University College London
Department of Computer Science

Cryptanalysis Lab 6

J. P. Bootle

April 4, 2018

jonathan.bootle.14@ucl.ac.uk
Version 2.0

Implementing the Pollard-Rho Algorithm

Click on the green letter before each question to get a full solution. Click on the green square to go back to the questions.

EXERCISE 1.

(a) Write a function `pollard_rho` which implements the low-memory version of the Pollard-Rho algorithm. The function should take input N, and output a, b such that $N = ab$. Run the algorithm for a fixed number of iterations. You may wish to structure your code as follows.

- Definition of a sub-function for iteration.
- Set the number of iterations to do.
- Main loop using the iterative function.
- At each step of the main loop, compute a greatest common divisor.
- Return a factorisation $[a, b]$ or output 'Fail'.

(b) According to the analysis of the running time of the Pollard-Rho algorithm, how many iterations should we expect to use before the algorithm succeeds in finding a factorisation?

(c) Test your algorithm by attempting to factorise the integers $M_n = 2^n - 1$, for $n = 80, 85, 90$. What is the largest value of n that your program can handle in 10 seconds?

Elliptic Curves

Click on the green letter in front of each sub-question (e.g. (a)) to see a solution. Click on the green square at the end of the solution to go back to the questions.

EXERCISE 2. Let $E : y^2 = x^3 + ax + b$ be an elliptic curve. Let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$. Write $+$ for the operation of adding two points. Beware: $P + Q \neq (x_1 + x_2, y_1 + y_2)$!

(a) Watch the tutorial on elliptic curve point addition at https://www.youtube.com/watch?v=XmygBPb7DPM.

(b) Browse the internet to find the formulae for the coordinates of $P + Q$ when $P \neq Q$. What about when $P = Q$? You can assume that $Q \neq (x_1, -y_1)$ since things are slightly different in this case.

(c) Let $E : y^2 = x^3 + 3x + 3$ be an elliptic curve, defined over $\mathbb{F}_7$.

Two points on the curve are $P = (4, 3)$ and $Q = (3, 2)$. Verify that $2^*P = Q$ (remember that $2^*P = P + P$).

(d) Construct E, P, Q in SAGE using the following commands. Check your answer to the previous part by typing $2 * P$ (the answer will have three coordinates, for reasons to be explained in lectures, but ignore the last coordinate). What is $P + Q$?

```
p = 7
E = EllipticCurve( GF(p),[3,3] )
P = E(4,3)
Q = E(3,2)
```

(e) Type *E.cardinality*() to find out how many points lie on this elliptic curve.

(f) Type *E.gens*() to obtain a set of points which generate all points in the elliptic curve group.

Elliptic Curve Diffie-Hellman

Click on the green letter in front of each sub-question (e.g. (a)) to see a solution. Click on the green square at the end of the solution to

go back to the questions.

EXERCISE 3. In this exercise, you will use Sage and share a Diffie-Hellman Key with a partner, using points on an elliptic curve. To create an elliptic curve E defined by $y^2 = x^3 + ax + b$ over $\mathbb{F}_p$, use `E = EllipticCurve(GF(p),[a,b])`.

(a) Create an elliptic curve E defined by $y^2 = x^3 + 70x + 355$, over the finite field of size 1031.

(b) The command `n = E.cardinality()` sets n to be the number of points on the curve. What is the value of n? What properties should n have in order to be suitable for Diffie-Hellman?

(c) Typing `E.gens()` gives a set of points which generate all the points on the elliptic curve. In this case, there is only one generator, and `P = E.gens()[0]` sets P to be a group generator for this curve. If $P = (x : y : z)$, then your partner can get P by typing `P = E(x,y,z)`.

(d) Choose a random integer a such that $0 \leq a < n$. Your partner should choose b similarly.

(e) Use Sage to find the elliptic curve point $A = a^*P$, and give this

to your partner. For example, if $A = (x : y : z)$ then your partner can type `A = E(x,y,z)` to get A.

Your partner should compute $B = b^*P$ and give this to you in the same way.

(f) Use Sage to find $a^*B = (ab)^*P$. Your partner will also find $(ab)^*P$ via b^*A. The point $(ab)^*P$ is your shared secret key. Check that you and your partner computed the same answer.

Elliptic Curve Factorisation Algorithm

Click on the green letter in front of each sub-question (e.g. (a)) to see a solution. Click on the green square at the end of the solution to go back to the questions.

EXERCISE 4. In this exercise, you will use Sage to explore how integers are factored using elliptic curves.

