CSE350: Programming Assignment 4

Ananya Lohani (2019018), Mihir Chaturvedi (2019061)

Digital Signatures

Digital signatures are electronic signatures that **prove the authenticity** of digital documents like PDFs. To sign a PDF with a digital signature, **a unique code is generated based on the document's content**. This code is encrypted using the signer's private key, creating the digital signature, which is embedded within the PDF.

When the recipient receives the signed PDF, they verify the signature **against already present, trusted certificates**. If it does, then the signature is valid, and the document is considered authentic and unchanged since it was signed.

PKCS#7 Standard

The PKCS#7 standard is a technical **specification** for digitally signing and encrypting data. It provides a standard syntax for digital signatures and encryption messages using public-key cryptography, **allowing multiple signers to sign the same data object**. PKCS#7 is widely used in security protocols such as S/MIME, TLS, and **secure PDF documents**.

Steps to digitally sign a PDF

1. Generate a signature placeholder of the required signature length and properties in the PDF.
2. Check if the pdf already contains an acroform field, and extract the acroform ID and field IDs if it exists.
3. Generate a signature annotation widget with the signature placeholder, and add it to the page.
4. Create or extend the existing acroform with the new widget, and link it to the root.
5. Create a signed PKCS#7 message (using libraries like node-forge) and replace the /ByteRange and /Contents sections (the placeholder) in the PDF with the signed message.
6. Save the PDF by writing it to the file system.

Important Questions

1. **How and where do you get the correct GMT date and time? Is the source reliable and the GMT date and time obtained in a secure manner?**
   `new Date()` in Node.js is a reliable way to get the current date and time in the local timezone of the server where the Node.js application is running. This method is secure and trustworthy because it relies on the system clock, which is generally considered to be a reliable source of time. Additionally, the `new Date()` method is built into the JavaScript language, which means it is a well-established and widely-used method for obtaining the current date and time. Since the `new Date()` method is

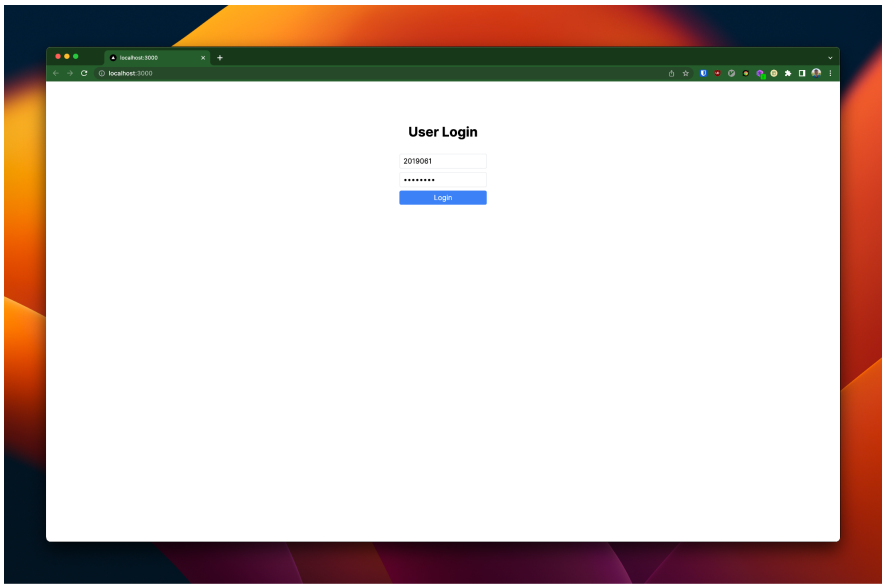
executed locally on the server, it is not affected by network latency or server response times, which can be potential vulnerabilities for web applications. The use of the `new Date()` method also ensures that the timestamp is generated with a high degree of accuracy and precision.

