**ĐỒ ÁN 03 – LÀM QUEN VỚI OPENSSL**

**3.3 TÌM HIỂU VỀ CHỮ KÝ ĐIỆN TỬ CỦA RSA CỦA OPENSSL**

**I. Cách OpenSSL sử dụng tệp chứa khóa bí mật để ký các tệp tin:**

Sau khi đã có khóa bí mật RSA chứa trong tệp <priv.pem> ta thực hiện việc ký với OpenSSL như sau :

$openssl pkeyutl -in <mess> -out <sign> -inkey <priv.pem> -sign

Trong đó:

• <mess> là tệp chứa tin nhắn cần ký,

• <sign> là tệp chứa chữ ký.

• <priv.pem> là tệp tin chứa khóa bí mật RSA mà OpenSSL sinh ra.

**Các bước thực hiện ký :**

+ **Bước 1:** Hash thông điệp trong <mess> .Khi không cung cấp thuật toán MD (Message Digest )cụ thể Openssl pkeyutl sử dụng SHA256 (152 bit) .

**+ Bước 2:** Thêm phần padding vào giá trị hash

Padding mặc định của openssl là PCKS1 v1.5 cho thuật toán SHA256 có dạng

00||01||PS||00||T||H

**PS :** Chuỗi hex FF được thêm vào để độ dài <mess> bằng độ dài của <priv.pem>

**T:** Phần định danh của cấu trúc chữ ký ( thường gọi là MAGIC byte )

|  |  |  |
| --- | --- | --- |
| Hash Function | Length (bit) | Octet String |
| MD2 | 144 | 3020300c06082a864886f70d020205000410 |
| MD5 | 144 | 3020300c06082a864886f70d020205000410 |
| SHA-1 | 120 | 3021300906052b0e03021a05000414 |
| SHA-256 | 152 | 3031300d060960864801650304020105000420 |
| SHA-384 | 152 | 3041300d060960864801650304020105000430 |
| SHA-512 | 152 | 3051300d060960864801650304020105000440 |

*MAGIC bytes (padding) của một số thuật toán băm*

H : giá trị hash của <mess> tính ở Bước 1

**+ Bước 3 :** Từ file <priv.pem> có được n, d

**+ Bước 4:** Ký vào thông điệp đã hash và thêm padding trong thuật toán RSA :

Private exponent

S = md mod (n)

Signature hashed with padding message modulus

**+ Bước 5:** Lưu chữ ký vào file <sign>

A diagram of a padlock and key

Description automatically generated

**II. Cách OpenSSL sử dụng tệp chứa khóa công khai để xác thực tập tin :**

Sau khi có khóa công khai RSA chứa trong tệp <pub.pem>, ta thực hiện việc xác thực với OpenSSL như sau :

**$openssl pkeyutl -in <mess> -sigfile <sign> -inkey <pub.pem> -pubin -verify**

Trong đó:

• <mess> là tệp chứa tin nhắn,

• <sign> là tệp chứa chữ ký cần xác thực.

• <pub.pem> là tệp tin chứa khóa công khai RSA mà OpenSSL sinh ra.

**Các bước thực hiện xác thực :**

**+ Bước 1:** Từ file <pub.pem> ta có được n, e

+ **Bước 2:** Chuyển chữ ký trong <sign> về lại dạng thông điệp đã được mã hóa và thêm padding trong thuật toán RSA :

Public exponent

M = Se mod (n)

