TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG THÀNH PHỐ HỒ CHÍ MINH
KHOA: HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM

BÁO CÁO MÔN HỌC
AN TOÀN VÀ BẢO MẬT HTTT

CHỮ KÍ SỐ TRÊN PDF

Giảng viên hướng dẫn : **ThS. Phạm Trọng Huynh**

CN. Nguyễn Phan Chí Thành

Lớp **: 09_ĐH_CNPM1**

Nhóm thực hiện: **Nhóm 7**

STT	Họ và Tên	MSSV
1	**Nguyễn Hoàng Trọng Nghĩa**	**0950080004**
2	**Huỳnh Phan Thu Đông**	**0950080012**
3	**Phạm Thế Vinh**	**0950080020**

TP. Hồ Chí Minh, tháng 5 năm 2024

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG
THÀNH PHỐ HỒ CHÍ MINH
KHOA: HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM

BÁO CÁO MÔN HỌC
AN TOÀN VÀ BẢO MẬT HTTT

CHỮ KÍ SỐ TRÊN PDF

Giảng viên hướng dẫn : **ThS. Phạm Trọng Huynh**

CN. Nguyễn Phan Chí Thành

Lớp **: 09_ĐH_CNPM1**

Nhóm thực hiện: **Nhóm 7**

STT	Họ và Tên	MSSV
1	**Nguyễn Trọng Hoàng Nghĩa**	**0950080004**
2	**Huỳnh Phan Thu Đông**	**0950080012**
3	**Phạm Thế Vinh**	**0950080020**

TP. Hồ Chí Minh, tháng 5 năm 2024

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến thầy Phạm Trọng Huynh và thầy Nguyễn Phan Chí Thành. Trong suốt thời gian học môn an toàn và bảo mật hệ thống thông tin, em đã được thầy dạy dỗ và hướng dẫn rất tận tâm. Thầy đã dành nhiều thời gian và tâm huyết để hướng dẫn em trong việc nghiên cứu về đề tài chữ kí số trên pdf.

Em đã tận dụng những kiến thức mà thầy đã truyền đạt để thực hiện việc tìm hiểu chữ kí số trên pdf. Mặc dù em đã cố gắng hết sức của mình, nhưng cũng không thể tránh khỏi những hạn chế và thiếu sót. Vì vậy, em rất mong nhận được sự đánh giá và góp ý từ thầy để có cơ hội hoàn thiện ứng dụng này hơn nữa.

Em chân thành cảm ơn sự hỗ trợ và khuyến khích không ngừng nghỉ của thầy. Kính chúc thầy luôn có nhiều sức khỏe, hạnh phúc và thành công trong công việc và giảng dạy, đào tạo ra những kĩ sư lập trình chất lượng và tâm huyết để góp phần phát triển đất nước.

Một lần nữa chúng em xin chân thành cảm ơn!

NHẬN XÉT

……
……
……
……
……
……
……
……
……
……
……
……
……
……

MỤC LỤC

Chương 1: Tóm tắt

Chữ ký điện tử (tiếng Anh: electronic signature hay e-signature) là thông tin đi kèm theo dữ liệu (văn bản, hình ảnh, video...) nhằm mục đích xác định người chủ của dữ liệu đó. Chữ ký điện tử có giá trị pháp lý tương đương như chữ ký viết tay, miễn là nó tuân theo các yêu cầu của quy chế cụ thể mà nó được tạo ra (ví dụ, eIDAS ở Liên minh châu Âu, NIST-DSS ở Hoa Kỳ hoặc ZertES ở Thụy Sĩ).

Chữ ký điện tử là một khái niệm pháp lý khác biệt với chữ ký số (digital signature), một cơ chế mã hóa thường được sử dụng để triển khai chữ ký điện tử. Trong khi chữ ký điện tử có thể đơn giản chỉ là tên được nhập vào tài liệu điện tử, chữ ký số ngày càng được sử dụng trong giao dịch điện tử và trong các tài liệu đăng ký để triển khai chữ ký điện tử một cách an toàn về mặt mã hóa. Các cơ quan tiêu chuẩn như NIST hay ETSI cung cấp các tiêu chuẩn để triển khai chúng (ví dụ: NIST-DSS, XAdES hoặc PAdES). Khái niệm này không phải là mới, với các quốc gia áp dụng thông luật đã công nhận chữ ký điện báo từ giữa thế kỷ 19 và chữ ký fax từ những năm 1980. Nhóm sử dụng thư viện Ironpdf để thực hiện việc ký chữ kí điện tử vào file pdf bằng file .pfx.

Chương 2: Giới thiệu

2.1. Lịch sử hình thành

Con người đã sử dụng các hợp đồng dưới dạng điện tử từ hơn 100 năm nay với việc sử dụng mã Morse và điện tín. Vào năm 1889, tòa án tối cao bang New Hampshire (Hoa Kỳ) đã phê chuẩn tính hiệu lực của chữ ký điện tử. Tuy nhiên, chỉ với những phát triển của khoa học kỹ thuật gần đây thì chữ ký điện tử mới đi vào cuộc sống một cách rộng rãi.

Vào thập niên 1980, các công ty và một số cá nhân bắt đầu sử dụng máy fax để truyền đi các tài liệu quan trọng. Mặc dù chữ ký trên các tài liệu này vẫn thể hiện trên giấy nhưng quá trình truyền và nhận chúng hoàn toàn dựa trên tín hiệu điện tử.

Hiện nay, chữ ký điện tử có thể bao hàm các cam kết gửi bằng email, nhập các số định dạng cá nhân (PIN) vào các máy ATM, ký bằng bút điện tử với thiết bị màn hình cảm ứng tại các quầy tính tiền, chấp nhận các điều khoản người dùng (EULA) khi cài đặt phần mềm máy tính, ký các hợp đồng điện tử online…\

2.2. Các loại chữ ký điện tử

Các loại chữ ký điện tử được sử dụng phổ biến hiện nay có thể kế đến như:

Chữ ký số: Chữ ký số là một dạng chữ ký điện tử. Hiểu về căn bản, chữ ký số cũng giống như chữ viết tay vậy được sử dụng để cam kết và điều đó không thể rút lại được. Tuy nhiên, chữ ký số không sẽ không phải sử dụng giấy mực, nó gắn đặc điểm nhận dạng của người ký vào bản cam kết. Chữ ký số đóng vai trò như chữ ký đối với cá nhân hay con dấu đối với doanh nghiệp và được thừa nhận về mặt pháp lý.

Chữ ký scan: Là loại chữ ký được chuyển thành dạng điện tử sau khi thực hiện ký tay trên giấy thông qua máy quét (scan) rồi gửi qua thư điện tử.

Chữ ký hình ảnh: Là loại chữ ký được ký bằng tay sau đó chuyển thành dạng ảnh và chèn vào hợp đồng điện tử.

2.3. Phân biệt chữ ký điện tử và chữ ký số

Giống nhau		Đều được sử dụng để thay thế chữ ký tay và con dấu của cá nhân, tổ chức, doanh nghiệp khi thực hiện các giao dịch trên môi trường điện tử	
		Chữ ký điện tử	**Chữ ký số**
Khác nhau	**Mục đích**	Xác nhận chủ thể ký và khẳng định sự chấp thuận của chủ thể đó đối với thông điệp dữ liệu	Bảo đảm tính xác thực, tính toàn vẹn và tính chống chối bỏ
	Bảo mật	Không sử dụng mã hóa	Sử dụng thuật toán khóa không đối xứng, gồm khóa bí mật và khóa công khai
	Xác nhận	Không có quá trình xác nhận.	Cần được xác nhận bởi nhà cung cấp dịch vụ chữ ký số hoặc cơ quan có thẩm quyền.
	Khả năng giả mạo chữ ký	Dễ thực hiện	Không thể sao chép, giả mạo
	Cách tạo chữ ký	Dễ dàng tạo bằng cách scan hình ảnh, tạo trên các website trực tuyến	Phải đăng ký sử dụng với các đơn vị cung cấp dịch vụ chứng thực chữ ký số

		hoặc tạo trực tiếp trên Word, Excel	
	Cách sử dụng chữ ký	Thực hiện chèn chữ k vào vị trí cần ký điện tử trên văn bản, tài liệu.	Người dùng cần kết nối với chữ ký số, nhập mã PIN bảomật và tiến hành ký vào vị trí cần ký số trên văn bản, tài liệu.
	Phần mềm độc quyền	Có thể được xác nhận bởi bất cứ ai mà không cần phần mềm xác minh độc quyền	Trong nhiều trường hợp, chữ ký điện tử không được ràng buộc về mặt pháp lý và sẽ yêu cầu phần mềm độc quyền để xác nhận chữ ký điện tử.

