НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“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:

ФБ-32 Рибчук Нікіта

ФБ-32 Луценко Євгеній

КПІ 2025

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 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) , (,) 1 n1 e 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 $0 < k < n$. 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

Генерація ключів для A і B:

```
>>> 1. Генерація ключів...
[Абонент A]
Public Key (e) (Dec): 65537
Public Key (e) (Hex): 10001
--------------------
Modulus (n) (Dec): 779720952101039452750960222691716935783889654710469049597078090715149322157839228522441647673998287648044999784103646321642580335869474134552298793289396 1
Modulus (n) (Hex): 94DFF75B75605AF7C04F9C4E7878058D36A8917900A72BCE3SCD8E565632FABADF20AE048C55EAF301D7D936E1CBDB27D2539451FAADCEFE221F115FD5ACC309
--------------------
Private Key (d) (Dec): 569053407067652120559049505643298000191394817962398868459469384910899065853008049270507842966090774190021283540662923703496076934443363161440370388598189
Private Key (d) (Hex): ADD79A10C24F71EA3C36BA5F276D7F8E167560E9DB495A6DC185F6DFFA682D3E9887A7C0340074BBE390F57E291E3B4E3C7C819E1E11CC65C690460391 77DAD
--------------------
[Абонент B]
Public Key (e1) (Dec): 65537
Public Key (e1) (Hex): 10001
--------------------
Modulus (n1) (Dec): 869501337396403340109915216803962620872599779117830208928112334605265534708633238744304743045845792171408305847507929755766441030354745073063064691808399 7
Modulus (n1) (Hex): A604570E2508001D6AEC4D2D2211C2EC0040AC732B360E9566761490BC465A5CCC9270A2AF1CAB991477E192CEF5BA544DDCBC2760957EB3B247D07FD838F59D
--------------------
Private Key (d1) (Dec): 959228324362725811220331571096121236692081786322521996818629503821668647625506720541463493928700599768293047412277852234705152821091630925089876712115 3
Private Key (d1) (Hex): 1D4DC6D00859D8BB58943D4B2A28586FAFDBBA94D99BC1173DEEB9D2926918A9DFDEEB13BC5850E2EABC12B8A85A1F722BBAED2052E84FE56E133B492657901
--------------------
```

Шифрування та дешифрування повідомлення:

```
>>> 2. Шифрування тексту (A -> B)
Вхідний текст: Hello IPT
Текст як число (Dec): 133547390882694 2034004
Текст як число (Hex): 48656C6C6F20495054
--------------------
Зашифроване повідомлення (Ciphertext) (Dec): 422015857637289010255965985472952685779125420176130275505223120045711162960724056975446676663927612968130038886979446041118995504317982497317014353256643 9
Зашифроване повідомлення (Ciphertext) (Hex): 5093B49728E24E14E183315E499BC774E62BE86A2165F8EBD9235613ACAC026872C73E85743D289CEDA764522C74C030D3EC9DE9F7F9A74741FF24931E0647A7
--------------------
Розшифроване число (Dec): 133547390882694 2034004
Розшифроване число (Hex): 48656C6C6F20495054
--------------------
Результат: 'Hello IPT'
```

Цифровий підпис та його перевірка:

```
>>> 3. Цифровий підпис повідомлення
Цифровий підпис S (Dec): 110104646847886624808546864496594460753370478643786866968158124029477459125964681989177144612020728465390436155564486576901900417944827299292744755840115 5
Цифровий підпис S (Hex): 1505CD9372CB4C374866C485423B8FFF6DD011F843B84F1DC08B077BDC202A1A3770955AA4AA6355D8E9187590782536 55F0E4C79D918E0923CC37001CB42C83
--------------------
Перевірка підпису: УСПІХ
```

Протокол обміну сеансовим ключем k:

```
>>> 4. Протокол обміну сеансовим ключем k
Згенерований ключ k (Dec): 201082509578672194218977850544758725332997983347071354368451222773832904626893054234062405594228888018777330870482905226590806723913718801121455148526869 5
Згенерований ключ k (Hex): 2664B55510290198F89730FF4305EB3D58AF4FB4EF989827EA25DB8004962B4133889 9D4EB18D2C4385626A553DBDB096BAE37576ACA735659BFE82555A842D7
--------------------
A відправив пакет (k1, S1)
k1 (Encypted k) (Dec): 602457645592750360222082085086017887478790949128599308565115958048582111105424150328722226571907686941496320166702965233615207937420557737701741310408528 93
k1 (Encypted k) (Hex): 7307842C7CF3939CDF3F463EAEAB416AFEF20B44268CF7B158A39D237FBB68F40D784F9CA6E76E6DD9958FCFC712AACDC895B7F75613EFCC85C0984666D69B9D
--------------------
S1 (Encrypted Signature) (Dec): 705802447728854987827674146997011292535589559467644345123601890664015435948398521627503473278955967487752113491202087319970303458265973778901201014630833
S1 (Encrypted Signature) (Hex): 86C2E7AE9E6E5FBAF5A9124E79B30C03AF05F302BD76556ECD6E481CFC3CE84499E379E9DC47155171A270234AD3A2D880266C48C95E9BC34A4D10DFC64E29B1
--------------------
B отримав та обробив пакет
Отримане k (Dec): 201082509578672194218977850544758725332997983347071354368451222773832904626893054234062405594228888018777330870482905226590806723913718801121455148526869 5
Отримане k (Hex): 2664B55510290198F89730FF4305EB3D58AF4FB4EF989827EA25DB8004962B41338899D4EB18D2C4385626A553DBDB096BAE37576ACA735659BFE82555A842D7
--------------------
Автентичність відправника (A): ПІДТВЕРДЖЕНО
```

Висновки

У ході виконання комп’ютерного практикуму було досліджено, а також програмно реалізовано асиметричну криптосистему RSA, включаючи алгоритм генерації великих простих чисел на основі ймовірнісного тесту Міллера-Рабіна. Було розроблено функції шифрування, дешифрування та цифрового підпису, а також реалізовано протокол конфіденційного обміну сеансовими ключами.

Практична робота дозволила глибше зрозуміти математичні основи та принципи роботи RSA, які забезпечують високий рівень безпеки.