**Лабороторна работа №11**

**Тема:** Алгоритм цифрового підпису “ECDSA”

**Мета:** Практичне відпрацювання та закріплення теоретичних відомостей з теми алгоритм цифрового підпису “ECDSA”

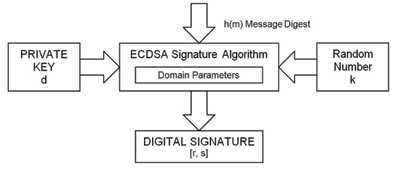
**ECDSA (Elliptic Curve Digital Signature Algorithm)** - алгоритм з відкритим ключем для створення цифрового підпису, аналогічний за своєю будовою DSA, але визначений, на відміну від нього, не над кільцем цілих чисел, а в групі точок еліптичної кривої.

**Вибір параметрів**

Для підписування повідомлень необхідна пара ключів - відкритий і закритий. При цьому закритий ключ повинен бути відомий тільки тому, хто підписує повідомлення, а відкритий - будь-кому хто бажає перевірити справжність повідомлення. Також загальнодоступними є параметри самого алгоритму.

**Параметри алгоритму**

Вибір хеш-функції ***H(x)***. Для використання алгоритму необхідно, щоб повідомлення, яке підписується, було числом. Хеш-функція повинна перетворити будь-яке повідомлення в послідовність бітів, які можна потім перетворити в число. Вибір великого простого числа ***q*** - порядок однієї з циклічних підгруп групи точок еліптичної кривої. Зауваження: Якщо розмірність цього числа в бітах менше розмірності в бітах значень хеш-функції ***H(x)*** то використовуються тільки ліві біти значення хеш-функції. Простим числом ***p*** позначається характеристика поля координат ***Fp.***

******

**Генерування ключів ECDSA**

Для простоти будемо розглядати еліптичні криві над полем ***Fp*** , де ***Fp*** - кінцеве просте поле. Причому, якщо необхідно, конструкцію можна легко адаптувати для еліптичних кривих над іншим полем.

Нехай ***E*** - еліптична крива, визначена над ***Fp***, і ***P*** - точка простого порядку ***q*** кривої ***E( Fp )***. Крива ***E*** і точка ***P*** є системними параметрами. Число p - просте. Кожен користувач - умовно назвемо його Аліса - конструює свій ключ за допомогою наступних дій:

1. Вибирає випадкове або псевдовипадкове ціле число ***x*** з інтервалу ***[1,q-1]***.
2. Обчислює (кратне) ***Q=xP***.

Відкритим ключем користувача Аліси ***A*** є точка ***Q***, а закритим - ***x***.

Замість використання ***E*** і ***P*** в якості глобальних системних параметрів, можна фіксувати тільки поле **Fp** для всіх користувачів і дозволити кожному користувачеві вибирати свою власну еліптичну криву ***E*** і точку ***P*** Element of ***E(Fp)***. В цьому випадку певне рівняння кривої ***E***, координати точки ***P***, а також порядок ***q*** цієї точки ***P*** повинні бути включені у відкритий ключ користувача. Якщо поле ***Fp*** фіксоване, то апаратна і програмна складові можуть бути побудовані так, щоб оптимізувати обчислення в тому полі. У той же час є величезна кількість варіантів вибору еліптичної кривої над полем ***Fp***.

**Обчислення цифрового підпису**Для того, щоб підписати якесь повідомлення, для якого підраховано значення ***h*** хеш-функції ***H***, користувач ***A*** повинен зробити наступне:

1. Вибрати випадкове ціле число ***k*** Element of ***[1, q-1].***

2. Обчислити ***k \* P = (x1, y1)*** і покласти в ***r = x1 mod q***, де ***r*** виходить з цілого числа ***x1*** між ***0*** і  ***(p-1)*** приведенням по модулю ***q***.

Зауваження: якщо ***r = 0***, то рівняння підписи ***s = k ^ - 1 (h + x \* r) mod q*** не залежить від секретного ключа ***x***, і отже, ***(r, s)*** не підходить в якості цифрового підпису. Значить, в разі ***r = 0*** необхідно повернутися до кроку 1.

3. Обчислити ***k ^ -1 (mod q)*** і покласти ***s = k ^-1 (h + x\*r) mod q***, де ***h*** - значення хеш-функції підписується повідомлення.

