Міністерство освіти і науки України

НТУУ«Київський політехнічний інститут»

Фізико-технічний інститут

«КРИПТОГРАФІЯ»

КОМП’ЮТЕРНИЙ ПРАКТИКУМ №4

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

Виконали:

Студентки групи ФБ-81

Лобанова Олександра,

Прима Аліна

Перевірив: Чорний О. М.

Хід роботи

Мета та основні завдання 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.

Порядок і рекомендації щодо виконання 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 необхід дно 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 ≤ 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) , (e1,n1) 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.

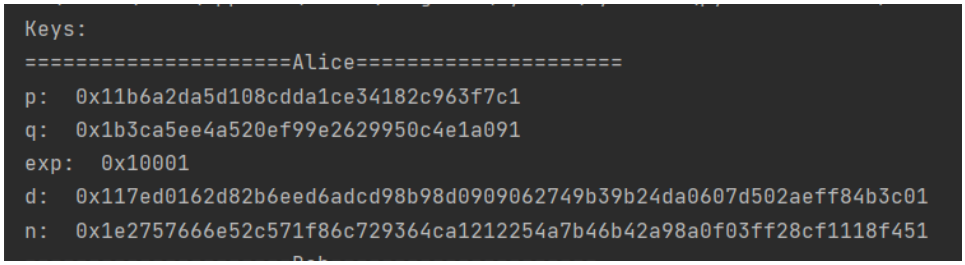
5. За допомогою раніше написаних на попередніх етапах програм організувати роботу протоколу конфіденційного розсилання ключів з підтвердженням справжності по відкритому каналу за допомогою алгоритму RSA. Протоколи роботи кожного учасника (відправника та приймаючого) повинні бути реалізовані у вигляді 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.

Кожну операцію рекомендується перевіряти шляхом взаємодіїіз тестовим середовищем, розташованим за адресою http://asymcryptwebservice.appspot.com/?section=rsa.

Виконання роботи

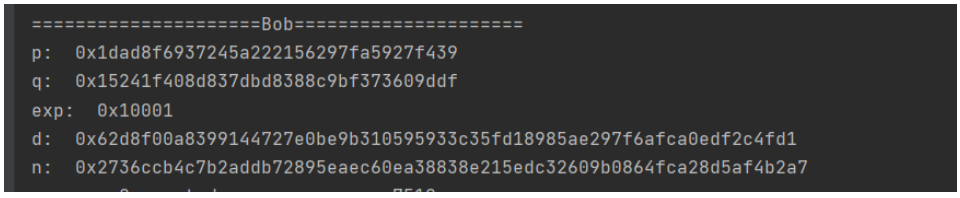
В ході виконання лабораторної роботи було створено функції піднесення до степеня великих чисел за схемою Горнера, пошук оберненого числа за модулем, перевірку на псевдопросте число, використовуючи імовірнісний тест Ферма, функцію генерування псевдопростого числа. Була створена функція для генерування ключів схеми RSA, функції зашифрування, розшифрування, створення цифрового підпису, функція перевірки цифрового підпису, функція відправлення та отриманя повідомлення. В ході виконання лабраторної роботи було створенно підключення та взаємодія з тестовим середовищем.

Скріни виконання програми



Keys of Alice:

p: 0x11b6a2da5d108cdda1ce34182c963f7c1
q: 0x1b3ca5ee4a520ef99e2629950c4e1a091
exp: 0x10001
d: 0x117ed0162d82b6eed6adcd98b98d0909062749b39b24da0607d502aeff84b3c01
n: 0x1e2757666e52c571f86c729364ca1212254a7b46b42a98a0f03ff28cf1118f451



Keys of Bob:

p: 0x1dad8f6937245a222156297fa5927f439
q: 0x15241f408d837dbd8388c9bf373609ddf
exp: 0x10001
d: 0x62d8f00a8399144727e0be9b310595933c35fd18985ae297f6afca0edf2c4fd1
n: 0x2736ccb4c7b2addb72895eaec60ea38838e215edc32609b0864fca28d5af4b2a7

Робота з тестовим середовищем

```
========Generated open message:  7510  =========
=======Verified=======
Decryped message:  7510
```

========Generated open message: 7510 =========
=======Verified=======
Decryped message: 7510

```
Server public key:{'modulus': '9777AFD60212F30C6F5E5C6F2C991E2FFBCDBA3A2A096422560339E04311BBD10D4B80C2F65', 'publicExponent': '10001'}
========Generated open message:  6727  =========
msg server got after verification: 6727
verification result: True
```

Server public key:{'modulus':
'9777AFD60212F30C6F5E5C6F2C991E2FFBCDBA3A2A096422560339E04311BBD10D4B80C2F65',
'publicExponent': '10001'}
========Generated open message: 6727 =========
msg server got after verification: 6727
verification result: True

Висновок

В ході лабораторної роботи ми ознайомилися з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA. Отримали практичні навички криптозахисту на основі системи RSA. Ознайомилися з практичним застосуванням схеми Горнера та дізналися методи перевірки на псевдопросте число. Покращили навчики у роботі з сервісом контролю версій та отримали базові навички у роботі з бібліотекою request.