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

Лабораторна робота №4

З дисципліни «Криптографія»

Виконали:

студенти 3 курсу ФТІ
групи ФБ-83
Волинко Д.В.
Бондаренко.Р.С.

Перевірив:

Чорний О. М.

Вариант - 5

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.

Порядок виконання:

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 p1 , q1 довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб $pq \leq p1q1$; p і q – прості числа для побудови ключів абонента A, p1 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 $0< k < n$.

Результати виконання:

Information about users:

Name - Alice

Message - 135,

d - 48255835906582409605518562451936769794325345625821952963150278149293870536257718592633298862126907676010354286232495788357142179306159219076340832899937,

e - 65537,

n - 75550471041798647355873627028489729622806979844230609922264208768998384933945874699350226559542833874638788266521512938675016883397838448044625587578174 3 ,

q - 17794865786518973869788014281827028175363619274287450078888861165568941274199,

p - 42456330914860924661477544650109899624792886070714875254546565146394252767657,

sign - 0

Name - Bob

Message - 26,

d - 73768231139530407546338216416661622027908409552781947053362249245679752258754119018454189375145587934098314610914799285744158759743729551954458495782041 7,

e - 65537, n - 95690053326037731713563479787398862356610522670084327218023508675536170136026263999518129618973225096394281916620540693130133573881430331735856002338465 7 ,

q - 84980696649069795061630881659994853064148141298487328066851816062338431687153, p - 11260210506533328360940048040747505445120241259715058150653281388586170588769, sign - 0

starting messaging:

Sender's message: 26, encrypted message: 30336170113889804725801663227325527084006025542746313547512695257222470434492149292452942848031399460233257906808052273943960576478385469441356693578905 5

The sign is: 16854110369586101322810686762896097322783679004007977546689588679634927816846938133771591678267653272078143746716724809138380954446257106090628051407680 4

Encrypted sign is: 60163368839171955729748399034662415406421732326647231051103127768456572397409487311521494700531842685371850821144147456087466195268323698559225017928362 0

Decrypted message: 26

Verifying sign; the sign is: 16854110369586101322810686762896097322783679004007977546689588679634927816846938133771591678267653272078143746716724809138380954446257106090628051407680 4

message exchanging successful, sign verified

Process finished with exit code 0

Ключі, що не підійшли, було записано в окремий файл, що буде надано разом із протоколом.

Висновок: В ході виконання лабораторної роботи було створено тест перевірки числа на простоту, алгоритми генерації ключів для асиметричної криптосистеми типу RSA, та її складові функції.