Tài liệu hướng dẫn tích hợp thư viện ký số và xác thực dữ liệu

Contents

1. Giới thiệu 2
2. Cấu trúc bộ thư viện 2
2.1. Mô đun VGCAToolkit 2
2.2. Mô đun PDFVerifier 3
2.3. Mô đun CertChecker 3
2.4. Mô đun OCSPVerifier 4
3. Hướng dẫn tích hợp 5
3.1. Hướng dẫn kiểm tra chứng thư số 5
3.2. Hướng dẫn ký số tệp PDF 5
3.3. Hướng dẫn xác thực chữ ký số tệp PDF 5

1. Giới thiệu

Thư viện ký số và xác thực dữ liệu được xây dựng trên ngôn ngữ Java phục vụ cho việc tích hợp trên các ứng dụng chạy trên hệ điều hành Android.

Thư viện cung cấp các chức năng ký số và xác thực tệp PDF sử dụng SIM PKI và các dịch vụ chứng thực chữ ký số như TSA, OCSP, CRL.

Thư viện hỗ trợ các tiêu chuẩn mật mã an toàn như: hàm băm SHA256, thuật toán ký số RSA lược đồ RSASSA-PKCS1-v1.5 và RSASSA-PSS.

2. Cấu trúc bộ thư viện

Thư viện bao gồm các mô dun sau:

- Mô đun VGCAToolKit cung cấp các chức năng ký số tệp PDF sử dụng lược đồ RSASSA-PKCS1-v1.5 và RSASSA-PSS

- Mô đun PDFVerifier có chức năng xác thực chữ ký trên tệp PDF

- Mô đun CertChecker có chức năng kiểm tra chứng thư số, trong đó sử dụng dịch vụ CRL

- Mô đun OCSPVerifier có chức năng kiểm tra trạng thái thu hồi chứng thư số sử dụng dịch vụ OCSP.

2.1. Mô đun VGCAToolkit

a. Hàm lấy chứng thư số

- Giao diện hàm: public static String GetCert(String phonenumber)

- Tham số đầu vào:

    phonenumber : là số điện thoại của người ký.

- Kết quả trả về:

    Nội dung chứng thư số dưới định dạng Base64

b. Hàm ký số tệp PDF sử dụng lược đồ RSASSA-PKCS1-v1.5

- Giao diện hàm:

```
public static void signPDF(String fileNguon, String fileDich,
        String phonenumber,Rectangle rectangle,
        byte[] imageBytes, byte[] fontBytes, Certificate[] chain)
```

- Tham số đầu vào:

    fileNguon: Đường dẫn tệp cần ký

    fileDich: Đường dẫn lưu tệp ký số

phonenumber: Số điện thoại người ký

rectangle: Vị trí và kích thước chữ ký

imageBytes: Hình ảnh chữ ký

fontBytes: Font hiển thị thông tin chữ ký

chain: Đường dẫn chứng thực của chứng thư số người ký

c. Hàm ký số tệp PDF sử dụng lược đồ RSASSA-PSS

- Giao diện hàm:

```
public static void signPSS(String fileNguon, String fileDich,
        String phonenumber,Rectangle rectangle,
        byte[] imageBytes, byte[] fontBytes, Certificate[] chain)
```

- Tham số đầu vào:

fileNguon: Đường dẫn tệp cần ký

fileDich: Đường dẫn lưu tệp ký số

phonenumber: Số điện thoại người ký

rectangle: Vị trí và kích thước chữ ký

imageBytes: Hình ảnh chữ ký

fontBytes: Font hiển thị thông tin chữ ký

chain: Đường dẫn chứng thực của chứng thư số người ký

2.2. Mô đun PDFVerifier

Hàm xác thực chữ ký trên tệp PDF

- Giao diện hàm:

```
public static ArrayList<String[]> verifySignatures(String path,
        X509Certificate CACert)
```

- Tham số đầu vào:

Path: Đường dẫn tệp PDF cần kiểm tra

CACert: Chứng thư số CA

- Kết quả trả về:

Thông tin kiểm tra tất cả chữ ký số trên tệp PDF.

2.3. Mô đun CertChecker

a. Hàm khởi tạo

- Giao diện hàm:

public CertChecker(X509Certificate cert, X509Certificate issuer, Date time)

- Tham số đầu vào:

cert: chứng thư số cần kiểm tra

issuer: chứng thư số CA

time: thời gian kiểm tra

b. Hàm kiểm tra thời gian hiệu lực

- Giao diện hàm:

public int Valid()

- Kết quả trả về:

0 : Chứng thư số còn hiệu lực

1: Chứng thư số hết hiệu lực

2: Chứng thư số chưa có hiệu lực

c. Hàm kiểm tra trạng thái thu hồi qua CRL

- Giao diện hàm:

public int CheckCRLOnline()

- Kết quả trả về:

0 : chứng thư số chưa bị thu hồi

1: không lấy được CRL

2: không tìm thấy thông tin CRL trên chứng thư số

3: không tải được CRL

4: chứng thư số đã bị thu hồi

5: CRL không hợp lệ

2.4. Mô đun OCSPVerifier

Hàm kiểm tra chứng thư số qua OCSP

- Giao diện hàm:

public int checkRevocationStatus(X509Certificate peerCert,

X509Certificate issuerCert)

- Tham số đầu vào:

peerCert: Chứng thư số cần kiểm tra

issuerCert: Chứng thư số CA

- Kết quả trả về:

0 : chứng thư số hợp lệ

1: chứng thư số đã bị thu hồi

2: kết quả unknown.

3. Hướng dẫn tích hợp

3.1. Hướng dẫn kiểm tra chứng thư số

Bước 1: Khai báo lớp CertChecker với đầu vào là chứng thư số người dùng cert, chứng thư số CA là cacert, và thời gian kiểm tra time:

CertChecker checker = new CertChecker(cert, cacert, time);

Bước 2: Kiểm tra thời gian hiệu lực chứng thư số:

int rv = checker.Valid();

Nếu rv = 0 thì chứng thư số còn hiệu lực, ngược lại nếu khác 0 chứng thư số không còn hiệu lực

Bước 3: Kiểm tra trạng thái thu hồi của chứng thư số

int rv = checker.CheckCRLOnline();

Kiểm tra rv = 0 thì chứng thư số hợp lệ. Nếu khác 0 chứng thư số không hợp lệ.

3.2. Hướng dẫn ký số tệp PDF

Sử dụng hàm signPDF hoặc signPSS trong mô đun VGCAToolkit để thực hiện ký số tệp PDF. Hàm signPDF sử dụng lược đồ RSASSA-PKCS1-v1.5 còn hàm signPSS sử dụng lược đồ RSASSA-PSS. Người dùng gọi trực tiếp hàm để thực hiện ký số, tham số đầu vào là đường dẫn tệp cần ký, đường dẫn tệp ký số, số điện thoại, vị trí, kích thước chữ ký, ảnh chữ ký, font hiển thị, đường dẫn chứng thực chứng thư số người ký:

VGCAToolkit.signPDF(inputFile, outputFile, “84167234479”, rect, image, fonts, chain);

3.3. Hướng dẫn xác thực chữ ký số tệp PDF

Sử dụng hàm verifySignatures trong mô đun PDFVerifier để kiểm tra chữ ký số trên tệp PDF. Người dùng gọi trực tiếp hàm để thực hiện xác thực, tham số đầu vào là đường dẫn tệp và chứng thư số CA.

```
ArrayList<String[]> result = PDFVerifier.verifySignatures(inputFile, cacert);
```