МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Кафедра інформаційної безпеки

КРИПТОГРАФІЯ

КОМП'ЮТЕРНИЙ ПРАКТИКУМ №4

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

Перевірили:

Виконав:

студент 3 курсу ФТІ

групи ФБ-91

Подус Олексій

Викладач:

Завадська Л.О

Савчук М.М.

Чорний О.М

КИЇВ 2021

Мета: 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 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().

Кожну операцію рекомендується перевіряти шляхом взаємодії із тестовим середовищем, розташованим за адресою http://asymcryptwebservice.appspot.com/?section=rsa. Наприклад, для перевірки коректності операції шифрування необхідно а) зашифрувати власною реалізацією повідомлення для серверу та розшифрувати його на сервері, б) зашифрувати на сервері повідомлення для вашої реалізації та розшифрувати його локально.

Хід роботи

Кандидати, що не пройшли перевірку частоти

```
100543811674367437908452058871740542026613690500355737311897647098925124121347
False
604827519993429588108645225457450653508403820471180407434460668809978012212 67
False
719378952615347078640267744259322265141117671363945495367272340553367212625 8
False
728279184960772959360330422939145289557112694942382342001858485524168255320 03
False
483859261092487034083916257472668391352080472902980841125409382113092850624 83
False
143429039662202132409511068917539061909866816293359690981560276015992969915 54
False
250116276935007195164182153172044311144122645535986288560138161215854866213 23
False
160596336898034803990231591481483807699638746444860159389963074542978861846 51
False
836483509175424505592409402593740542278367839990806069003255594547919500554 01
False
273597494100548087242605679425301273991119866385064040543670987806297873552 9
False
461137658730999186603571857005335687726689163544970527853940663522874319646 51
False
117941727106543900082432230785591366301967355215314002095072605700305181601 81
False
279538530729131764889294152951342434823002907147407427512545695716995428163 98
False
993904881400817377156979472941410946725214436067794743661364587617694049877 34
False
648952003733671001312603879320275765072339205037974301135341514799417653575 12
```

p 360338761620126183053368455657088882544166748145397310330833383097019433 36489

q 416543115770329541985195051453497593538716308706015941889419120242622877 05361

p1 9499709394788620071997446627884802134078474834341209713897902268517112149210 7

q1 3528546702545923402231309252361770098130406695887509357565856928414754492991 9

Відкритий текст:
3485905013154034027126245487008876778421266745043173056238140826409743768640469 7
3001601529491911628655509182470688773458841417032726016306369475571943 60

Для абонента А:

e:
2541059445405340597565867488177137463240864454450193510268992665154700931155520 2
8626831311638575974438515367272335095627306196396992865525380259908271903

d:
1042952789848789229428817287173848819182546251539783927814322782789846393021089 1
8090776602259588595309796955211011105006475139926870323682051084118695932 7

n:

1500966304980694001804544915107562947101675179459034432295270174953541065301648 4
6540374897303565672545958031738591452737896875779267105412530614411221752 9

шт:

1331691273895404905363271938589899917751029333314268652576716007466286488277474 8
8925293221184311219424316595715426987588833399192793178084458558580541156 6

цифровий підпис:

6265340888212735763090644114621476381334440064771685152600357937910183088776568 3
2014092829036263613237307035718736210931679956389319560362535149031075224

Для абонента Б:

e:
8619845150707024359506637091434482186918408170117164747404547322262291386159793 9
3993020907623046619911275979818846618819463590618788819388844501380080409

d:
1732175149485616888147162295642987700887022965932642167165971879336770531443859 6
7114705550779558351713366553339106344224717277999567587054184261020825327 3

n:
3352016826012591501518932806959691001146240444324080739055212916725440647025647 4
8021665439581066982728721407013064154537229363211635360296246356202664933 3

шт:
1584727433144869727964863754989282392311514778716184135173881926481371957149426 9
3717788358584686698788130819231107259013214808411649890824700003125706540 2

цифровий підпис:
9899534187340738095238133098965096221573290590630871027340228650096598096266164744552284876206194432310018831240080170660518975358135312518829178056743 90

Результат роботи програми:

```
Message was sent
From A to B
k1: 196317824599502452869261531431300548916750148178302315263268576424852645944405699167341402327326946525360625125666036904262755186980275848870265015480329 8
s1: 790780640526542236337284317961838175419417906351816911793910878021022458413754750922379658123897420931821530700878513677451326304179294009045831686979050
Message was verified!
```

Результат роботи на сайті: