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:
n is: 17b22191d53874ba7ad37f2808a2a1d8da4dfb317b4ffbd2b8fbb44b65ee989830f995841a05b101016ed5f991d645cc226c32925a0481e4c4e9369baf31dfe89c
e is: 10001
fi(n) is: 17b22191d53874ba7ad37f2808a2a1d8da4dfb317b4ffbd2b8fbb44b65ee989830f995841a05b101016ed5f991d645cc226c32925a0481e4c4e9369baf31df
d is: b44ec322ef56a8e455bd05d79bddebb0317931f354614bd62bc52428977af9ab1b0012b9234a2b997b4de8e928d1524c03a34d2e57cbf49cd5c1c5c6e358a07440
p is: 136848214abcb172eeb9418b07b35e865f0fd083fe8490e7bf9ef6cabfa4a1312af9977d271b547edfabceed1c420caca5117d071c6f0de53e665dce34336791f
q is: 1389198f3d92e8057063fe237b6bd6cbe78161560a02b40e9e49723234847464941b68040e9aa19959bcc5b3c7304af04c8896fbb163a399dd8104b918258886d
```

Keys of Alice:

n is:
17b22191d53874ba7ad37f2808a2a1d8da4dfb317b4ffbd2b8fbb44b65ee989830f995841a05b101016ed5f991d645cc226c329
25a0481e4c4e9369baf31dfe89cca401b5b24844987a40c6bafe00174b541d523464f12e1d1a967c8c00b9f207ec371603571fb7
285310d383dc6c57dbda08a5540d3b33554e3f3c42ef320a33

e is: 10001

fi(n) is:
17b22191d53874ba7ad37f2808a2a1d8da4dfb317b4ffbd2b8fbb44b65ee989830f995841a05b101016ed5f991d645cc226c329
25a0481e4c4e9369baf31dfe875d8de6ad2d4ead12886ccbd2cc0cc226eb0a3493dc7cdeb73c0fecbcbe2898abfae71deffbc055a
4bc878975a546de0cc067652730101b638fc913ce29a308a8

d is:
b44ec322ef56a8e455bd05d79bddebb0317931f354614bd62bc52428977af9ab1b0012b9234a2b997b4de8e928d1524c03a34d
2e57cbf49cd5c1c5c6e358a0744049fc2ae0214aae45b3783c084f0906f941214030dc320f72c12015a35a2b419f51cb2b16cfa1e
b76bdfee1b478bfa1d0df5b7c079fa166f83ae8f0635ce601

pis:
136848214abcb172eeb9418b07b35e865f0fd083fe8490e7bf9ef6cabfa4a1312af9977d271b547edfabceed1c420caca5117d07
1c6f0de53e665dce34336791f

q is:
1389198f3d92e8057063fe237b6bd6cbe78161560a02b40e9e49723234847464941b68040e9aa19959bcc5b3c7304af04c8896f
bb163a399dd8104b918258886d

```
Keys of Bob:
n is: 2211eb9214392a4b0fd3a1468bfee5be9a9c792d0e3fc0c2238aa76f34a3ce9b7ccbef769abea44f427472e9774c9ad821f17c1a24de1a42bb719dbf45d5890f53
e is: 10001
fi(n) is: 2211eb9214392a4b0fd3a1468bfee5be9a9c792d0e3fc0c2238aa76f34a3ce9b7ccbef769abea44f427472e9774c9ad821f17c1a24de1a42bb719dbf45d589
d is: 757ad59321a8fced57e5a8785ac4d7f227b0e72735bb20324fdc0cfea9c52ee979aabe4f7bc9654ae91c222269fcfcf3ca17a8977e095f043cd5c9dd84308e2c64
p is: 11cda112d60137dbb2e09a20f2aee67d10b4011a580bb89a46c27be9e68c04b198fd6f5b04be0f1f97afbe2e1eae5b43b8354477d98466d534fdc07210e663da3
q is: 1e9e807d08e3ff9ff99ae09fbf54b6a0789742fc5f1a5fedb48b10256252b19cee4c38a4ecf63ca74ee4b8865ce627062ea58daa035627b48a6c58bc6e7114ef1
open text Alice wants to send: 10124704
Encrypted text with Bob's public key: 17406262287137051790523495925073053116493590863519377831722354104618115423689489368820136707536531
The text Bob received after decryption with his private key: 10124704
```

Keys of Bob:

n is:
