Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №4

з дисципліни

«Криптографія»

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

Виконали:

студенти 3 курсу ФТІ

групиФБ-82

Руднік

Анатолій та Сумовська Юлія

Перевірили:

ЧорнийО.

Савчук

М.М.

Завадська Л.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.

Постановка задачі:

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 p 1 , q 1 довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб pq <= p 1 q 1 ; p і q – прості числа для побудови ключів абонента А, p 1 і 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) , (e 1 , n 1) та 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 М і знайти криптограму для абонентів А и 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 $0 < k < n$.

Кожна з наведених операцій повинна бути реалізована у вигляді окремої процедури, інтерфейс якої повинен приймати лише ті дані, які необхідні для її роботи; наприклад, функція Encrypt(), яка шифрує повідомлення для абонента, повинна приймати на вхід повідомлення та відкритий ключ адресата (і тільки його), повертаючи в якості результату шифротекст. Відповідно, програмний код повинен містити сім високорівневих процедур: GenerateKeyPair(), Encrypt(), Decrypt(), Sign(), Verify(), SendKey(), ReceiveKey().

Хід роботи, опис труднощів, що виникали, та шляхів їх розв'язання;

програма має такі функції:

gcd - пошук найбільшого загального дільника

miller_rabin - тест Міллера рабина з попередніми розподілом

choose_random_prime - функція для вибору випадкового простого числа з певного інтервалу

findModInverse - пошук зворотного по модулю

good_random - генерує хороші прості числа для кращої стійкості

gen_p_q - генерує q, p, q1, p1 такі що q * p <= q1 * p1

GenerateKeyPair - генерує відкритий і закритий ключ користувачам A і B

Encrypt — шифрування

Decrypt — розшифрування

Sign - цифровий підпис

Verify - перевірка цифрового підпису

SendKey - відправити ключ з підтвердженням справжності

ReceiveKey - отримати ключ і перевірити справжність відправника

труднощі виникли на моменті зашифрування, коли потрібно перетворювати рядок в число, проблема була вирішена взяттям значення ascii кожного символу і конкатенацією цих значень.

значення вибраних чисел p, q, p 1 , q 1 із зазначенням кандидатів, що не пройшли тест перевірки простоти, і параметрів криптосистеми RSA для абонентів A і B

p користувача A:
0x2c33f23718c8f8f1908baef04604d800a29c0b5dc30ce6cd93262f1fc7b7b48ca0c32ec112e7e3950a30efdd49b20cebdf92f07a31f07d8fb659df050c0b9666f6103

q користувача A:
0x3a5f81de55a38151223d908d3a089537160205aa4a2baa0948c1da8e7606994ab33dd77be3b403f5154c9063009c62bcfdf412bc6244aaf982800ddd380c253504d13

n користувача A:
0xa144293eef3051393d455d68e72042da5762a1a13a871ad3915708adc04384fdcc4890a13f36c52d1c49d7983369db1620f1e0dcdd16980980051202eb5c3c0654d941d96704cf9c060c2ffd4d1204d2895fea151747702660ced8c1ed22ca532a4bba81e6881633eca5e8d49c192155ce253bfa1b2932174203b1ffd537e351a64621a39

e користувача A: 0x10001

d користувача A:
0x8b1d789340d5268c412dd059c9600c2bbc4c6a6fe2446040f854e338b7edc71105a3e7982227064b56e6e063fc9e298fda55ae0bf9711f30bace24ee7dc8704369f834980fecaea2cc6cc16420fee8d5fce7335ca0f817485d2a6966310ffea1ffd72e01a8fb4542e7cd438fb0870b5fe1035a9ae0a2bbce2afa7afdd781dd3ca211ac23d

p користувача B:
0x272623c159e49e78e7a4f4ada10664abad563e4ab72b1a55b70d2e10e1daaed53c2e3405c319efca0d6ee99ee891ceed1f216ce844fc69674ca1af891392f77e35b67

q користувача B:
0x4344e38e6fa674ef2892f578e20bd235ce76194872e7a66dca74b640bdfa942eb29af8f8e91b510d7abe0183c541b37627f438306065d6dac96ed691b204db3073f1b

n користувача B:
0xa498449b3d1fc473270a8667bc9dc318e804e625bdd51d1ae26fb6602413d9b8475fcbf3a1f449ecd9a3a5a14a2675e0e6d4620e45bfc64bf89b60c685dd205da233b7729f65c9235b219105288da3124e4a7c36faf95fdc0448c6f22353dd4a3bc0dd73f9d02c492a47377d232548a708e80b4062f72bab7e2d84b0f1330549e1d49fcdd

e користувача B: 0x10001

d користувача B:
0xa388f2a9772ac80a4da10477af8ce41a2bf52a2217a1d6002f3925701288728ac8c7a2a197a95b85eb9bcfe618016c9f9f59276fdcd73513a897e0a87eba7cfceb3aa36f47299c41560a36b2362d4a3b2d0753b5a051de60f1284a1cc34ddb780be7d40e79e031ac1704b3a0357649149f93ec4037541002577642fd3a4ae68668cca3eb5

