BÀI TẬP VỀ NHÀ – MÔN: AN TOÀN VÀ BẢO MẬT THÔNG TIN

Chủ đề: Chữ ký số trong file PDF

Giảng viên hướng dẫn: Đỗ Duy Cốp

Sinh viên thực hiện: Nguyễn Lam Sơn

Lớp: 58KTPM Thời gian nộp: 31/10/2025

I. MÔ TẢ CHUNG

Bài tập yêu cầu sinh viên nghiên cứu, phân tích và thực hành việc nhúng, xác thực chữ ký số trong file PDF. Nội dung bám sát chuẩn PDF
1.7/PDF 2.0 và PAdES/ETSI, đồng thời sử dụng công cụ thực thi như
iText7, OpenSSL hoặc PyPDF.

II. CẤU TRÚC PDF LIÊN QUAN CHỮ KÝ

Trong tài liệu PDF, chữ ký số được lưu trữ thông qua các đối tượng có
cấu trúc dạng cây. Dưới đây là các thành phần chính:

Thành phần Chức năng

Catalog (/Root) Gốc của tài liệu, liên kết đến các đối tượng khác như Pages và AcroForm

Pages tree (/Pages) Quản lý danh sách các trang trong tài liệu

Page object Mỗi trang cụ thể chứa nội dung hiển thị

Resources & Content streams Lưu văn bản, hình ảnh và các nội dung hiển thị khác

AcroForm (/AcroForm) Lưu thông tin biểu mẫu, trong đó có trường chữ
ký

Signature field (Widget) Định nghĩa vùng hiển thị chữ ký trên trang

Signature dictionary (/Sig) Lưu thông tin chữ ký, bao gồm /Contents và
/ByteRange

/ByteRange Chỉ định vùng byte được ký, ngoại trừ phần chứa chữ ký

/Contents Chứa dữ liệu chữ ký PKCS#7 hoặc CMS

DSS (Document Security Store) Lưu chứng chỉ, OCSP, CRL phục vụ
xác minh lâu dài (LTV)

Sơ đồ cấu trúc đối tượng chữ ký PDF:

Catalog ↓ Pages Tree → Page → Content Streams ↓ AcroForm → Signature Field (Widget) ↓ Signature Dictionary (/Sig) ├──

/ByteRange ├── /Contents ├── /M └── /Filter, /SubFilter

III. THỜI GIAN KÝ ĐƯỢC LƯU Ở ĐÂU

Thông tin thời gian có thể được lưu ở nhiều vị trí khác nhau trong PDF:

- /M: Lưu dạng văn bản trong Signature dictionary (không có giá trị pháp lý). - Timestamp token (RFC 3161): Lưu trong PKCS#7 attribute
timeStampToken. - Document timestamp object (PAdES): Dạng đối tượng /DocTimeStamp. - DSS (Document Security Store): Lưu timestamp và dữ liệu xác minh.

Sự khác biệt: /M chỉ là metadata, trong khi timestamp RFC3161 là chữ
ký số được TSA xác nhận, có giá trị pháp lý vì chứng minh được thời
điểm tồn tại của chữ ký.

IV. CÁC BƯỚC TẠO VÀ LƯU CHỮ KÝ (ĐÃ CÓ PRIVATE RSA)

Quy trình thực hiện ký số trong PDF bao gồm:

1. 1. Chuẩn bị file PDF gốc.
2. 2. Tạo Signature field (AcroForm) và vùng /Contents (8192 bytes).
3. 3. Xác định /ByteRange (loại trừ vùng /Contents).
4. 4. Tính hash SHA-256 trên vùng ByteRange.
5. 5. Tạo gói PKCS#7/CMS detached, kèm chứng chỉ.
6. 6. Chèn chữ ký vào /Contents đúng vị trí.
7. 7. Ghi incremental update.
8. 8. (Tuỳ chọn) Cập nhật DSS với Certs, OCSPs, CRLs.

Ví dụ minh hoạ bằng Python:

from PyPDF2 import PdfReader, PdfWriter from cryptography.hazmat.primitives import hashes, serialization from cryptography.hazmat.primitives.asymmetric import padding import hashlib, datetime reader = PdfReader("original.pdf")
writer = PdfWriter() for page in reader.pages: writer.add_page(page) with open("original.pdf", "rb") as f: data = f.read() digest = hashlib.sha256(data).digest() with open("private_key.pem", "rb") as key_file: private_key = serialization.load_pem_private_key(key_file.read(), password=None) signature = private_key.sign(digest, padding.PKCS1v15(), hashes.SHA256()) print("Đã tạo chữ ký số thành công!")

V. CÁC BƯỚC XÁC THỰC CHỮ KÝ TRÊN PDF ĐÃ KÝ

9. 1. Đọc Signature dictionary: /Contents, /ByteRange.10. 2. Tính lại hash vùng ByteRange.11. 3. So sánh messageDigest
trong PKCS#7.
12. 4. Xác thực chữ ký bằng public key.
13. 5. Kiểm tra chuỗi chứng chỉ (chain).
14. 6. Kiểm tra OCSP/CRL.
15. 7. Kiểm tra timestamp RFC3161.
16. 8. Kiểm tra incremental update để phát hiện sửa đổi.
Mã Python minh hoạ xác thực chữ ký:
from PyPDF2 import PdfReader import hashlib reader =
PdfReader("signed.pdf") sig =
reader.trailer["/Root"]["/AcroForm"]["/Fields"][0].get_object()
contents
= bytes.fromhex(sig["/V"]["/Contents"]) byte_range =
sig["/V"]["/ByteRange"] with open("signed.pdf", "rb") as f:
pdf_bytes =
f.read() data_to_hash = pdf_bytes[byte_range[0]:byte_range[1]] +
pdf_bytes[byte_range[2]:byte_range[3]] digest =
hashlib.sha256(data_to_hash).digest() print("Đã xác minh hash
thành
công!")

VI. RỦI RO BẢO MẬT VÀ KẾT LUẬN

Một số rủi ro bảo mật trong quá trình sử dụng chữ ký số PDF: - Lộ private key do quản lý không an toàn. - Tấn công padding oracle trên RSA. - Replay chữ ký trên tài liệu khác.
- Không xác minh OCSP dẫn đến tin vào chứng chỉ đã bị thu hồi. - Sửa
incremental update để chèn nội dung độc hại.
Kết luận: Việc hiểu rõ cấu trúc và quy trình ký số trong PDF giúp sinh
viên nắm vững nguyên lý và đảm bảo tính toàn vẹn, xác thực và chống
chối bỏ của tài liệu điện tử.