R07922106 曾俊為 Fintech Homework 2

1. 4G

(10338857399563508035974916425421659830878883530402360147780309523428649499368 3 :
37057141145242123013015316630864329550140216928701153669873286428255828810018 : 1)

2. 5G

(21505829891763648114329055987619236494102133314575206970830385799158076338148 :
98003708678762621233683240503080860129026887322874138805529884920309963580118 : 1)

3. Q = 2106G

(91063512616269295105126874043630389817347371097093778755952242141067937507215 :
17164231569873168822386559926976266793122514524372287693171331474023035444063 : 1)

4. Double-and-add algorithm

d=2106

將 2106 轉成二進位表示成 100000111010

所以 2106G=2(2(2(2(2(2(2(2(2(2G))))))+1)+1)+1))+1)

總共需要 11 個 doubles 和 4 個 additions。

5. Evaluate 2106G as fast as possible

距離 2106 最近的 2 的 power 為 2 的 11 次方也就是 2048，如果要用減法的方式又要減少計算量勢必要從 2 的 12 次方也就是 4096 扣回來，但是 4096-1990 用 double-and-add algorithm 顯然不會比原本的 11 + 4 = 15 次計算量還少，因為 4096 就需要 12 個 doubles，1990 不可能在 3 個計算內完成，所以保持第四題的計算方式上還是比較快。

6. ECDSA Signing

照講義 P31 步驟 ，以下用 secp256k1 來當橢圓曲線。

(1) calculate e=hash(m)，這裡的 hash function 為自訂 3*m+2021。

```
sage: def hash(m):
....:     answer = 3*m+2021
....:     return answer
```

(2) let z be the 6 leftmost bits of e，這裡假設 message m 為 20210505。

```
sage: m=20210505
sage: e=hash(m)
sage: e
60633536
sage: z = 606335
```

(3) select a random integer k from [1,n-1]

```
sage: k = randint(1,n-1)
sage: k
106581734417594869786225077978122487501813801923270667247542498411314370175 31
```

(4) calculate the curve point (x1,y1) = k*G

```
sage: x1 = (k*G)[0]
sage: x1
401560289144635691256669952233731099339386605201780390340514006709797944419 87
sage: y1 = (k*G)[1]
sage: y1
488630565429367021328003762201290914170863952234819170721145006395594068120 71
```

(5) calculate r =x1 mod n, go back to step3 if r = 0

```
sage: def mod(test):
....:     quo = test//n
....:     remainder = test-quo*n
....:     return remainder
....:
sage: mod(x1)
0
```

這裡我經過多次嘗試，結果都為 0，所以接下來的步驟無法繼續進行。

(6) calculate s = k^(-1)(z+r*2106) mod n, go back to step3 if s = 0

(7) signature = (r,s)

7. ECDSA Verification

照講義 P32 步驟

(1) verify r and s are integers in [1,n-1]，由於簽章未完成且 r = 0，這步驟也無法繼續進行。

(2) calculate e =hash(m)

(3) let z be the 6 leftmost bits of e

(4) calculate w = s^(-1) mod n

(5) calculate u1 = zw mod n and u2 = rw mod n

(6) calculate the curve point(x1,y1) = u1*G+u2*Q

(7) the signature is valid if r=x1 (mod n)

8. Quadratic polynomial

P(1) = 10, P(2) = 100, P(3) = 2106

By Lagrange Interpolation,

P(x) = 10*(x-2)(x-3)/(1-2)(1-3)+100*(x-1)(x-3)/(2-1)(2-3)+2106*(x-1)(x-2)/(3-1)(3-2)

可得 P(x) = 958x^2-2784x+1836。