чисельні значення прикладів ВТ, ШТ, цифрового підпису для A і B

чисельне значення ВТ 0x61747461636b206174206461776e

чисельне значення ШТ

0x138093801a459194ab70982a7979a8c4b20ccc5a2c4d905ffbb352a703171330d8d1b290af67b8d5c5aed405
33e5ee0933a61155291952cf2db23103f7cb09b294115f0532b606543050c16e1d3b8dc7994bc88fca9ddcd468
04e34b5fb3fea46b2db7f82ce80fdfe7aff899bc0bd86e30275de4a98c19d3932066b562d20ad82333049b1

підпис користувача A (attack at dawn,

0x2805625a752e4f367b0acddbd14d752919a331d7cc8a8f2a934a9cf1770919fcc557e007ca60a2cc80ebb0bb2
a6fb0f91bf71ea57c1a62864f37e3150aac524115295206ecc5f1d1183721b58023dd7cd08e29fb36ffa7c1f557c9
cc3a13645754793f320967b763e77b27da314009d99c4301e7f9a41767729cdda2a326df847f51137c0)

підпис користувача B (meet near the fountain,

0x35ceb38cf257c71daad294cea518fc6dd43d7d0927d8b548ea3a191aaeeb768e8e4bd510086baa97d3f249678
92ef52463afb0d83abb960e5d68cd37370350e7cc24ea678dcd1045d7f1ab17d32c990474e7cec62a8ae0ccff79d
f041f463b9a96bfabb326beb6a1918ff6fcc48a947fd8fb8746d3378110edbdee8fbc0eac441ba20752d)

опис кроків протоколу конфіденційного розсилання ключів з підтвердженням справжності, чисельні значення характеристик на кожному кроці

процедура SendKey приймає 2 параметри: одержувач і відправник,

спершу бере n, e відправника

n=0xa144293eef3051393d455d68e72042da5762a1a13a871ad3915708adc04384fdcc4890a13f36c52d1c49d7
983369db1620f1e0dcdd16980980051202eb5c3c0654d941d96704cf9c060c2ffd4d1204d2895fea1517477026
60ced8c1ed22ca532a4bba81e6881633eca5e8d49c192155ce253bfa1b2932174203b1ffd537e351a64621a39

e=0x10001

,потім генерує k

k=0x164d039f14ae0283a625d16ceed70a88a86bf9f4fddac8db5f02cb07b3a43589e2dae2ef66ab79f683787197
54d0c0e8ded4bf685f673f075afdd836408912f41f618c76df0d4aaae83f389941c4fd4863179092b5a6a97279d8
f00845fcf9806f922fe94cd5252a2f4eaddc180ce2180203d93fbab7ceab45e786436d1909209548e9cb2

,потім бере n,e одержувача и зберігає в змінні n1,e1,

n1=0xa498449b3d1fc473270a8667bc9dc318e804e625bdd51d1ae26fb6602413d9b8475fcbf3a1f449ecd9a3a5
a14a2675e0e6d4620e45bfc64bf89b60c685dd205da233b7729f65c9235b219105288da3124e4a7c36faf95fdc0
448c6f22353dd4a3bc0dd73f9d02c492a47377d232548a708e80b4062f72bab7e2d84b0f1330549e1d49fcdd

e1=0x10001

підносить k^e1 mod n1 і зберігає в змінну k1

k1=0x980c0120f11eae0f2dbf13967f442ab165fa0b5895beef4f21941c0bc9002cd06c502f1acae7f58b77a3bfd0
6ef61ada51d60b62e9f2fe663d2dbe3f1858be646d7401a0e15fe91eddf3d5ba9bcb2eda44fced38d4a7a3da9663
4ec471be8e91d2fdb9ee35583b6616c55cd1949f088682f95796fc9cefcb588adaa5009c3e7006da1eb8f

,бере закрытий ключ відправника и зберігає в змінну d

d=0x8b1d789340d5268c412dd059c9600c2bbc4c6a6fe2446040f854e338b7edc71105a3e7982227064b56e6e0
63fc9e298fda55ae0bf9711f30bace24ee7dc8704369f834980fecaea2cc6cc16420fee8d5fce7335ca0f817485d2a
6966310ffea1ffd72e01a8fb4542e7cd438fb0870b5fe1035a9ae0a2bbce2afa7afdd781dd3ca211ac23d

,підносить k^d mod n і зберігає в змінну s

s=0x1579ad0a4739bded119135a85d3ac8e1f22d5ceb67b33e4fc279ff8d5060962f8551c89ebf141d75aaad993a
a62aeeb0cd561eb7eec1fae19bd4ff967651c36281d58f00b06cd5d0ad03b4737bbf2ec68035cef00a1b7e40ec3d
cc5c1453855c892cddf44685351dc9d185fcce57b172a6b42716bb21b17e697b2b0d6bad228ee0df2dcae

