|  |
| --- |
| HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  **KHOA CÔNG NGHỆ THÔNG TIN 1** |
|  |
| **BÀI THỰC HÀNH 16**  **THỰC TẬP CƠ SỞ** |
| **Họ và tên : Đinh Quang Hiếu**  **Mã sinh viên: B19DCAT065**  **Giảng viên giảng dạy: Hoàng Xuân Dậu** |
| **HÀ NỘI, THÁNG 5/2022** |

**Bài 16: Tìm hiểu và cài đặt mã hóa bất đối xứng RSA**

**1.1.Mục đích**

Sinh viên tìm hiểu một giải thuật mã hóa phổ biến và lập trình được chương trình mã hóa và giải mã sử dụng ngôn ngữ lập trình phổ biến như C/C++/Python/Java, đáp ứng chạy được với số lớn

**1.2 Nội dung thực hành**

**1.2.1 Tìm hiểu lý thuyết**

**- Tìm hiểu về lập trình số lớn với các phép toán cơ bản**

**- Tìm hiểu về giải thuật mật mã khóa công khai RSA**

Hệ mã RSA được giới thiệu vào năm 1977 bởi 3 nhà khoa học Ron Rivest, Adi Shamir, Len Adlerman. Đây là một trong những hệ mã được sử dụng phổ biến nhất hiện nay, ứng dụng cho truyền dữ liệu an toàn qua internet, email. RSA còn là nền tảng mật mã cho các giao thức SSL/TLS, SET, SSH, PGP, … RSA cũng được ứng dụng trong chữ ký số Digial Signature.

Breaking RSA là dạng bài thường xuyên gặp phải trong các cuộc thi CTF dưới nhiều hình thức. Bài viết này tôi sẽ mô tả ngắn gọn về hệ mã này, liệt kê các điểm yếu có thể khai thác của nó, cũng như các công cụ toán học, các tool cần thiết giúp giải quyết các bài RSA hay gặp khi chơi CTF.

RSA Cryptosystem

RSA thuộc nhóm hệ mã khóa công khai, dựa vào độ khó của bài toán phân tích 1 số ra thừa số nguyên tố (factoring problem). Để tạo cặp khóa Public key và Private key, Alice cần:

• Chọn 2 số nguyên tố lớn p, q với p ≠ q

• Tính n = pq

• Tính giá trị hàm số Ơle φ(n) = (p-1)(q-1)

• Chọn 1 số e sao cho 1 < e < φ(n) và gcd(e,φ(n)) = 1

• Tính d = e-1 (mod φ(n)), số d thỏa mãn ed ≡ 1 (mod φ(n))

Public Key gồm:

• n – module.

• e – số mũ mã hóa.

Private Key gồm:

• n – module.

• d – số mũ giải mã.

Khi Bob muốn gửi một tin nhắn M cho Alice, Bob chuyển M thành một số m < n theo 1 cách thỏa thuận trước. Bob sẽ tính ra bản mã c từ bản rõ m theo công thức:

c = me (mod n)

Để giải mã, Alice dùng Private Key của mình để tính ngược lại:

m = cd (mod n)

Quá trình giải mã có thể thu lại được m ban đầu là do:

c d ≡ (me ) d ≡ med (mod n) ≡ m (mod n) hay m = cd (mod n)

Dấu ≡ cuối cùng là tôi đã áp dụng định lý Euler. Chi tiết hơn về thiết kế hệ mã cũng như ví dụ có thể đọc ở đây RSA – Wikipedia

Độ mạnh của hệ mã RSA dựa trên việc bạn cần phân tích được n ra thừa số nguyên tố để tính d nếu muốn phá mã, và đến nay chưa có giải thuật nào hiệu quả trong thời gian đa thức giúp ta phân tích thừa số nguyên tố đối với các số lớn.

Hệ mã RSA nếu được thiết kế một cách đúng đắn với việc chọn các tham số n, p, q, e hợp lý thì sẽ rất an toàn, thế nhưng trong các bài CTF, các tham số này thường được chọn theo một cách nào đó khiến cho hệ mã yếu đi và dễ bị tấn công.

**1.2.2 Chuẩn bị môi trường**

**- Môi trường lập trình theo mong muốn.**

**1.2.3 Các bước thực hiện và kết quả cần đạt**

**a) Các bước thực hiện**

- Lập trình thư viện số lớn với các phép toán cơ bản để sử dụng trong giải thuật mã hóa/giải mã RSA

- Thử nghiệm chứng minh thư viện hoạt động tốt với các ví dụ phép toán cho số lớn

A screenshot of a computer

Description automatically generated

- Lập trình giải thuật mã hóa và giải mã

+ Sử dụng thư viện BigInteger để triển khai mã hóa RSA

Graphical user interface, text

Description automatically generated

+ Khởi báo các giá trị của mã hóa RSA

A screenshot of a computer

Description automatically generated with medium confidence

+ Khởi tạo các giá trị của mã hóa RSA

A screenshot of a computer

Description automatically generated

- Mã hóa 6 số:

Graphical user interface, text

Description automatically generated

- Mã hóa với số cực lớn

A screenshot of a computer

Description automatically generated with medium confidence

- Mã hóa xâu kí tự dài và có nhiều kí tự đặc biệt như \*,#...

Text

Description automatically generated

- Mã hóa xâu kí tự “I am B19DCAT065”

Text

Description automatically generated