2211eb9214392a4b0fd3a1468bfee5be9a9c792d0e3fc0c2238aa76f34a3ce9b7ccbef769abea44f427472e9774c9ad821f17c1a24de1a42bb719dbf45d5890f53b4082ce4c49425e3668ff6d0e3b53f6423ad00d65b12b2a6fc8569a44c9cd2594f7c67326e798829086c5fe4b62ba5d8cf50bafeacba3ae4c792bfbe1dab073

e is: 10001

fi(n) is:
2211eb9214392a4b0fd3a1468bfee5be9a9c792d0e3fc0c2238aa76f34a3ce9b7ccbef769abea44f427472e9774c9ad821f17c1a24de1a42bb719dbf45d5890f2347e69d05df5caa36eb15361ee01821dad868ea1f34fa2aabaef95a5b6de683d205d46740ba2dc14273f5ab6921a95bf1f47e9921d22bb1255d79913ec6323e0

d is:
757ad59321a8fced57e5a8785ac4d7f227b0e72735bb20324fdc0cfea9c52ee979aabe4f7bc9654ae91c222269fcfcf3ca17a8977e095f043cd5c9dd84308e2c64bbc9ba2a45ca82265cf90765efd546e81ad3aa7b71723131506903e9c14cddbf97da932fdecd675a7e00e2cfe202dbc9b609d47f8af89e1ef9f3f7a33e4a81

p is:
11cda112d60137dbb2e09a20f2aee67d10b4011a580bb89a46c27be9e68c04b198fd6f5b04be0f1f97afbe2e1eae5b43b8354477d98466d534fdc07210e663da3

q is:
1e9e807d08e3ff9ff99ae09fbf54b6a0789742fc5f1a5fedb48b10256252b19cee4c38a4ecf63ca74ee4b8865ce627062ea58daa035627b48a6c58bc6e7114ef1

open text Alice wants to send: 10124704

Encrypted text with Bob's public key:
17406262287137051790523495925073053116493590863519377831722354104618115423689489368820136707536531326914818340903518186076131785528598668883062756784014478708910506708610102117722249390739594219903741020284262640714192352009924450029905212325092993198121717408580686987252685532988403715597182808140340729331 2

The text Bob received after decryption with his private key: 10124704

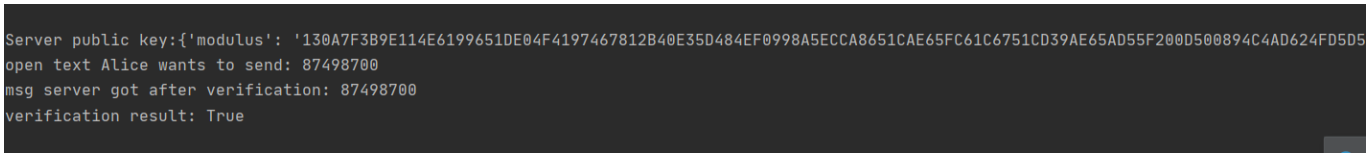
Робота з тестовим середовищем

```
open text Alice wants to send: 17324815
encrypted msg is:1facf34d8ca1ea9eaaed35a61543e8dbda16929c747fd516c27ee39abd9d08c8896f99875d8b972f0e95ab3efd01741c0e6cad199730a0dac3fa6194e
signature is: 16aaad3fa4f7f6ae36e85ba7b2fd242ba58508cc30c128a8919b434528f3fbc6cd40b4e8d4b16709d5724b1f61868efb7f0fec95cee4127c55d3412a4009
Signature is valid
encrypted msg is: 17324815
```

open text Alice wants to send: 17324815

encrypted msg
is:1facf34d8ca1ea9eaaed35a61543e8dbda16929c747fd516c27ee39abd9d08c8896f99875d8b972f0e95ab3efd01741c0e6cad

199730a0dac3fa6194ec8e55db1942feec1148b39ff2943ed29442b9bc2ee8218cb661fefd1321a92e9c5c0492b6e424894564c75cbcd90af0a435d3b4eaeef2526b2b5d4be15cfd06d92755e41
signature is:
16aaad3fa4f7f6ae36e85ba7b2fd242ba58508cc30c128a8919b434528f3fbc6cd40b4e8d4b16709d5724b1f61868efb7f0fec95cee4127c55d3412a400904c85a04bd58608846128c962e6284d5d0512d27c9c773005d1013c4c6d2a1072bb00c38fad6a2cbcfc74030dfaf3de3f02de6540948018843ebf4d8266e28e04587c
Signature is valid
encrypted msg is: 17324815



Server public key:{'modulus':
'130A7F3B9E114E6199651DE04F4197467812B40E35D484EF0998A5ECCA8651CAE65FC61C6751CD39AE65AD55F200D500894C4AD624FD5D59635222BB5A55B63C077DA256542D3E2DBBB8E14A98DA6E952B79CBB76023A4D9A9687B28787A3F96F9C50563C85D7C271AC8333870F683E769AE0C7A6C1076C9ED7B1F23390313BDD', 'publicExponent': '10001'}
open text Alice wants to send: 87498700
msg server got after verification: 87498700
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.