Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут

КРИПТОГРАФІЯ

КОМП'ЮТЕРНИЙ ПРАКТИКУМ №4

Вивчення криптосистеми RSA та алгоритму електронного підпису; ознайомлення з методами генерації параметрів для асиметричних криптосистем

Виконали:

Акент'єв Влад, Шапоренко Микита

Група: ФБ-06

Київ – 2022

Мета роботи

Ознайомлення з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практичне ознайомлення з системою захисту інформації на основі криптосхеми RSA, організація з використанням цієї системи засекреченого зв'язку й електронного підпису, вивчення протоколу розсилання ключів

Хід роботи

1. Написати функцію пошуку випадкового простого числа з заданого інтервалу або заданої довжини, використовуючи датчик випадкових чисел та тести перевірки на простоту. В якості датчика випадкових чисел використовуйте вбудований генератор псевдовипадкових чисел вашої мови програмування. В якості тесту перевірки на простоту рекомендовано використовувати тест Міллера-Рабіна із попередніми пробними діленнями. Тести необхідно реалізовувати власноруч, використання готових реалізацій тестів не дозволяється.

2. За допомогою цієї функції згенерувати дві пари простих чисел p, q і 1 1 p , q довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб $pq \leq p1q1$; p і q – прості числа для побудови ключів абонента A, 1 p і q1 – абонента B.

3. Написати функцію генерації ключових пар для RSA. Після генерування функція повинна повертати та/або зберігати секретний ключ (d, p,q) та відкритий ключ (n,e) . За допомогою цієї функції побудувати схеми RSA для абонентів A і B – тобто, створити та зберегти для подальшого використання відкриті ключі (e,n) , (,) 1 n1 e та секретні d і d1 .

4. Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів A і B. Кожна з операцій (шифрування, розшифрування, створення цифрового підпису, перевірка цифрового підпису) повинна бути реалізована окремою процедурою, на вхід до якої повинні подаватись лише ті ключові дані, які необхідні для її виконання. За допомогою датчика випадкових чисел вибрати відкрите повідомлення M і знайти криптограму для абонентів A и B, перевірити правильність розшифрування. Скласти для A і B повідомлення з цифровим підписом і перевірити його.

5. За допомогою раніше написаних на попередніх етапах програм організувати роботу протоколу конфіденційного розсилання ключів з підтвердженням справжності по відкритому каналу за допомогою алгоритму RSA. Протоколи роботи кожного учасника (відправника та приймаючого) повинні бути реалізовані у вигляді окремих процедур, на вхід до яких повинні подаватись лише ті ключові дані, які необхідні для виконання. Перевірити роботу програм для випадково обраного ключа $0 < k < n$.

Результати

Message : 550

-------A-------

Public exponent: 100001

Public n:
22567734247098143656746626025424180778502360054019602287129462697400353222560714327669172928766967109467909614475702280297714950905365251864275248333177193

Encrypted A:
17476986337044044962955870588881974794248745960309488102139870362934289270590881638089585136999685984719319286320181586766447171636703210567702793660465454

Decrypted A: 550

Sign A:
72342813165187885628712302696158196795515120710327670465116627709367890973632884367492374978449399349323984985663637540338667997894016087500854088794574555

Verify A: True

-------B-------

Public exponent1: 100001

Public n1:
34206572980826333200735636961950984240863311134225994503291798037992679157118330498768277132951331989918174471541658169098952846250440477197736071198674451

Encrypted B:
39547616701724621279058982967235168096824484389640020691087519331730284396941861050290766575770074939800448463695121381444549198139004229569860462241225527

Decrypted B: 550

Sign B:
33237315664585475383428832676017621740434275416015299200685453526805018276328587345591061516850112535124263230980497850682072237974165288095403664024091909

Verify B: True

Key ok?: True

```
Message :  550
-------A-------
Public exponent:  100001
Public n:  22567734247098143656746626025424180778502360054019602287129462697400353222560714327669172928766967109467909614475702280297714950905365251864275248333177193
Encrypted A:  17476986333704404496295587058888197479424874596030948810213987036293428927059088163808958513699968598471931928632018158676644717163670321056770279366046454
Decrypted A:  550
Sign A:  72342813165187885628712302696158196795515120710327670465116627709367890973632884367492374978449399349323984985663637540338667997894016087500854088794574555
Verify A:  True
-------B-------
Public exponent1:  100001
Public n1:  34206572980826333200735636961950984240863311134225994503291798037992679157118330498768277132951331989918174471541658169098952846250440477197736071198674451
Encrypted B:  39547616701724621279058982967235168096824484389640020691087519331730284396941861050290766575770074939800448463695121381444549198139004229569860462241225527
Decrypted B:  550
Sign B:  33237315664585475383428832676017621740434275416015299200685453526805018276328587345591061516850112535124263230980497850682072237974165288095403664024091909
Verify B:  True
Key ok?:  True
```

Висновки

В результаті лабораторної роботи, ми ознайомились з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми. Ознайомились з системою захисту інформації на основі криптосхеми RSA. Вивчення протокол розсилання ключів