SecureSign PDF

Information Security

BCS-7E

Group Members

Muhammad Usaid (20L-2062)

Muhammad Ibtasam (20L-0901)

Ahmed Farooq (20L-1232)

Instructions:

Following instructions are to be followed in order to run the code.

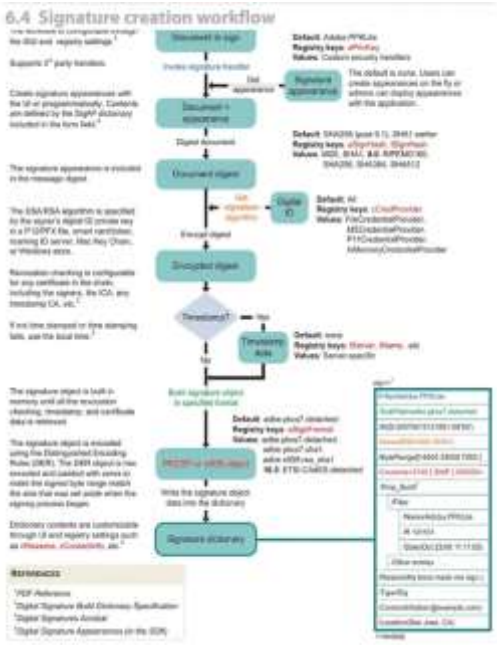
- Open **src/Index.js.**
- Update the two arguments in **SignPDF** class in **index.js** (if you don't want to use the files that are in the **test_assets** folder)**.**
- Create the folder **exports** in the **pdf_sign** project at the hierarchy same as **src** folder.
- Run the command **npm i** in the terminal.
- Run the command **npx babel ./src -d ./dist.**
- Run the command **node dist/index.js.**

Development Methodology:

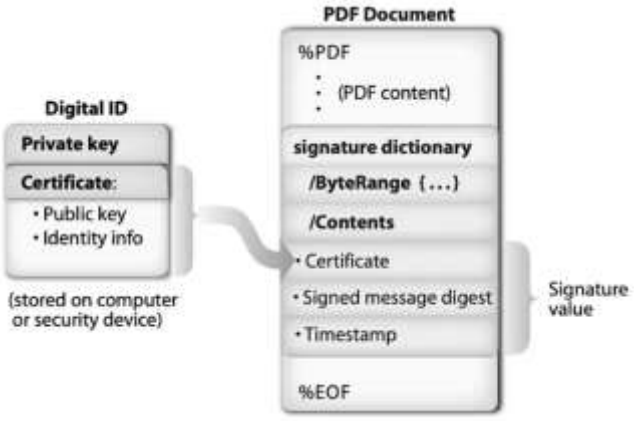
1. Adding a PDF Signature Placeholder

- Use node-signpdf along with pdf-lib.js to identify the area in the PDF where the signature will be placed.
- Define information to be stored in the placeholder:
    i. Signature length (usually a fixed-size byte, e.g., 3322)
    ii. ByteRange placeholder
    iii. Current Date
- Place this information in the PDF as bytecode, leaving space for the signature.

2. Creating the Signature



- Generate the actual signature using node-signpdf.
- Identify the Byterange in the PDF, which consists of four numbers indicating:
    i. Start of the document
    ii. Start of the signature
    iii. End of the signature
    iv. End of the PDF
- A digital signature comprises:
    i. Certificate (cryptographic file verifying identity, provided by a CA Authority)
    ii. Document Digest (hashed state of the PDF before signing)
    iii. Timestamp



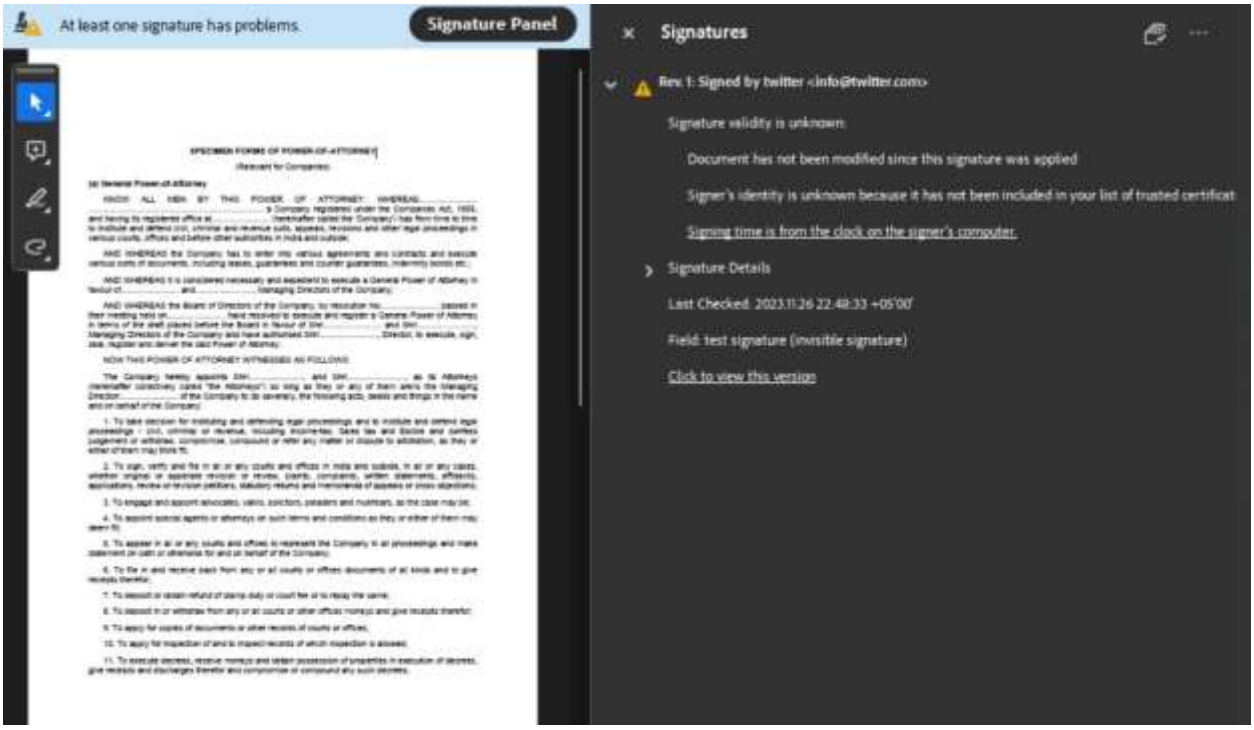
- For self-signing, use OpenSSL to generate the certificate and private key:
    i. Command: openssl req -x509 -newkey rsa:2048 -nodes -keyout mykey.pem -out cert.pem -days 365
- Merge the generated certificate and private key into a single file:
    i. Command: openssl pkcs12 -export -out keystore.p12 -inkey mykey.pem -in cert.pem

3. Embedding the Signature

- Sign the PDF using keystore.p12, replacing the digital signature with the placeholder's Byterange values.

Results:

Once the document is signed using the digital certificate we can verify it by opening it in the acrobat reader as shown in the image below:



The above image shows that the document is signed. Since the digital certificate is issued by a global authority, but here we are doing self-signing where we are the authority and we are doing the signing using the library openSSL.