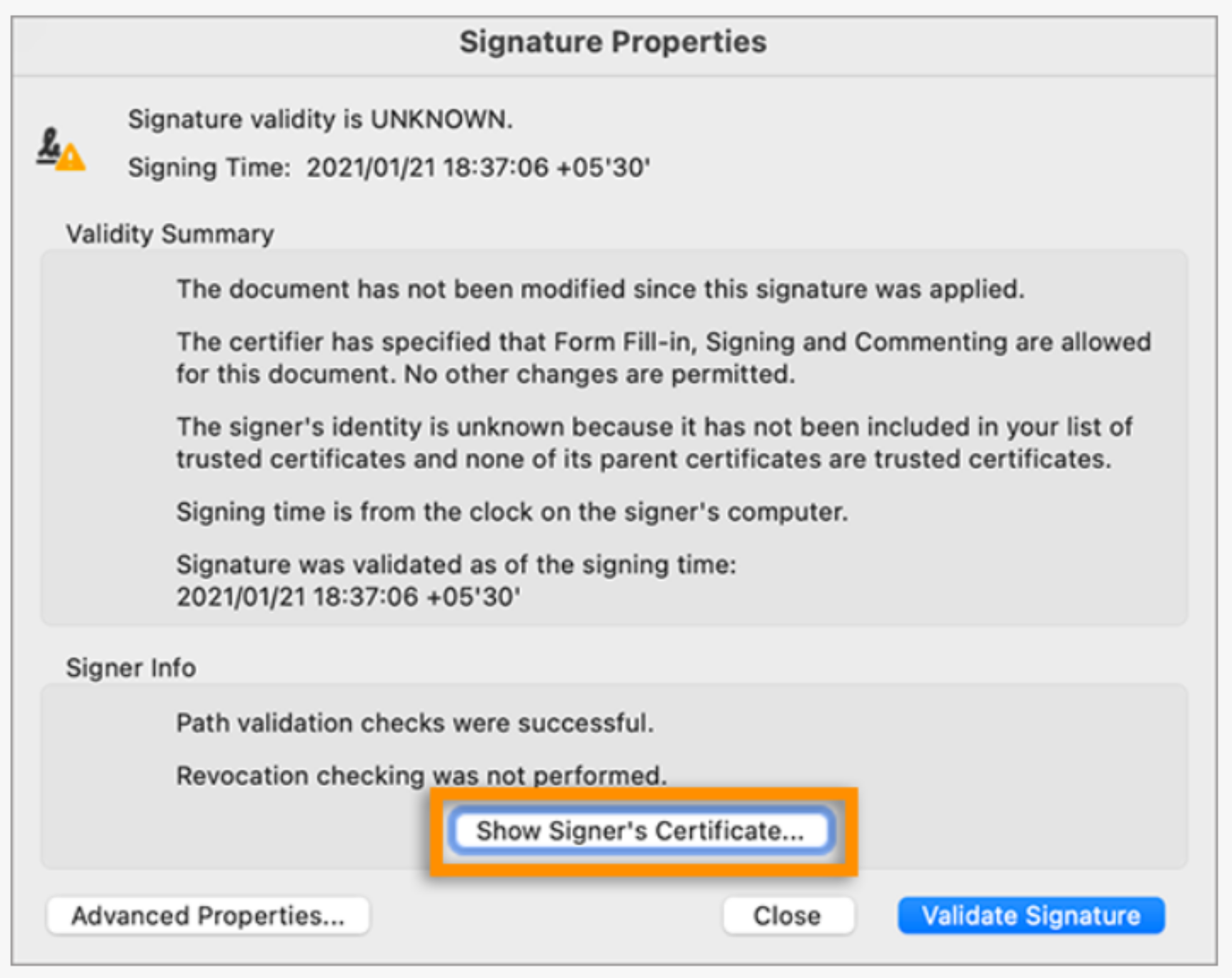
Add the certificate that was used to apply the digital signature into Adobe’s list of Trusted Identities, do the following:

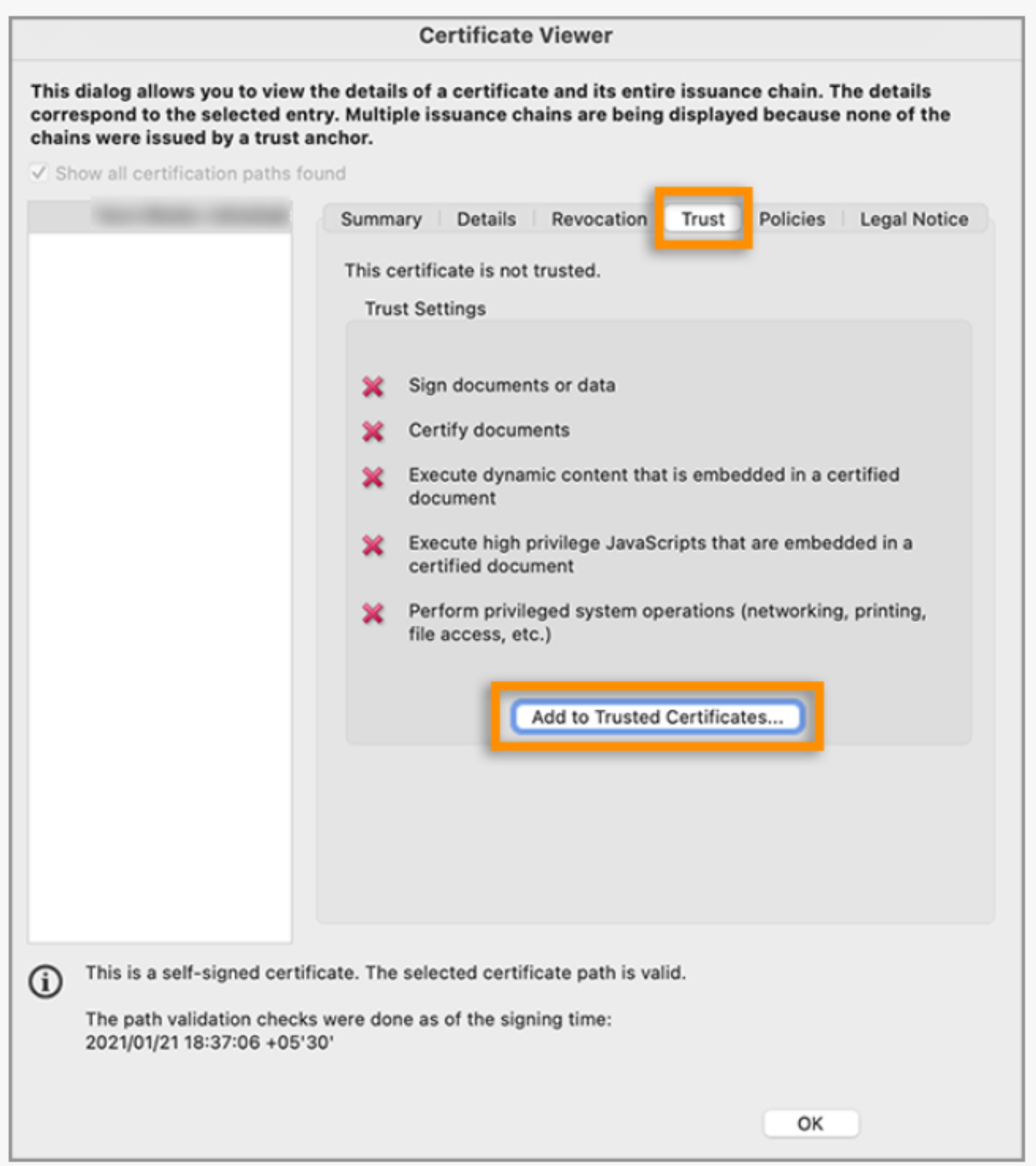
1. Click the Signatures button in the left-pane.

2. Right-click the signature, and then click **Show Signature Properties**.



3. In the Signature Properties dialog box, click **Show Signer's Certificate**.



4. In the Certificate Viewer dialog box, click the **Trust** tab, and then click **Add To Trusted Certificates.**

5. Click **OK** in the trust settings pop-up dialog, and then click **OK.**

**Timestamp server Configuration**

Follow the instructions given in [this](https://helpx.adobe.com/sign/config/time-stamp-settings/overview.html) document to configure a custom time stamp server.