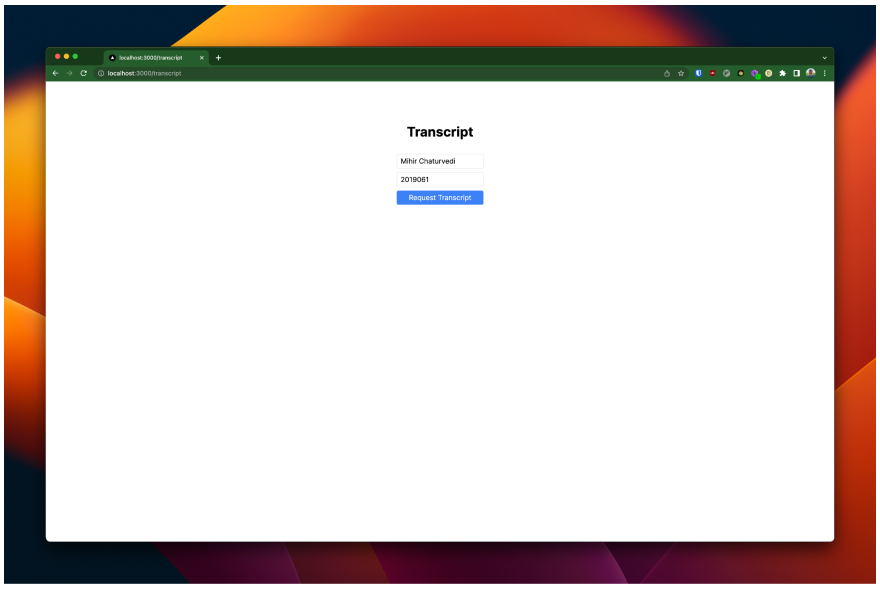
2. **How do you ensure that only the graduate is able to download it (by providing information beyond the roll no, such as date of birth, home pin code, etc.?**
   To ensure that only the graduate is able to download the degree certificate and grade card, the web server implements a secure authentication mechanism using JWT. The student's roll number is used as the username, and the home pin code is used as the password. When a student requests their degree certificate and grade card, they are prompted to enter their roll number and home pin code. The web server then authenticates the student's credentials by verifying that the entered roll number and home pin code match those stored in the database.

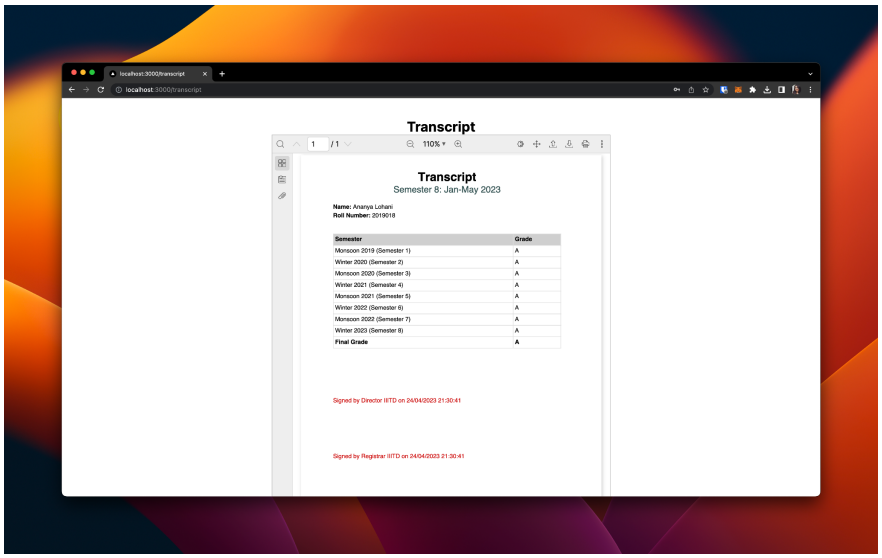
3. **Should the graduate decide to share the document with others, how can one trace the origin of the document (could watermarks be useful?)?**
   The server adds watermarks on the documents when they are signed. Digital watermarks are unique, identifiable markers that are embedded into the document and cannot be removed or altered without detection. These watermarks contain information about the signer and the correct date and time of signing. By embedding this information into the document, the origin and authenticity of the document can be easily traced.

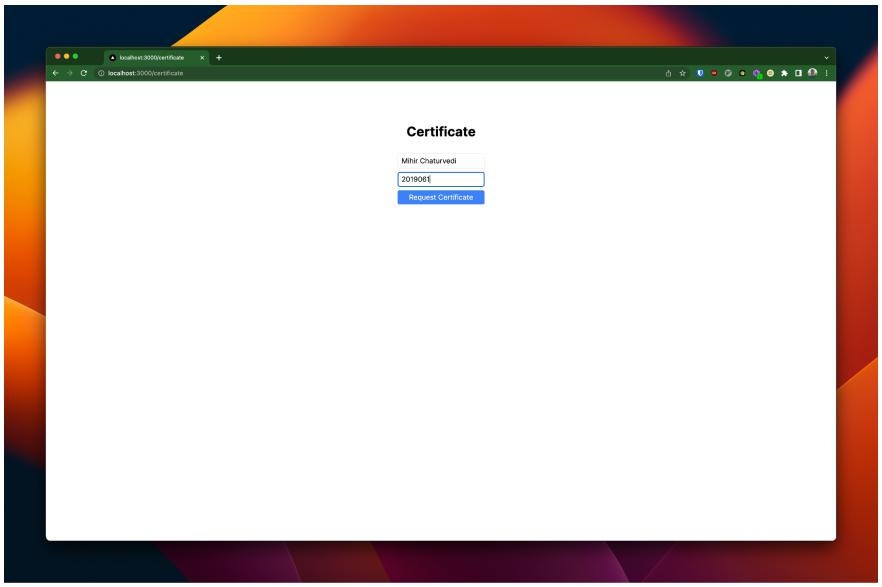
4. **Do we need to have access to public-keys, and if so how?**
   We need to have access to the signer’s PKCS#12 file, which is a file with extension .p12 containing the public key, private key and a digital certificate for integrity. The .p12 file is passphrase-protected, and the signer needs to provide the passphrase in order to sign a document. In our project, we have assumed that the server already has access to the .p12 file as well as the passphrase, which is stored in the .env file of the server. Once the server has access to the .p12 file, it can use the private key to sign the degree certificate and grade card, along with the current date and time.

