МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря
Сікорського» ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

Криптографія

КОМП’ЮТЕРНИЙ ПРАКТИКУМ

Робота№4

Виконали студенти групи ФБ-83
Осінній Максим
Яненко Наталія

Перевірив:
Чорний О.М.

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

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

Відкинуті значення::

1156099920723685404186067048281256622711617703207844092634243596975439308056438682396053452322308340781657784191458031151264163410020654062772773273627473111568987584802386915521787745068455939078668832356684944032862659635263934400167602061454327725184122331883514727087130907305283740405673190304515415507011773238523567840422171823063538549491010581458578134967891293938995599679826891029221844485248292132156300367264514374296765815796830979433518436465957612137769303416260701259945464177074105702386399060395690237040070627666765572109787756672903850390331500407882445119378234876903175637562127648731021476726279114855559057287585004685758975257495241914174745581876003742617627105901066277585695296362649215416773129721503067217498143391528332829536461500480599225839197811003374583105522777291820903758042628975584437015132912322074086332980384678198661742295226256584486455955915180670072333907383961318218619055066253103757979063099567705445873727869893678385097273108747870252411495540191837783102645336294719938398245858648835037674968375663756664572686730094187349922329784948822798776917110664892657720904011753619067360386023502522726733277144976418175573185402632708829270801709801336229135957595591336136647267467419483767000553645990936976682322403940766876640507808950282910947801848249484600449725688676134427087100993992853939758547242436577431579263692245521510390399639042704545099115860729902836464516625044776674140990039905988714547264571224388383215205277013153473329441660096768771838199196786568812899329589580091559581686727757262679659469589665517353686380168543796256301993052033089409087011138942489271521550277948566074282853163210113992310765947631711232277480767171019072563377126986817874324747193061079739815697637356983263374432308298219579146371839014429513732118701978373695633820546850465887080902639506533478604191389014429513732118701978373695633820546850465887080902639506533478604111142651340292363369234007931819229549688611288946200473917140350903205324333969674957696782117676309709194490978066655966439053662862966017054529773341473100363933735973330428013680571672746531040461711864487918637703952356217020169831388528945288968100272548846707992018169968656956309775994813315003430043878633662513692579506491225245354317823129351397858860408705327927576618244267966065539833299387873127916057000845893838496125995040927623508251176266344579990504915333492513429055697475456378072652097921902313822166842818452632441310629107352030094999921801819507337437923241076775992401344465422670092317061393381501437765157519385075746158258947119336814153284605720712513691265178823890980613927824982613692529307547044886811356191220228040981927837511636378219268814140745524603307113308282393702262730306446571365949012192156485740374960005657567661249807533219686683054843259096020724629918822802603089391639933752302120793370749553584314903342223845821478829827648951808024960207048581017471286457856199714603455947737838727963257795533420066991970810817381748487188067939997274420522194321592856845055704081739009389023546400847431114277443108664234542366992337435785629331080796697736130317512769219247891053816770227736357339877923461991114598859593625157005461719140033346071084331655988240317109366517631818202934755997569731972603112375272255016080071514106759111397396941819200447585280720835027079277593495127284308260332496053082134105021213

p 61300007768369390949500709196286988022097454212930419547183561119074079913059

q 98581529243334143503135423794880828053058849730532141425707069451319379272601

p1 10167698043301863783175825065344011160615340871628422955054082056683290366951 7

q1 10920324716022746329372782756523377494717933558501211143104623676958179853034 1

A::

n

60430485084341172836522553404510642224035783844575595084346524634635435947658119602803926147080344769977873241387739907098629852299664600498751287407964590
e
38041217027704607629086593903961662638186619828184586215625390868827036011584413140428060258017152304786631343744630722456237269639748145078710775433940530
d
22412044095203625186236202475116869754706445655423254203932199956166362293179918775999018085778152692761455387965569357444828934454038079351976070558891170

B::
n
11103456424732545912893617037877324255161094915604213069563586973476803492312366881693828606788136088980321479391792049396327236364905279794419616561315297
e
59216499229907263911422764342114012257404026000076712193751328991556573661052644353061759163130220117417948480465179904886733686787908485803107609548103030
d
95474454735132614221419548663999753793518032431230426145850200018433899892184818248544082505437342781152791110922220326919511243973186513811909781981606070

BT::
56048702436795061544856448820205717051700191207291320384853650918418327879304647478432169377795062382590941176923948539128657211367773279058904422467703660
ШТ::
15396596485130912740433481580854350785587616583585372134001729201480229121243867367126057318970780724215264146748431476393170194826516856668321852788562860
Підпис::
56048702436795061544856448820205717051700191207291320384853650918418327879304647478432169377795062382590941176923948539128657211367773279058904422467703660

Tests::
start   256
conf   256
auth   256

Перевірка verify
{'verified': True}

Перевірка sign
{'signature':
'875F4CD0852EAC069657601E9B7BDF4E14ACDDCB94A70EDA8995289F49014266510198940A9ACB411159245FFEB2DD914386D0BD0BCFDC1419124CCEC9B22D15'}

Перевірка sendKey
{'key':
'1FEFB9CE09E0851BFD883ABD2768974318C4EC5A0DBFC28B05A87F53A3C9A7CDD2C69DCAFD80F61BCD4ED1C786EF67DD8A26E893237233E77799B847D510566B', 'signature':
'1BE94F1DBB99F096E245F53B0624AD4D551492BAC1A215B37B485F7AF62E339459F33E01898FDAB6C40FC89056F921F1C788428EBBA92CCF661BCAABDD8C922C'}

Перевірка create_message
{'key': '0100', 'verified': True}

Ми шифруємо, сервер дешифрує
{'cipherText':

'534045EAAE03BCE758BC88DE2B39050548BDE87173F9544ECB2498F668562F102240599AF3A13C8
B48057FD6BA88BEA0373A9949EF88DA9DBCD9D7DF5FDAFCEC'}
b'{"message":"0100"}'

Сервер шифрує, ми дешифруємо
{'cipherText':
'666D4B877C9990C7A83F51C7D9CB98CB60F75081CC35144FABE811BF07F9D49464E79287FCE54C2
5CDF3305474C106A578FB67B2A423A7B251C0B58726D1EE82'}
res 100

Висновки: під час виконання даної роботи ми реалізували тест Міллера-Рабина, алгоритм асиметричного шифрування RSA, та на їх основі організували роботу протоколу конфіденційного розсилання ключів по відкритому каналу з підтвердженням справжності.