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

Криптографія

Комп’ютерний практикум №4

Виконали студенти групи
ФБ-81 Скляр Б.Ю., Висіцький
С. І.
Перевірив: Чорний О.М.

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 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 1 1 pq p q ; p і q – прості числа для побудови ключів абонента A, 1 p і 1 q – 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 1 e n та секретні d і 1 d . 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

1. Була написана функція генерації випадкових простих чисел заданої довжини. Перевірку на простоту проходили за допомогою теста Міллера-Рабіна:
   0xfbddb407ff4e49d1fcf65b11077ac8d6c96810c69444ef1dbd83dfe98bba947b is not a prime number, will find more.
   0x8c4656e62c3bcdcd47ed5270b169978514c75c78e26b05a66a8d6289d9ba26e9 is not a prime number, will find more.

2. Сгенеровані p,q для локального кейсу:
   Pair p and q for Alice:0xa578ea932ffa869f08b7a9ddf26e0f3758003df5da3cc417c33a317af1bd1869, 0xc90a2e5452a573be9eafb5ca11cd9c0ab2dbbb44e90f8f658af05fe61bc4b9bb
   Pair p and q for Bob:0xd4b18143528116bf5e0d646888e5dac93ea576ff1253142d51d42572f47c360f, 0xcb384c7fb18b984884dfe54e0a4939ffb1beba35196f66d1e2d7b995c7db1e53

3. Відкритий ключ і протокол ReceiveKey:
   Open key (n=0x81f284d8f5aaab49c05fe29eeb81d10b4eb6d83ee50dd33604db7d08d5323a95136572fc7c68bb2bfea1cf3f6a2dfa3ca6c977419f7cf23d5b46e92fff28b5b3, e=0x56f431ad7580f215e4a382b5256d140e94edee0c84c94f4a779fb455f09841f0bd33fc98da05217e8017480fb06990ff5e68f74237db335bb7061c4d1d1d3f15) and secret key (d=0x5d35ad3bffc0cb81218e8a96fc57bd09c11e08811f00a05349ba0cfba040dfdbc71eb30e95ff0a7a6060a73582b471e7f0390504e0c151b89d5e09ba816ffcfd, pq=0xa578ea932ffa869f08b7a9ddf26e0f3758003df5da3cc417c33a317af1bd1869, 0xc90a2e5452a573be9eafb5ca11cd9c0ab2dbbb44e90f8f658af05fe61bc4b9bb) for Alice.

   Open key (n=0xa8d787e373183b715ae736eff11137fa580a3693de1f9f25ca0341d4d1a3034d6ab5bf87f71ed93acb5fc2d1dabf941c963dd5d9080bbac07d849eeeba7048dd, e=0x68f66c341bafc8b9560599a20dec30ae9b0a790ea2a5332adc7a6f4b5381ad83140cd4b85b5b62260f6a590f6ea88790398841dd43e4938d7179da667f6034db) and secret key (d=0x215e032ff95f5c57e6dd2de137611f644e2689f97d44a77e8307b06101b6fe7cf827430aed2518ec51c7286255d07195e78a6b4b3fe6b08a987232c85fcfff4b, pq=0xd4b18143528116bf5e0d646888e5dac93ea576ff1253142d51d42572f47c360f, 0xd4b18143528116bf5e0d646888e5dac93ea576ff1253142d51d42572f47c360f) for Bob.

   Alice`s secret message '0x65de8b71b6d737e9d23694e1fda861fd0a9b5c5191ebf87250596bedf41a2e3a3f3c732f619030873b1b297a3ac840dce7a7cfb81809652ad4b28e0b2a305679' for Bob.