,підносить s^e1 mod n1 і зберігає в змінну s1

s1=0x5a13ee178411611e7dc515cc5f353fdedd99c3aa16383e582b97d9637c14f8c4d46e5f307dc8fe5dbd6db5
e1c0b93fbf7b70953bbfe9f1243457d8fa6375ed14f7d40f47662fbb141e5b3aba5777e7a4bc20e1a98d6b3dd7e0
84996874ceeb7cd8cf5b9a96e4016f3bba711c7f9a5826962a26b97b83c21760dd78c646a479dc8b483c6e8

,відправляє k1 и s1 одержувачу

процедура ReceiveKey також приймає два параметри одержувач і відправник,

бере k1,s1

k1=0x980c0120f11eae0f2dbf13967f442ab165fa0b5895beef4f21941c0bc9002cd06c502f1acae7f58b77a3bfd0
6ef61ada51d60b62e9f2fe663d2dbe3f1858be646d7401a0e15fe91eddf3d5ba9bcb2eda44fced38d4a7a3da9663
4ec471be8e91d2fdb9ee35583b6616c55cd1949f088682f95796fc9cefcb588adaa5009c3e7006da1eb8f

s1=0x5a13ee178411611e7dc515cc5f353fdedd99c3aa16383e582b97d9637c14f8c4d46e5f307dc8fe5dbd6db5
e1c0b93fbf7b70953bbfe9f1243457d8fa6375ed14f7d40f47662fbb141e5b3aba5777e7a4bc20e1a98d6b3dd7e0
84996874ceeb7cd8cf5b9a96e4016f3bba711c7f9a5826962a26b97b83c21760dd78c646a479dc8b483c6e8

,котрі відправив відправник,бере n одержувача та зберігає в змінну n1

n1=0xa498449b3d1fc473270a8667bc9dc318e804e625bdd51d1ae26fb6602413d9b8475fcbf3a1f449ecd9a3a5
a14a2675e0e6d4620e45bfc64bf89b60c685dd205da233b7729f65c9235b219105288da3124e4a7c36faf95fdc0
448c6f22353dd4a3bc0dd73f9d02c492a47377d232548a708e80b4062f72bab7e2d84b0f1330549e1d49fcdd

,бере закрытий ключ одержувач та зберігає у змінну d1

d1=0xa388f2a9772ac80a4da10477af8ce41a2bf52a2217a1d6002f3925701288728ac8c7a2a197a95b85eb9bcf
e618016c9f9f59276fdcd73513a897e0a87eba7cfceb3aa36f47299c41560a36b2362d4a3b2d0753b5a051de60f
1284a1cc34ddb780be7d40e79e031ac1704b3a0357649149f93ec4037541002577642fd3a4ae68668cca3eb5

,бере відкритий ключ відправника і зберігає в змінні n и e

n=0xa144293eef3051393d455d68e72042da5762a1a13a871ad3915708adc04384fdcc4890a13f36c52d1c49d7
983369db1620f1e0dcdd16980980051202eb5c3c0654d941d96704cf9c060c2ffd4d1204d2895fea1517477026
60ced8c1ed22ca532a4bba81e6881633eca5e8d49c192155ce253bfa1b2932174203b1ffd537e351a64621a39

e=0x10001

,підносить k1^d1 mod n1 і зберігає в змінну k

k=0x164d039f14ae0283a625d16ceed70a88a86bf9f4fddac8db5f02cb07b3a43589e2dae2ef66ab79f683787197
54d0c0e8ded4bf685f673f075afdd836408912f41f618c76df0d4aaae83f389941c4fd4863179092b5a6a97279d8
f00845fcf9806f922fe94cd5252a2f4eaddc180ce2180203d93fbab7ceab45e786436d1909209548e9cb2

підносить s1^d1 mod n1 і зберігає в змінну s,

s=0x1579ad0a4739bded119135a85d3ac8e1f22d5ceb67b33e4fc279ff8d5060962f8551c89ebf141d75aaad993a
a62aeeb0cd561eb7eec1fae19bd4ff967651c36281d58f00b06cd5d0ad03b4737bbf2ec68035cef00a1b7e40ec3d
cc5c1453855c892cddf44685351dc9d185fcce57b172a6b42716bb21b17e697b2b0d6bad228ee0df2dcae

якщо к = s^e mod n

s^emodn=0x164d039f14ae0283a625d16ceed70a88a86bf9f4fddac8db5f02cb07b3a43589e2dae2ef66ab79f68
378719754d0c0e8ded4bf685f673f075afdd836408912f41f618c76df0d4aaae83f389941c4fd4863179092b5a6a
97279d8f00845fcf9806f922fe94cd5252a2f4eaddc180ce2180203d93fbab7ceab45e786436d1909209548e9cb2

то автентифікація пройшла успішно

висновки:

в цій лр ми реалізували криптосистему RSA та алгоритм електорнного підпису,

цей практикум дуже важливий,тому що алгоритм rsa використовується у таких сучасних протоколах як:PGP,TLS/SLL,IPSEC та ін.і треба розуміти як це працює