Chương 3: Các bước trong thuật toán

3.1 Tạo khóa RSA và Mã hóa/Giải mã

B1. Tạo khóa RSA:

Tạo một cặp khóa RSA (công khai và bí mật).

Xuất khóa công khai và khóa bí mật ra các tệp XML (`pubKey.xml` và `priKey.xml`).

```
RSA rsa = RSA.Create();string pubKey = rsa.ToXmlString(false);string priKey = rsa.ToXmlString(true);

File.WriteAllText("pubKey.xml", pubKey);

File.WriteAllText("priKey.xml", priKey);
```

B2. Mã hóa thông điệp:

Đọc khóa công khai từ tệp.

Mã hóa thông điệp sử dụng khóa công khai với padding PKCS1.

```
RSA rsa = RSA.Create();string pubKey = File.ReadAllText("pubKey.xml");

rsa.FromXmlString(pubKey);byte[] b = Encoding.UTF8.GetBytes(m);return rsa.Encrypt(b, RSAEncryptionPadding.Pkcs1);
```

B3. Giải mã thông điệp:

Đọc khóa bí mật từ tệp.

Giải mã thông điệp đã mã hóa trở lại thành dạng ban đầu.

```
RSA rsa = RSA.Create();string priKey = File.ReadAllText("priKey.xml");

rsa.FromXmlString(priKey);byte[] rs = rsa.Decrypt(cipher,
RSAEncryptionPadding.Pkcs1);return Encoding.UTF8.GetString(rs);
```

3.2 Ký số và xác thực chữ ký XML

B1. Ký số tài liệu XML:

Tải tài liệu XML từ tệp.

Sử dụng khóa bí mật để ký số tài liệu XML.

Thêm chữ ký vào tài liệu XML và lưu lại.

```
XmlDocument doc = new XmlDocument();

doc.Load("data.xml");

RSA rsa = RSA.Create();string priKey = File.ReadAllText("priKey.xml");

rsa.FromXmlString(priKey);

SignedXml signed = new SignedXml(doc);

signed.SigningKey = rsa;

Reference refe = new Reference();

refe.Uri = "";

XmlDsigEnvelopedSignatureTransform transform = new
XmlDsigEnvelopedSignatureTransform();
```

```
refe.AddTransform(transform);

signed.AddReference(refe);

signed.ComputeSignature();

XmlElement element = signed.GetXml();

doc.DocumentElement.AppendChild(element);

doc.Save("DataSigned.xml");
```

B2. Xác thực chữ ký XML:

Tải tài liệu XML đã ký.

Sử dụng khóa công khai để xác thực chữ ký.

```
RSA rsa = RSA.Create();string pubKey = File.ReadAllText("pubKey.xml");

rsa.FromXmlString(pubKey);

XmlDocument doc2 = new XmlDocument();

doc2.Load("DataSigned.xml");

SignedXml signed = new SignedXml(doc2);

XmlNode node = doc2.GetElementsByTagName("Signature")[0];

signed.LoadXml((XmlElement)node);

System.Console.WriteLine(signed.CheckSignature(rsa));
```

3.3 Tạo và xuất chứng chỉ tự ký

B1. Tạo chứng chỉ tự ký:

Sử dụng khóa bí mật để tạo yêu cầu chứng chỉ (Certificate Request).

