ЛАБОРАТОРНА РОБОТА № 9

ТЕМА: УЗГОДЖЕННЯ СЕАНСОВОГО КЛЮЧА ПО ПРОТОКОЛУ ДІФФІ-ХЕЛЛМАНА

МЕТА: НАДАТИ ПРОГРАМНУ РЕАЛІЗАЦІЮ КРИПТОСИСТЕМИ НА ОСНОВІ ДАНОГО ПРОТОКОЛУ

ТЕОРЕТИЧНІ ВІДОМОСТІ

В алгоритмі Діффі-Хеллмана симетричний сеансовий ключ не генерується і не розподіляється між учасниками. Алгоритм забезпечує формування одного й того ж секрету двома сторонами, який можна використовувати для побудови сеансового ключа в симетричному алгоритмі.

Основні повідомлення в протоколі Діффі-Хеллмана представляються наступною діаграмою :

A↔B: N,Q;

A: x; A→B: $L=Q^x \pmod N$;

B: y; B→A: $M=Q^y \pmod N$;

A: $k_x = M^x \pmod N$;

B: $k_y = L^y \bmod N$);

Алгоритм гарантує, що $K_y=K_x$ і можуть бути використані в якості секретного ключа для шифрування.

Платформа .NET надає реалізацію CNG алгоритму Діффі-Хеллмана на еліптичних кривих (ECDH) через використання об'єктів класу **ECDiffieHellmanCng** з простору імен **System.Security.Cryptography**.

Клас **ECDiffieHellmanCng** дозволяє двом сторонам обмінюватися матеріалом закритих ключів, навіть якщо взаємодія здійснюється по відкритим каналам. Обидві сторони можуть обчислити одне і те ж таємне значення, яке називається *секретною угодою*. Секретна угода може надалі використовуватися для різних цілей, в тому числі як симетричний ключ. Проте, замість прямого представлення секретної угоди клас **ECDiffieHellmanCng** робить деяку її обробку перед наданням значення. Ця постобробка називається *функцією формування ключа* (key derivation function, KDF).

Порядок узгодження сеансового ключа шифрування за допомогою об'єктів класу **ECDiffieHellmanCng** такий:

1. Аліса створює сховище ключів і експортує з нього свій публічний ключ для передачі Бобу:

```
CngKey aliceCngKey = CngKey.Create(CngAlgorithm.ECDiffieHellmanP256);

byte[] alicePublicKeyBlob = aliceCngKey.Export(CngKeyBlobFormat.EccPublicBlob);
```

2. Боб створює сховище ключів і експортує з нього свій публічний ключ для передачі Алісі:

```
CngKey bobCngKey = CngKey.Create(CngAlgorithm.ECDiffieHellmanP256);

byte[]  bobPublicKeyBlob = aliceCngKey.Export(CngKeyBlobFormat.EccPublicBlob);
```

3. Аліса імпортує публічний ключ Боба в окреме сховище:

```
CngKey bobPubCngKey = CngKey.Import(bobPublicKeyBlob,
CngKeyBlobFormat.EccPublicBlob);
```

4. Аліса створює екземпляр класу ECDiffieHellmanCng з ключами, що беруться з її сховища ключів:

```
ECDiffieHellmanCng aliceAlgorithm = new ECDiffieHellmanCng(aliceCngKey)
```

5. Аліса отримує секретний ключ з ключового матеріалу:

```
byte[] aliceKey = aliceAlgorithm.DeriveKeyMaterial(bobPubCngKey);
```

6. Боб імпортує публічний ключ Аліси в окреме сховище і використовує його для отримання секретного ключа з ключового матеріалу:

```
CngKey alicePubCngKey = CngKey.Import(alicePublicKeyBlob,
CngKeyBlobFormat.EccPublicBlob);

ECDiffieHellmanCng bobAlgorithm = new ECDiffieHellmanCng(bobCngKey);

byte[] bobKey = bobAlgorithm.DeriveKeyMaterial(alicePubCngKey);
```

ПОРЯДОК ВИКОНАННЯ РОБОТИ

1. Ознайомитись з алгоритмом Діффі-Хеллмана.
2. Написати програму для узгодження сеансового ключа і шифрування та розшифрування повідомлень з використанням алгоритму AES.
3. Підготувати звіт про виконання роботи. Звіт оформлюється у вигляді документу Word з такою структурою: титульний лист, тема і мета роботи, блок-схема алгоритму методу, програмний код.
4. Електронну копію звіту відправити за адресою: George@aprodos.kpi.ua.