

# **BÁO CÁO BÀI TẬP VỀ NHÀ**

## **MÔN: AN TOÀN VÀ BẢO MẬT THÔNG TIN**

### **Chủ đề: Chữ ký số trong file PDF**

**Sinh viên thực hiện:** Nguyễn Thị Linh

**MSSV:**K225480106040

**Lớp:**K58KTP

**Nội dung:** Tập PDF dùng để minh họa quy trình ký số theo 8 bước; báo cáo tóm tắt cấu trúc PDF liên quan chữ ký, nơi lưu thời gian ký, phân tích rủi ro và biện pháp giảm thiểu.

### **1. MỤC TIÊU BÀI TẬP**

- Trình bày, giải thích và minh họa bằng ví dụ cách thức chữ ký số được lưu và bảo vệ trong file PDF.

+ Vị trí và cấu trúc lưu chữ ký trong PDF (AcroForm, Signature Field, Signature Dictionary).

+ Cách lưu thời gian ký (khác nhau giữa /M và timestamp RFC -3161).

+ Các rủi ro bảo mật phổ biến và biện pháp giảm thiểu.

#### **1) Cấu trúc PDF liên quan chữ ký (Nghiên cứu)**

- **Mô tả ngắn gọn:** Catalog, Pages tree, Page object, Resources, Content streams, XObject, AcroForm, Signature field (widget), Signature dictionary (/Sig), /ByteRange, /Contents, incremental updates, và DSS (theo PAdES).

- Liệt kê object refs quan trọng và giải thích vai trò của từng object trong lưu/truy xuất chữ ký.

- **Đầu ra:** 1 trang tóm tắt + sơ đồ object (ví dụ: Catalog → Pages → Page → /Contents ; Catalog → /AcroForm → SigField → SigDict)

➤ Các objects chính trong PDF:

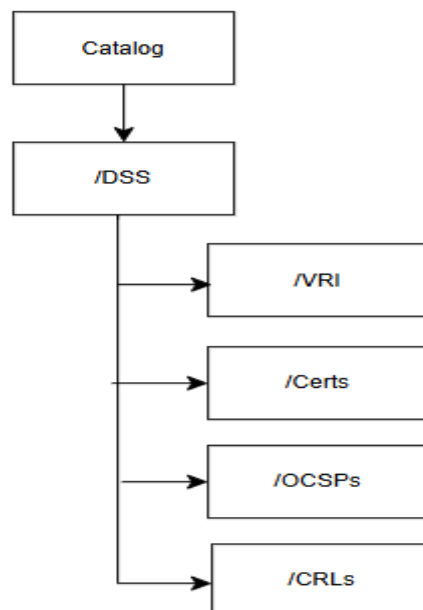
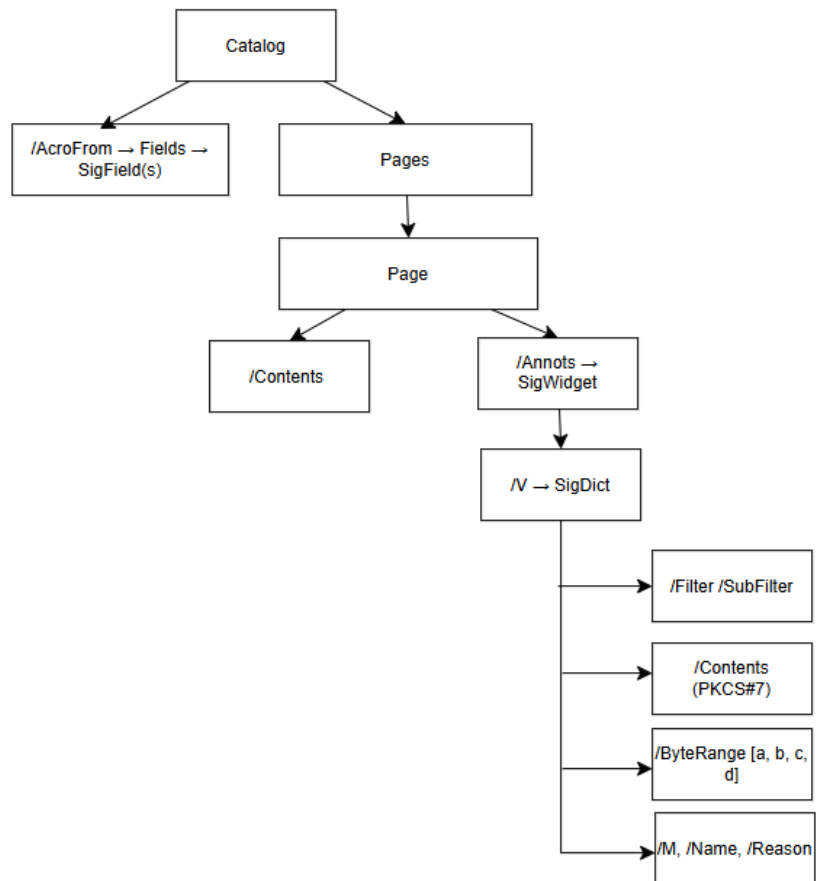
- **Catalog(Root):** đối tượng gốc của PDF, chứa các tham chiếu đến /Pages, có thể chứa thêm /AcroForm ( các from filed) và /DSS (Document Security Store cho PadEs).
- **Pages tree →Page object :** Mỗi trang PDF có /Resources và /Contents ( chứa nội dung hiển thị). Vùng hiển thị chữ ký( nếu có) thường là From XObject nằm trong /Resources và được tham chiếu qua /Annots.
- **AcroForm:** Chứa danh sách các from field. Trường chữ ký ( Signature field) nằm ở đây, là một “widget field” có kiểu /FT /Sig.
- **Signature field (Widget):** Đối tượng đại diện cho trường chữ ký. Khi ký xong, nó sẽ tham chiếu đến Signature dictionary thông qua khóa /V.
- **Signature dictionary (/Sig):** Nơi lưu dữ liệu chữ ký thông tin meta:
  - /Type,/Sig: Xác định kiểu đối tượng.
  - /Filter và /SubFilter: quy định dạng
  - /Contents: Vùng byte chứa chữ ký PKCS#7/CMS hoặc CMS + timestamp
  - /ByteRange: mảng [start1 length1 start2 length2] chỉ định vùng dữ liệu được ký( loại trừ /Contents).
  - M: thời gian ký dạng text ( không có giá trị pháp lý).
  - /Name, /Reason, /Location: Thông tin người ký,lý do,nơi ký.
- **/Contents vs. /ByteRange :** /ByteRange cho biết vùng dữ liệu được băm, /Contents là vùng bị loại trừ để chèn chữ ký sau khi băm.
- **Incremental updates:** PDF cho phép “ ghi thêm phần mới” mà không thay đổi dữ liệu cũ. Khi ký ,PDF sẽ ghi một incremental update; nhờ đó ta có thể phát hiện sửa đổi sau khi ký.
- **XObject (From Xobject):** dùng để hiển thị chữ ký trên trang.

- **DSS ( Document Security Store):** vùng lưu trữ thông tin xác minh dài hạn(LTV) như chứng chỉ, OCSP, CRL, timestamp.

➤ Các object refs quan trọng

- **Catalog (Root):** chứa /AcroForm và liên kết tới pages; entry bắt đầu của traversal.
- **AcroForm:** chứa /Fields (mảng SigField refs) và /SigFlags.
- **SigField (Widget annotation):** vị trí visual trên Page, tham chiếu tới SigDict qua /V sau khi signed.
- **Signature dictionary (SigDict):** chứa /Contents (PKCS#7), /ByteRange, /M, /Filter, /SubFilter (ví dụ /adbe.pkcs7.detached), /Name, /Location.
- **Page:** chứa /Annots array (với widget ref) và /Contents (appearanceXObject có thể reference tới SigField appearance).
- **/Contents (Signature placeholder in SigDict):** vùng nhúng blob DER PKCS#7.
- **DSS / VRI (PAdES):** chứa hỗ trợ xác thực lâu dài: certs, ocspResponses, crls, vri entries referencing signature byte ranges.
- **Incremental update (xref/trailer of appended revision):** chứa new objects (SigDict, updated AcroForm field V pointer) appended — cho phép detection of post-sign changes.