Tạo chứng chỉ tự ký và xuất ra tệp PFX.

```
RSA rsa = RSA.Create();

rsa.FromXmlString(File.ReadAllText("priKey.xml"));

CertificateRequest request = new
CertificateRequest("CN=TNMT;O=HeThongThongTin;Email=abc.com,", rsa,
HashAlgorithmName.SHA256, RSASignaturePadding.Pkcs1);

X509Certificate cert = request.CreateSelfSigned(DateTimeOffset.Now,
DateTimeOffset.Now.AddYears(1));byte[] arr =
cert.Export(X509ContentType.Pfx, "123456");

File.WriteAllBytes("cert.pfx", arr);
```

B2. Đọc chứng chỉ từ tệp PFX:

Tải chứng chỉ từ tệp PFX và lưu trữ dưới dạng đối tượng `X509Certificate2`.

```
X509Certificate2 cert = new X509Certificate2("cert.pfx", "123456",
X509KeyStorageFlags.Exportable | X509KeyStorageFlags.MachineKeySet |
X509KeyStorageFlags.PersistKeySet);
```

3.4 Ký số tài liệu PDF và chèn hình ảnh

B1. Ký số tài liệu PDF:

Thiết lập khóa cấp phép cho IronPDF.

Tạo nội dung HTML từ thông tin chi tiết của chứng chỉ SSL.

Ký số tài liệu PDF bằng chứng chỉ từ tệp PFX và lưu tài liệu đã ký.

```
License.LicenseKey = "IRONSUITE...";string htmlContent = $@"
    <h1>SSL Certificate Details</h1>
    <p><strong>Subject:</strong> {cert.Subject}</p>
    <p><strong>Issuer:</strong> {cert.Issuer}</p>
    <p><strong>Valid From:</strong> {cert.NotBefore}</p>
    <p><strong>Valid Until:</strong> {cert.NotAfter}</p>
";var renderer = new IronPdf.ChromePdfRenderer();
PdfDocument pdf = renderer.RenderHtmlAsPdf("<h1>anh em ta nè kkkk chữ kí điện tử trên pdf 100%</h1>" + htmlContent);var sig = new PdfSignature(cert);
pdf.Sign(sig);
pdf.SaveAs("signed10.pdf");
```

B2. Chèn hình ảnh vào PDF:

Mở tài liệu PDF đã ký và chèn hình ảnh vào tài liệu.

```
var pdf2 = PdfDocument.FromFile("signed10.pdf");
```

```
pdf2.ApplyWatermark("<img src='vnd1.png'/>", 90, VerticalAlignment.Bottom,
HorizontalAlignment.Right);

pdf2.SaveAs("11.pdf");
```

B3. Xác thực chữ ký PDF:

Kiểm tra xem tài liệu PDF đã ký có hợp lệ không bằng cách xác thực chữ ký số trong tài liệu.

```
PdfDocument pdf1 = PdfDocument.FromFile("signed8.pdf");bool isValid =
pdf1.VerifyPdfSignatures();

System.Console.WriteLine(isValid);
```

Chương 4: Cài đặt

- **Visual Studio Code**
  - **https://code.visualstudio.com/download**
- **Download .Net 8**
  - **https://dotnet.microsoft.com/en-us/download**
- **Cài đặt gói IronPDF NuGet**
  - PM> Install-Package IronPdf

Chương 5: Kết luận

Thành công trong việc ký trên Pdf, chữ ký được áp dụng chính xác, sau khi đã chứng thực chữ ký thì cho thấy kết quả chữ ký là hợp lệ

Thư viện IronPdf đã cung cấp một cách tiếp cận dễ dàng và linh hoạt cho việc ký trên file Pdf, giúp cho chúng ta có thể thực hiện quy trình này một cách thuận lợi và hiệu quả, giúp tiết kiệm thời gian và công sứ so với việc triển khai giải pháp chữ ký số từ đầu, thư viện cũng thể hiện sự ổn định và đáng tin cậy trong quá trình ký trên file pdf, đảm bảo tính toàn vẹn và bảo mật của tài liệu

TÀI LIỆU THAM KHẢO

[1] Ironpdf. [Online]. Available: https://ironpdf.com/how-to/signing/#sign-a-pdf-with-a-digital-certificate

[2] Advancedinstaller.[Online]. Available: https://www.advancedinstaller.com/what-is-pfx-certificate.html

[3] Microsoft.[Online]. Available: https://learn.microsoft.com/en-us/answers/questions/736832/how-to-get-the-pfx-certificate-password

[4] Groupdocs. [Online]. Available: https://products.groupdocs.com/signature/net/

[5] Devexpress. [Online]. Available: https://www.devexpress.com/products/net/office-file-api/digital-signature-api.xml