(a) To create an elliptic curve Ep defined by $y^2 = x^3 + ax + b$ over $\mathbb{F}_p$, use `E = EllipticCurve(GF(p),[a,b])`. Create an elliptic curve Ep defined by $y^2 = x^3 + x + 4$, over the finite field of size 11.

(b) To create a curve EN defined by $y^2 = x^3 + ax + b$ over $\mathbb{Z}_N$, use `E = EllipticCurve(Integers(N),[a,b])`. Create a curve EN defined by $y^2 = x^3 + x + 4$, over the ring of integers modulo 438713.

(c) Type $PN = EN(100584, 115601)$ to create the corresponding point on EN. Similarly, type $Pp = Ep(100584, 115601)$ to create the same point, reduced modulo 11, on Ep. Type `Pp` to view the point modulo 11. The point should be expressed as $(x : y : 1)$. The point at infinity is $(0 : 1 : 0)$.

(d) Type `Ep.cardinality()` to find out the number of elliptic curve points modulo 11. What is the number of points? Type `a*Pp` to compute multiples of the point Pp. What is the order of Pp in the elliptic curve group Ep?

(e) Set $QN = 8 * PN$ and use SAGE to compute QN. Now, try to compute $9 * PN = QN + PN$. What happens? Compute the difference between the x coordinates of PN and QN, and compute the greatest common divisor of this with N. Look at the formulae for adding elliptic curve points. Does this explain the error?

(f) Set $N = 20077$. Consider the curve E defined by the equation

$y^2 = x^3 + x + 5$. Assume that N has a prime factor p with $\mid E(\mathbb{Z}_p) \mid$ being 5-powersmooth. Given that $P = (427, 466)$ is a point on $E(\mathbb{Z}_N)$, factor N.

Solutions to Exercises

Exercise 1(a) The following code implements the Pollard-Rho algorithm.

```
def pollard_rho(N):
    n = floor(sqrt(sqrt(N))) # adjust this value
    ai = randint(1,N-1)
    a2i = ai
    for k in range(1,n):
        ai = (ai*ai + 1) % N
        a2i = (a2i*a2i + 1) % N
        a2i = (a2i*a2i + 1) % N
        d = gcd(abs(ai-a2i),N)
        if not (d in [1,N]):
            return [d,floor(N/d)]
    return 'fail'
```

□

Exercise 1(b) According to the heuristic analysis based on the Birthday paradox, we would expect to succeed after $O(\sqrt{p})$ iterations, where p is the smallest prime factor of N. □

Exercise 2(b) If $P \neq Q$, we set $s = (y_1 - y_2)(x_1 - x_2)^{-1}$. If $P = Q$, we take $s = (3x_1^2 + a)(2y_1)^{-1}$. Then, $(x_3, y_3) = (x_1, y_1) \oplus (x_2, y_2)$, where $x_3 = s^2 - x_1 - x_2$, and $y_3 = s(x_1 - x_3) - y_1$.

These formulae come from the definition of addition on an elliptic curve that you saw in the video. This uses different points of intersection between straight lines and the curve. □

Exercise 2(c) Substituting the coordinates of P into the correct formula from the previous part shows that $2^*P = Q$. □

Exercise 2(d) You should find that $P + Q = (1, 0)$. □

Exercise 3(a) Use `E = EllipticCurve(GF(1031),[70,355])` to produce the correct elliptic curve. □

Exercise 3(b) You should get $n = 1009$. For secure Diffie-Hellman key exchange, we ideally want n to be large and prime so that the Discrete Logarithm problem is hard in the elliptic curve group. □

Exercise 3(c) An example generator is the point $P = (5 : 393 : 1)$. It doesn't matter which generator you use, as long as you and your partner are using the same generator. □

Exercise 3(d) You can use `a = randint(0,1009)` to get a. □

Exercise 3(e) The point a^*P is computed in Sage exactly as written here: `a*P`. □

Exercise 3(f) Get b^*P from your partner. □

Exercise 4(a) Use `Ep = EllipticCurve(GF(11),[1,4])` to produce the correct elliptic curve. □

Exercise 4(b) Use `EN = EllipticCurve(Integers(438713),[1,4])` to produce the correct curve. □

Exercise 4(d) There are 9 points on the elliptic curve Ep defined modulo 11. The order of a group element divides the order of the group, 9, and Pp is not the point at infinity, so we only have to check whether $3 * Pp$ is equal to the point at infinity or not. However, $3 * Pp = (3 : 1 : 1)$, so 3 is not the order of Pp. Therefore, the order of Pp is 9. □

Exercise 4(e) You can use `a = randint(0,1009)` to get a. □