❖ Sơ đồ quan hệ object



## 2) Thời gian ký được lưu ở đâu?

- **Nêu tất cả vị trí có thể lưu thông tin thời gian:**

+ **/M trong Signature dictionary (dạng text, không có giá trị pháp lý).**

+ **Timestamp token (RFC 3161) trong PKCS#7 (attribute timeStampToken).**

+ **Document timestamp object (PAdES).**

+ **DSS (Document Security Store) nếu có lưu timestamp và dữ liệu xác minh.**

- **Giải thích khác biệt giữa thông tin thời gian /M và timestamp RFC**

- **/M trong Signature dictionary (dạng text, không có giá trị pháp lý)**
  - + Chuỗi text kiểu (D: D:YYYYMMDDHHmmss±TZ).
  - + Không được bảo vệ bằng chữ ký, có thể chỉnh sửa → không có giá trị pháp lý.
- **Thuộc tính signingTime trong PKCS#7/CMS**
  - + Nằm trong SignedAttributes, được bao phủ bởi chữ ký → có giá trị pháp lý.
- **RFC 3161 Timestamp Token (TST)**
  - + Token do TSA ( Time Stamp Augthority) cấp, xác nhận thời điểm tồn tại của dữ liệu.
  - + Được nhúng trong PKCS#7 dưới dạng timeStampToken (unsigned attribute).
  - + Cung cấp bằng chứng mạnh mẽ hơn về thời gian ký.
- **Document Timestamp (PAdES)**
  - + Một dạng chữ ký đặc biệt áp dụng cho toàn bộ tài liệu, thường dùng trong xác thực dài hạn (LTV).
- **DSS (Document Security Store)**
  - + Có thể chứa thêm timestamp và dữ liệu xác minh để lưu lâu dài.

➤ **Khác biệt giữa /M và RFC3161 timestamp**

- /M: chỉ là text → dễ bị sửa, không ràng buộc mật mã.
- RFC3161 timestamp: do TSA cấp, có chữ ký riêng → bằng chứng hợp pháp về thời điểm tài liệu tồn tại.

### 3) Các bước tạo và lưu chữ ký trong PDF (đã có private RSA)

- Viết script/code thực hiện tuần tự:

1. Chuẩn bị file PDF gốc.
  2. Tạo Signature field (AcroForm), reserve vùng /Contents (8192 bytes).
  3. Xác định /ByteRange (loại trừ vùng /Contents khỏi hash).
  4. Tính hash (SHA-256/512) trên vùng ByteRange.
  5. Tạo PKCS#7/CMS detached hoặc CAdES:
    - Include messageDigest, signingTime, contentType.
    - Include certificate chain.
    - (Tùy chọn) thêm RFC3161 timestamp token.
  6. Chèn blob DER PKCS#7 vào /Contents (hex/binary) đúng offset.
  7. Ghi incremental update.
  8. (LTV) Cập nhật DSS với Certs, OCSPs, CRLs, VRI
- .- Phải nêu rõ: hash alg, RSA padding, key size, vị trí lưu trong PKCS#7.
- Đầu ra: mã nguồn, file PDF gốc, file PDF đã ký.

#### 4) Các bước xác thực chữ ký trên PDF đã ký

- Các bước kiểm tra:

1. Đọc Signature dictionary: /Contents, /ByteRange.
2. Tách PKCS#7, kiểm tra định dạng.
3. Tính hash và so sánh messageDigest.
4. Verify signature bằng public key trong cert.
5. Kiểm tra chain → root trusted CA.
6. Kiểm tra OCSP/CRL.
7. Kiểm tra timestamp token.
8. Kiểm tra incremental update (phát hiện sửa đổi).

- Nộp kèm script verify + log kiểm thử

#### 5. Rủi ro chính và biện pháp giảm thiểu

Rủi ro	Mô tả	Phát hiện và Biện Pháp
<b>Thay đổi nội dung (Tampering)</b>	<ul style="list-style-type: none"><li>- Kẻ tấn công có thể sửa nội dung ngoài hoặc thay đổi hash, khiến trước vùng ByteRange, hoặc chỉnh sửa trực tiếp giá trị ByteRange.</li><li>- Làm chữ ký không còn khớp với dữ liệu ban đầu.</li></ul>	<ul style="list-style-type: none"><li>- Trong quá trình xác minh (verify), so sánh hash trên ByteRange với messageDigest trong PKCS#7.</li><li>- Nếu khác nhau → báo chữ ký không hợp lệ.</li></ul> <p><i>Biện pháp:</i> chỉ sử dụng incremental update đúng chuẩn PDF, trình verify phải</p>

		kiểm tra ByteRange và modification level.
<b>Replay / Incremental Update Abuse</b>	<ul style="list-style-type: none"> <li>- Kẻ xấu lợi dụng cơ chế incremental update của PDF để chèn thêm các Signature Dictionary giả hoặc che dấu sửa đổi trước đó.</li> <li>- Dễ khiến người dùng nhầm là tài liệu vẫn “được ký hợp lệ”.</li> </ul>	<ul style="list-style-type: none"> <li>- Yêu cầu timestamp bắt buộc từ TSA trong mỗi lần ký.</li> <li>- Ghi nhận và lưu toàn bộ trailer/timestamp vào DSS (Document Security Store).</li> <li>- Trình xác minh phải phân tích lịch sử incremental để phát hiện hành vi bất thường (ví dụ: signature xuất hiện sau cùng không liên quan đến bản gốc).</li> </ul>
<b>Không kiểm tra thu hồi chứng chỉ (Revocation: OCSP / CRL)</b>	<ul style="list-style-type: none"> <li>- Trường hợp chứng chỉ của người ký đã bị thu hồi nhưng hệ thống verify không kiểm tra OCSP/CRL, dẫn đến chấp nhận chữ ký không hợp lệ.</li> </ul>	<ul style="list-style-type: none"> <li>- Trong quá trình ký, nhúng OCSP responses / CRL vào DSS.</li> <li>- Trình xác minh cần thực hiện kiểm tra trạng thái chứng chỉ (revocation check).</li> </ul> <p><i>Hỗ trợ LTV (Long-Term Validation) để đảm bảo tài liệu vẫn xác minh được sau nhiều năm.</i></p>
Lộ khóa riêng (Private Key)	<ul style="list-style-type: none"> <li>- Khóa riêng bị đánh cắp hoặc lưu trữ kém bảo mật, khiến kẻ xấu có thể ký thay cho người hợp</li> </ul>	<ul style="list-style-type: none"> <li>- Khóa riêng bị đánh cắp hoặc lưu trữ kém bảo mật, khiến kẻ xấu có thể ký thay</li> </ul>



Exposure) / Quản trị yếu	pháp. - Thường do người dùng lưu khóa trong máy tính hoặc chia sẻ file .pfx không an toàn.	cho người hợp pháp. - Thường do người dùng lưu khóa trong máy tính hoặc chia sẻ file .pfx không an toàn.
-----------------------------	---	---

## 6. Khuyến nghị kỹ thuật

- Dùng SHA-256 hoặc mạnh hơn cho message digest.
- Dùng RSA 2048+ hoặc RSA-PSS(khuyến nghị) cho chữ ký và server TSA đáng tin cậy cho timestamp RFC -3161.
- Thực hiện LTV (PAdES-LTV) bằng cách nhúng chứng thư, OCSP/CRL và timestamp token và DSS.
- Kiểm tra modification level và đảm bảo trình verify báo rõ ràng khi có incremental updates.

## 7. Minh họa File đính kèm

Trong bài nộp kèm các file mẫu :

- original.pdf -file gốc.
- signed.pdf- file sau khi đã ký ( chứa/Contents PKCS#7 và ByteRange hợp lệ).
- tampered.pdf - phiên bản đã bị chỉnh sửa ngoài vùng được ký ( dùng để minh chứng verify thất bại).

## 8.Kết luận

Bài tập này giúp hiểu rõ cơ chế lưu và xác minh chữ ký trong PDF thông qua các thành phần /ByteRange ,/Contents và incremental update. Trường /M chỉ lưu thời gian hiện thị không, không có giá trị pháp lý, trong khi timestamp RFC-3161 trong PKCS#7 mới chứng minh được thời điểm ký thực tế. Để đảm bảo tính pháp lý và xác minh lâu dài (LTV), cần kết hợp PKCS#7 + timestamp từ TSA và nhúng dữ liệu OCSP/CRL vào DSS theo chuẩn PAdES.