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

ЛАБОРАТОРНА РОБОТА №4
«Криптоаналіз афінної біграмної підстановки»

Виконали:
студенти групи ФБ-04
Дмитренко Даніїл та Сербіненко Олексій
Перевірив:
Чорний О.

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 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 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 $0 < k < n$.

Хід роботи

Ми реалізували функцію пошуку випадкового числа, та завдяки тесту Міллера-Рабіна шукали прості числа. З отриманих даних генерували p, та q для A та B. Також реалізували, функції шифрування та дешифрування, підпису та верифікації повідомлення.

Результат роботи

Our message is: 123456789

Public e, e1 = 65537

Public n =
24413691458731805985021634885163108379948715443138901789300342465109125545688137601913764225848404652408069915594988909943584595599795130442194241556436243

Public n1 =
29690558901277527990921889374258751810100243449215929405376749540864274802682947437081527489703992057572410762757335446534262491971483727796689545805270311

Encrypted A =
74069952089962635900891586768462746323518732387774233258741366303530687061799276083527227096659679179491045692579014155411203588327110618048361304290994 47

Encrypted B =
27467307660170016555526463565735327850697425180855968653342214568245713649438842720773554041282428561607079277631770168072476221348814140695529302960366998

Decrypted A = 123456789

Decrypted B = 123456789

Sign A =
17848126826404933269722034590852916708012047212461271228698171738601501393953959793558204760742493176578081679374283505055647756813493485524645917133729 98

Sign B =
24058853177569426868666844441153963279316202423922543567693892551448906667581537863391518763320919818434329995815507280768475566491646785316884154867046258

Verify sign A = True

Verify sign B = True

Is key okay? – True

Висновки

Ми ознайомилися з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA, ознайомилися з системою захисту інформації на основі криптосхеми RSA, вивчили протоколу розсилання ключів.