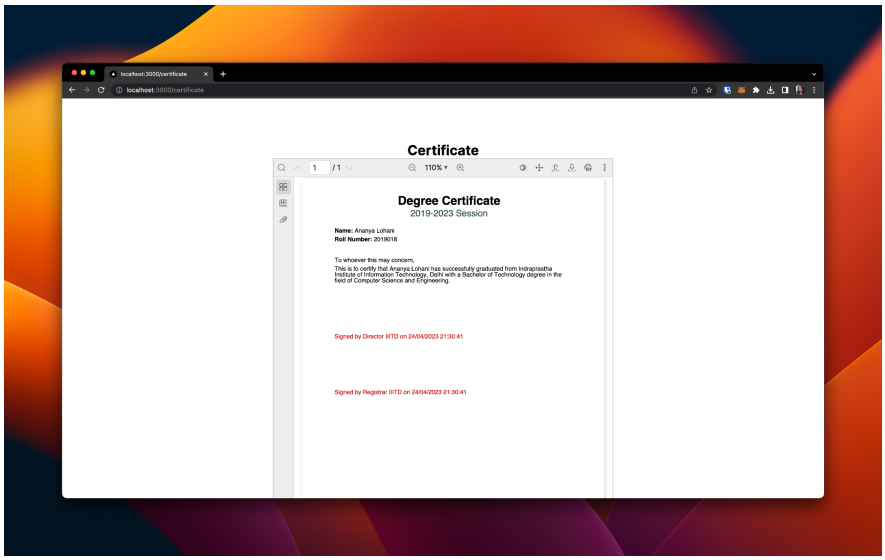
Screenshots



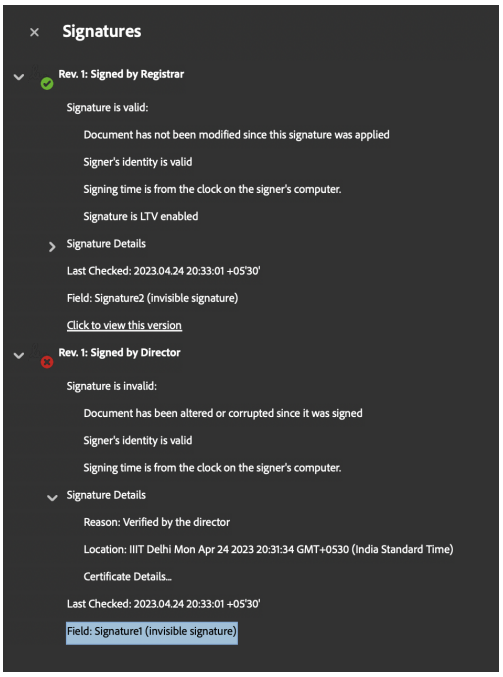








Verified signatures on Adobe Acrobat



When a PDF document is signed, it is possible to make changes to the document without invalidating the original signature. This is because PDFs support incremental updates, which means that when changes are made to the document, they are appended to the end of the file without overwriting the previous content.

As a result, each revision of the PDF is valid and can be verified independently of the other revisions. If a PDF document has multiple signatures, each signature applies only to the revision of the document that existed when it was signed. Any changes made to the document after that point will not be reflected in the signature.

Signatures in Raw PDF

```
23 0 obj
<<
/Type /Sig
/Filter /Adobe.PPKLite
/SubFilter /adbe.pkcs7.detached
/ByteRange [0 13676 21870 9905]
/Contents <3082089e06092a864886f70d010702a082088f3082088b020101310f300d0609608648
/Reason (Verified by the director)
/M (D:202304241501362)
/ContactInfo (director@iiitd.ac.in)
/Name (Director)
/Location (IIIT Delhi Mon Apr 24 2023 20:31:34 GMT+0530 \(India Standard Time\))
>>
endobj
```

```
26 0 obj
<<
/Type /Sig
/Filter /Adobe.PPKLite
/SubFilter /adbe.pkcs7.detached
/ByteRange [0 22832 31026 749]
/Contents <3082089e06092a864886f70d010702a082088f3082088b020101310f300d0609608648
/Reason (Verified by the registrar)
/M (D:202304241501362)
/ContactInfo (registrar@iiitd.ac.in)
/Name (Registrar)
/Location (IIIT Delhi Mon Apr 24 2023 20:31:34 GMT+0530 \(India Standard Time\))
>>
endobj
```