



МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Кафедра інформаційної безпеки

Комп'ютерний практикум №4

з дисципліни

Криптографія

на тему:

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

Виконав:

Студент групи ФБ-91

Шостак А. А.

Київ 2021

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 $1 \leq p, q$ 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, $1 \leq p$ і q – 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) , $(,)$ і 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 $0 < k < n$.

Хід виконання завдань:

Кандидати, які не пройшли тести перевірки простоти:

None:

111195795415741554232192485483835742406958017020637040296092679642612256929975

None:

47700028139119835508307191612331302791651740661471219939320764719007471792155

None:

47331971427486736834019214822597130129576011589452876709384339279263179543840

None:

113355362069685854577213327841079652835808711547292331347019082261991861544396

...

Кандидати, які пройшли перевірку:

A secret keys:

$p =$

100421553524124201309151446681403970722706635198085189800766725781977924977277;

$q =$

99964587581160914639141589556976716641880796791801015393708013403472688892389;

$d =$

64022160073197278028712629072178268993092971696925443468996089354663916567228598
90185658308210988954426202854454185360972361763449577493811290849042144597.

B secret keys:

$p1 =$

66567152544001899521033952116500562218200712981123225951343008589052831775739;

$q1 =$

37686694880984064623140029776745397477759823495924806919048297084666312185649;

$d1 =$

87455533845292158288264114590250714141593483591196765184016267858913401022818825
1940299279443057025579320150332398279649753090639103552271171093878712383.

Також $f =$

10038599182298552216137716915495278851272985824204397559709209668390554322253116
732769072046540159074108472142144296733489770455946556346959603152509375088.

A open keys:

$n =$

10038599182298552216137716915495278851272985824204397559709209668390554322253317
118910177331656107367144710522831661320921760342151750821698788603123244753;

$e =$

93264091264269443607467578730592966501269237474526906850573267164731103010242552
95142257253869634976425442489627566755976671003094869417679397667631094957.

B open keys:

$n1 =$

25086959670217217412713427992078504620669677561057858882726762140269509335639145
31209799517017270310763812621048169886031826727014718907640301370902169611;

$e1 =$

11851376288545597828623167116997834143554941768128806860688766648775960356032867
92036636393576135639827484536787187088522575311531342723961059216726698335.

Open text M:

11489999654317105460402667836062313584141247180516686752839550785892526037816466
33216342251740352591621213010650542187716032264607084774067981650926667728.

Encrypted text C:

55299032538167428661884881674729085029853149231774592781138415550739477879511283
20064869577069885496728812003869976699024930381144571761512471122982310956.

Decrypted text M_decr:

11489999654317105460402667836062313584141247180516686752839550785892526037816466
33216342251740352591621213010650542187716032264607084774067981650926667728.

Sign:

39492184578665395464822332827437218072132728439215295356820071634798395270910097
52688440897584691358714663747016161090357522224345739375450040286559082175.

A generated secret value k:

88130154917063643898662060245211048573140198798219731168243801012739110266423831
1064226417377419326409741371236633870188915112728742075098781671670444238

A created a message with parameters:

k_1:

15986994292119635896870711816073696374318578468915264795324491121411931213338160
26742945849285874512179249104532957404693672277739923071099793139878073595

S_1:

13014659854281931610376372497318161006964979352360215205610106983912806433344207
23135079727565434950257465947991607015463618430991197513954358796348973904

B received parameters:

k:

88130154917063643898662060245211048573140198798219731168243801012739110266423831
1064226417377419326409741371236633870188915112728742075098781671670444238.

S:

53001726638114841360015343713110887028627703865240138947564988683379969716789097
6863859895525965392874415929523197400164655979609602259172293914264449289.

А також перевірка достовірності тексту на сайті <https://www.dcode.fr/rsa-cipher>:

The screenshot displays the 'RSA DECODER' interface on the dCode website. On the left, under 'Results', it shows a successful decryption using parameters C, D, and N, with the decrypted text: 1148999965431710546040266783606231358414124718051668675283955078589252603781646633216342251740352591621213010650542187716032264607084774067981650926667728. Below the text is an advertisement for TCH cigarettes. On the right, the 'RSA DECODER' section contains input fields for various cryptographic parameters: VALUE OF THE CIPHER MESSAGE (INTEGER) C=, PUBLIC KEY E (USUALLY E=65537) E=, PUBLIC KEY VALUE (INTEGER) N=, PRIVATE KEY VALUE (INTEGER) D=, FACTOR 1 (PRIME NUMBER) P=, FACTOR 2 (PRIME NUMBER) Q=, INTERMEDIATE VALUE PHI (INTEGER) Φ=, and a DISPLAY section with radio buttons for output format (Plaintext as character string, Computed values, Plaintext as integer number, or Plaintext as hexadecimal format). A 'CALCULATE/DECRYPT' button is at the bottom right.

Створення ключа, сигнатури, її відправка прийняття та перевірка отриманих даних локально:

```
Before sending:
modulus = 3668924544228771259036029966615089895764095629575513138383040686022496044345493486460416832539542005695899204
8118418457084728512601231621208038663848223297463
public exponent = 65537
k: 2845122624493474161404692023938310902622787205999805091932486128103394559160
S: 54052042405914121921099305500704263132449160542442320664948840807491673638352633662649322402813976294522910036183051
03338930396388980452355964518697796883609
After sending:
k 1: 540520424059141219210993055007042631324491605424423206649488408074916736383526336626493224028139762945229100361830
5103338930396388980452355964518697796883609
S 1: 329582569742444891960947721083885509288017143156541998810035611870342950801363646954704353492723254942345100671812
27304219719201282794021065456696701956606722
After receiving:
k: 2845122624493474161404692023938310902622787205999805091932486128103394559160
Received S: 54052042405914121921099305500704263132449160542442320664948840807491673638352633662649322402813976294522910
03618305103338930396388980452355964518697796883609
Key verified!
```

Створення ключа, отримання повідомлення на сайті <https://asym-crypt-study.herokuapp.com/>:

Server
Key

Encryption

Decryption

Signature

Verification

Send
Key

Receive
Key

Get server key

✖ Clear

Key size

256

Get key

Modulus

ABCFCD94F22E422DAA7015F4F8F48C1901F126F74B03C39C95

Public
exponent

10001

Server
Key

Encryption

Decryption

Signature

Verification

Send
Key

Receive
Key

Receive key

✖ Clear

Key

5163F0028950B5605CBEA6F8521A2D0D1DE4671981CBEFE90E

Signature

64A484761DF27C8E2DD23A01741C1718C70AFE2A8B18B55BDf

Modulus

ABCFCD94F22E422DAA7015F4F8F48C1901F126F74B03C39C9f

Public
exponent

10001

Receive

Key

064A484761DF27C8E2DD23A01741C1718C70AFE2A8B18B55Bf

Verification

true

✓

Висновки:

Під час виконання даного комп'ютерного практикуму я ознайомився з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; отримав практичні навички з захисту інформації на основі криптосхеми RSA, організував використання цієї системи засекреченого зв'язку й електронного підпису, вивчив протокол розсилання ключів.