

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 $1 < p, q$ довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб $pq \leq p_1q_1$; p і q – прості числа для побудови ключів абонента А, $1 < p < q_1$ – абонента В.
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 А і В – тобто, створити та зберегти для подальшого використання відкриті ключі (e, n) , (d, n) і секретні d і d_1 .
4. Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів А і В. 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 А і В, перевірити правильність розшифрування. Скласти для А і В 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:

19316344555679821407103556583099837811988287079534965412589478220764970271249297364
628747880097743851572410299266041605512907875111994209022177532831211801

Encrypted A:

81710427392601253283456786472331670577939429476120371200469904924262779935223020429
10015943421377847504923165095186679477849724459601130337549703984432228

Decrypted A: 550

Sign A:

19147745344846407913628866979284650607460323483969132327716454447772199520681395305
572188369895810817661216776155263195127091211556331393253996918690701226

Verify A: True

-----B-----

Public exponent1: 100001

Public n1:

47278581270682572701653393789430943754651470889310194099838564059014530593019450760
164895304858875414717208978757434440648531734232865905061347015265672121

Encrypted B:

11160173652529781982779328244919599368482737425714278398539562677995585191264556740
323577649673969347096765546112713609493661155898362496891696407190649127

Decrypted B: 550

Sign B:

36890579487991270713232001117826387065691877316554475848869821402194396300306169675
702125249912728409098072131758269965132688834318918018617616995831783309

Verify B: True

k =

10081693015949278419446447970398502303321569772782986655941683359720083169845418720
742078871085790813693049770487083863437441146671662669978922852786722790

Key received

Encryption:

81710427392601253283456786472331670577939429476120371200469904924262779935223020429
10015943421377847504923165095186679477849724459601130337549703984432228

Decoding: 550

Checking the text: True

```
Message : 550
-----A-----
Public exponent: 100001
Public n: 1931634455679821407103556503099837811988287079534965412589478220764970271249297364628747880097743851572418299266041605512907875111994209022177532831211801
Encrypted A: 81710427392601253283456786472331670577939429476120837120046990492426277993522302042910015943421377847504923165095186679477849724459601130337549703984432228
Decrypted A: 550
Sign A: 19147745344846407913628066979284650607460323483969132327716454447772199520601395305572108369095010817661216776155263195127091211556331393253996918690701226
Verify A: True
-----B-----
Public exponent1: 100001
Public n1: 47278501270682572701653393789430943754651470889310194099830564059014530593019450760164095304058075414717288978757434440640531734232865905061347015265672121
Encrypted B: 11160173652529781982779328244919599368482737425714278398539562677995585191264556740323577649673969347096765546112713609493661155898362496891696407190649127
Decrypted B: 550
Sign B: 36890579487991270713232001117826387065691877316554475848069821402194396300306169675702125249912728409098072131758269965132688034318918018617616995831783309
Verify B: True

k = 10001693015949278419446447970398502303321569772782986655941683359720083169045418720742078871085790813693049770487083863437441146671662669978922852786722790

Key received

Encryption: 81710427392601253283456786472331670577939429476120837120046990492426277993522302042910015943421377847504923165095186679477849724459601130337549703984432228
Decoding: 550
Checking the text: 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