Text is encrypted.
emsg=0xa2cdb36d0769c633b67579e0a4a8a73ca50a1e770a4524f2671556fbafb1b77a2251dda198a8429a5de0b60f999409c98672d64bc3c6d6b2ad82fa27faf5bcf7

Message was signed with a secret key of Alice.

digsA=0x7844e66c3578d5cc35728e487a0097123892799f112bcc1579d099c63df6c38873724e540b7cee870344c29c50d5a83e68215e70a32ef591c86da227ee9ee303

Sign of Alice was signed with open key of Bob.

digsA1=0x3686caf7c97ff6c2b00399a7caa562e92b395fb33c51879c12876a5d20a13b221fc8c604c3637fa86f8fefbcbb633d6fc7fc5b5283c967cf7c1244ec0cc0da56

decmsg is
0x65de8b71b6d737e9d23694e1fda861fd0a9b5c5191ebf87250596bedf41a2e3a3f3c732f619030873b1b297a3ac840dce7a7cfb81809652ad4b28e0b2a305679

decs is
0x7844e66c3578d5cc35728e487a0097123892799f112bcc1579d099c63df6c38873724e540b7cee870344c29c50d5a83e68215e70a32ef591c86da227ee9ee303

decs to power e by module n is(sa^emod(n))
0x65de8b71b6d737e9d23694e1fda861fd0a9b5c5191ebf87250596bedf41a2e3a3f3c732f619030873b1b297a3ac840dce7a7cfb81809652ad4b28e0b2a305679
[68058944150625056981950181794610763357643716014636219111292580249446620486396584317744234898135630659394030355527977886322714285926085688212075134368291 07,
45541443839447005974284133460618223538288595712007588698062988090491329352473380367571922939031378871419982222730386213139176141724675704832776548057904 85]
\Work done

4. SendKey:
An open key
(n=0x74fdfee2e65c3b73f1f37728948a6c086fc134188fc809782196a34b71bbe5197b63e2eadd2c61eb51570eff52b15aba7a10f7f2e119fc288d3de1aa3562eedf,
e=0x578d7fa293bc9b2cb065020ebc7538a3d69e09001b23b6b701bab6a7b9cb8d3cbb6a06f8e0852880709375bc847af79ad3382aaa0d78a64f22ec409626f911d9) and secret key
(d=0x45463fad2561382caf30a1a7af29951beb0671bff632c46b2b76c7153933bebc4a26f1283d0eea0da92ad80ca199dc29afbf35c7e23b3ba543c6d5bb12b6db79,
pq=0xae62dd30437d0f9b47db44ad7932e2775ea1e36c4a918e0e2722d582afe15d4b,
0xabbec74d4915178ee39ab2da821dda7adcaf10d8f41742803de1481eafc29c3d) for Alice were generated.

Alice generated a secret message
'0x2b716f8f75b1cc7cf3dc1a811a4088388584033d4200fd1f624d9133f5d1f7b0b1f8fc24219fb9f0d5f6a6a797b9770c0b9b2303fc5ebfb12a50c99679a77b2d' for Bob.

Site has generated an open key
(n=504029443489576183279431106042042896239174505273500913842288572072
717271804983481537412992122943247714017212121327263124258372358666161
2907157407241976091, e=65537) for Bob)

Text is encrypted.
emsg=0x36b4315f1576560cb58498292714dc8bf9db8cf603a7b34cc3079fab6aa5f1fe
3ce7192974c7cd2699ad14c5bb7b9d863da082d6b9d41e15d10a8d3e4c018a82

Message was signed with a secret key of Alice.

digsA=0x72e95003e23cbf54f0811157eeb427d5bafcfcf105eafc0ab5319f0d2bbe982f4
dcf943d9f70d9f8e502b05fdac537f9ed4597127a9f7deacd5c1f9575e2fa45

Sign of Alice was signed with open key of Bob.

digsA1=0x449eaa20ee26d06ce0de44da2bae901770b02d3999125359e218043619c94
b55748f51b2627e0d29bf6628228adf94fa0e994a26a8d438e5b544945548c385e2

z=
0x36b4315f1576560cb58498292714dc8bf9db8cf603a7b34cc3079fab6aa5f1fe3ce719
2974c7cd2699ad14c5bb7b9d863da082d6b9d41e15d10a8d3e4c018a82
s=
0x449eaa20ee26d06ce0de44da2bae901770b02d3999125359e218043619c94b55748f5
1b2627e0d29bf6628228adf94fa0e994a26a8d438e5b544945548c385e2

Висновки

В цьому лабараторному практикумі ми ознайомились з тестом числа на простоту (тест Міллера- Рабінсона) і способами підбору простих чисел великої довжини. Реалізували систему захисту інформації на основі криптосхеми RSA, а також протокол конфіденційного розсилання ключів по відкритих каналах зв'язку з підтвердженням справжності відправника.