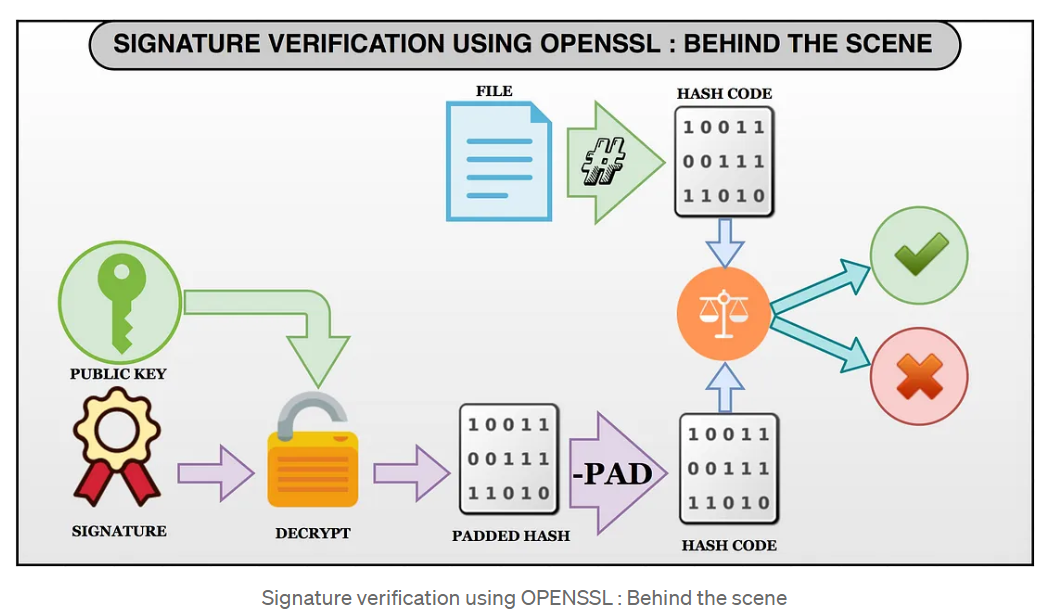
hashed with padding message signature modulus

**+ Bước 3:** Cắt phần padding khỏi hash message

**+ Bước 4:** So sánh phần hash message thu được và hash <mess> xem có giống nhau không

• Giống nhau : Xuất ra Signature Verified Successfully

• Khác nhau : Xuất ra Signature Verified Failed



**III. Mã nguồn**

**3.1) Mã nguồn :**

Chương trình ký đọc tệp chứa khóa bí mật <priv.pem> và tệp chứa tin nhắn cần ký <mess>, xuất ra tệp chứa chữ ký <sign> .

**Trong thư mục /Source/3.3\_RSA\_Sign/RSA\_Sign.py**

Chương trình xác thực đọc tệp chứa khóa công khai <pub.pem>, tệp chứa tin nhắn <mess>, và tệp chứa chữ ký <sign>, xuất ra câu trả lời tin nhắn được xác thực bởi chữ ký hay không

**Trong thư mục /Source/3.3\_RSA\_Sign/RSA\_Verify.py**

**3.2) Ngôn ngữ lập trình :** **Python verison 3.11.4 , openssl command** sử dụng **openssl version 3.2.0**

**3.3) Môi trường chạy thử nghiệm :** Windown 11, 64 bits

**3.4) Thư viện cần cài đặt :** **cryptography**

Cách thức cài đặt :

pip install cryptography

Cách thức chạy chương trình :

$python RSA\_Sign.py <mess> <sign> <priv.pem>

<mess> : Đường dẫn đến file chứa thông điệp cần ký

<sign> : Đường dấn đến file chứa chữ ký do chương trình tạo ra

<priv.pem> : Đường dẫn đến file private key

$python RSA\_Verify.py <mess> <sign> <pub.pem>

<mess> : Đường dẫn đến file chứa thông điệp đã ký

<sign> : Đường dấn đến file chứa chữ ký

<pub.pem> : Đường dẫn đến file public key

**3.5) Video demo :**

Video demo được tải lên youtube, đường dẫn truy cập :

<https://youtu.be/AYncbXDR_HA>

**IV. Tài liệu tham khảo :**

<https://medium.com/@bn121rajesh/rsa-sign-and-verify-using-openssl-behind-the-scene-bf3cac0aade2>

<https://en.wikipedia.org/wiki/PKCS_1>

<https://www.openssl.org/docs/manmaster/man1/pkeyutl.html>