Зауваження: якщо ***s = 0***, то значення ***s ^-1 mod q***, потрібне для перевірки, не існує. Значить, в разі ***s = 0*** необхідно повернутися до кроку 1.

Підписом для повідомлення є пара цілих чисел ***(r, s).***

**Перевірка цифрового підпису**

Для того, щоб перевірити підпис користувача Аліси ***(r, s)*** на повідомлення, користувач Боб ***B*** повинен зробити наступне:

1. Отримати підтверджену копію відкритого ключа ***Q*** користувача ***А***;

2. Перевірити, що числа ***r*** і ***s*** є цілими числами з інтервалу ***[1, q-1]***, і обчислити значення хеш-функції ***h*** від повідомлення;

3. Обчислити ***u1 = s ^-1 h mod q*** і ***u2 = s ^-1 r mod q***;

4. Обчислити ***u1 \* P + u2 \* Q = (x0, y0)***, і щодо ***x0***, як цілого числа між ***0*** і ***(p - 1)***, покласти ***v = x0 mod q***;

5. Прийняти підпис, тоді і тільки тоді, коли ***v = r***.

Зауважимо, що, якщо користувач Аліса вирахувала свій підпис правильно, то ***u1P + u2Q = (u1 + xu2) P = (s ^- 1 \* h mod q + x \* s ^ - 1 \* r mod q) \* P = k \* P***, так як ***k = s ^- 1 (h + xr) (mod q)***, і тому ***v = r.***

Для підтвердження публічного ключа ***Q*** потрібно виконати наступне (***O*** тут позначає нескінченно віддалену точку):

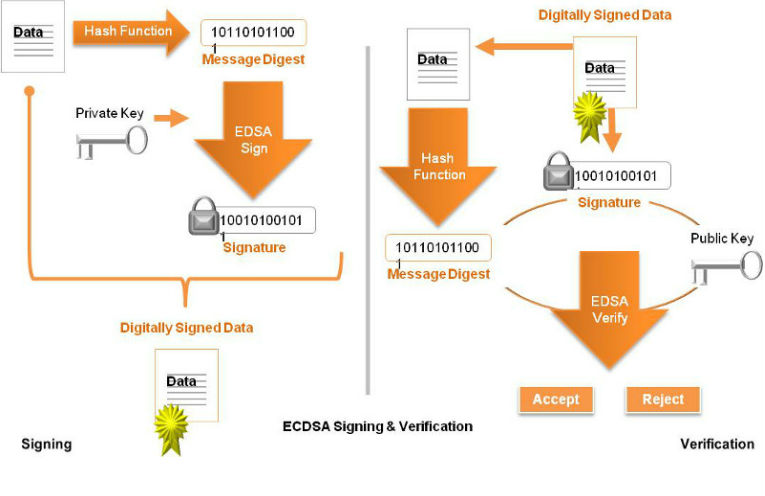
Перевірити, що ***Q*** не дорівнює ***O*** і координати вірні;

Перевірити, що ***Q*** лежить на кривій;

Перевірити, що ***qQ = O***;

**Переваги ECDSA перед DSA**

ECDSA є дуже привабливим алгоритмом для реалізації ЕЦП. Найважливішою перевагою ECDSA є можливість його роботи на значно менших полях Fp. Як, загалом, з криптографією еліптичної кривої, передбачається, що бітовий розмір відкритого ключа, який буде необхідний для ECDSA, дорівнює подвійному розміру секретного ключа в бітах. Для порівняння, при рівні безпеки в 80 біт (тобто атакуючому необхідно приблизно 2^80 версій підписи для знаходження секретного ключа), розмір відкритого ключа DSA дорівнює, принаймні, 1024 біт, тоді як відкритого ключа ECDSA - 160 біт. З іншого боку розмір підпису однаковий і для DSA, і для ECDSA: ***4t*** біт, де ***t*** - рівень безпеки, який вимірюється в бітах, тобто - приблизно 320 біт для рівня безпеки в 80 біт.



**Висновок:** Стійкість алгоритму шифрування ґрунтується на задачі дискретного логарифма в групі точок еліптичної кривої. На відміну від задачі простого дискретного логарифма і задачі факторизації цілого числа, не існує субекспоненціального алгоритму для задачі дискретного логарифма в групі точок еліптичної кривої. З цієї причини «сила на один біт ключа» істотно вище в алгоритмі, який використовує еліптичні криві.