Understanding ECC
HW4 - CNS Sapienza

Gianfranco Romani 1814407

20 November 2020

1 Introduction

Nowadays the most used asymmetric cryptography for encryption and digital signature is RSA. But to be considered secure, RSA has to use a very big key length and the number of bits required will continue to increase over time (1024 bits were considered enough until not much time ago and now it is suggested to use 2048 bits instead). As a result of that RSA is much slower than other cryptography schemes like AES. Fortunately there are other asymmetric systems that can be used with smaller keys, for example Elliptic curve cryptography (ECC) which is the subject of this report.

2 Elliptic curve cryptography(ECC)

The idea on which ECC is based on is algebraic structure of elliptic curves over finite fields. An elliptic curve is a set of points that satisfy the equation $y^2 = x^3 + ax + b$. This kind of curves have some features that make them useful. First of all, a line between two points on a curve of this kind always intersects with a third point of the curve. This allows to 'jump' all over the curve quite easily and get in an endpoint such that it is very difficult to reverse the path. The idea is to use these two points (the starting one, P, and the endpoint T) and the number, d, defining how many 'jumps' were done. Very useful to our purpose is to define a group over the elliptic curve. A group is defined as a set of elements and an operation, the addiction, that have to respect some properties:

- closure: if x and y are members of the group than x+y is a member of the same group;

- associativity: (x+y)+z = x+(y+z):
- identity element 0: x+0=x;
- inverse b: a+b=0;

If we add the property of commutativity, than the group is called abelian. The identity element and the inverse are unique. A group over elliptic curves has as elements the points of the curves, as the identity element the point at infinity, which is indicated with 0. The inverse of a point P is his symmetric about the x-axis. Addiction is given by the 'jumps' seen before and it works in the following way: given two points P and Q, we draw the line between the two points and then we extend it until it intersects with the curve. At this point we mirror the point to obtain the point $R = P + Q$. This mathematical operation is called point addiction.

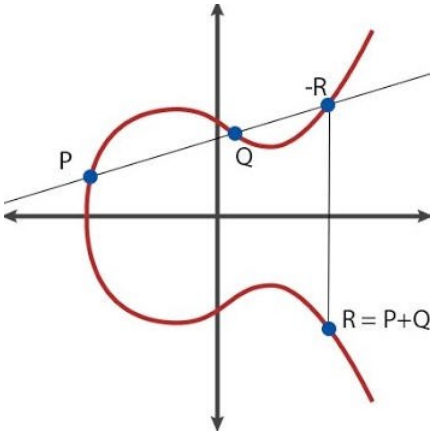

Figure 1: Example of an elliptic curve and the point addiction R = P + Q

To turn this geometric method into an algebraic one, given two points, $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$, we need to compute the slope of the line passing through them, $m = \frac{y_P - y_Q}{x_P - x_Q}$, and its intersection, $R = (x_R, y_R)$, that is obtainable from $y_R = y_Q + m(x_R - x_Q)$.If we are adding P to itself, we are just taking the line tangent to the point and seeking for the intersection with the curve. Sometime this is called point multiplication (or point doubling).

We can define another operation, the scalar multiplication, that is $d*P = P+P+...+P$, so adding P to itself for d times. Computing T, given P and d, can seem difficult but there is an algorithm (double and add algorithm, similar to square-and-multiply algorithm) that can do it in $O(\log d)$ (or

$O(k)$ with k the length in bits), while obtaining d, knowing P and T, is what is called the logarithm problem. Now, applying some multiplication, we can see some patterns repeating that can help us to write an algorithm to solve this problem. To avoid that, we reduce the domain of the curve to a finite field to convert the problem into the discrete logarithm problem that is way more complicated (while the multiplications to reach T are still easy). An example of finite field is the set of integers modulo p, where p is a prime number, on which are defined two binary operations: addition and multiplication. The properties of closure, associativity and commutativity are valid. The set of integers of the field consists in all the integers from 1 to p-1. Addition and multiplication work as in modular arithmetic. The multiples of a point P, called generator, are finite and represent a cyclic subgroup for P. For an ECC algorithm we need subgroups with a high order. In fact there is a theorem that tells us how big is the cardinality of an elliptic curve E, the Hasse's theorem. According to this theorem, given an elliptic curve $E \ mod \ p$, the number of points on the curve ($\#E$) is bounded by: $p+1-2\sqrt{p} \leq \#E \leq p+1+2\sqrt{p}$. That means that if the EC is choosen carefully, the best known algorithm for computing the ECDLP (Elliptic Curve Discrete Logarithm Problem) requires $\approx \sqrt{p}$ steps. If p is described by 160 bits, $\#E$ is a number of 80 bits.

Given this characteristics we can define a cryptographic scheme in which the private key is the random integer d, while the public key is the endpoint of the 'jumps', T, so a point of the curve. That because, even if we know P and T, finding the private key d is hard.

3 Elliptic Curve Diffie-Hellman key exchange

ECDH is a straightforward variant of Diffie-Hellman algorithm for elliptic curves, so it is a key-agreement protocol. To explain how it works we defined as usual: two parties (Alice and Bob), who wants to exchange informations, and an attacker (Oscar), who wants to decode their messages. The phases of the protocol are:

1. Set-up of the elliptic curve and the selection of the primitive element;

2. Protocol: both Alice and Bob generate their private and public keys: a=k_{prA}=2,3,...,E-1 and A=k_{pubA}=a*P=(x_A, y_A) for Alice and b=k_{prB}= 2,3,...,E-1 and B=k_{pubB}=b*P=(x_B, y_B) fro Bob. The two parties then exchange their public keys A and B over the insecure channel. Alice computes T=a*B =(x_{AB}, y_{AB}) and Bob similarly T=b*A=(x_{AB}, y_{AB}).

T is called shared secret. The attacker knows only the public keys but he can't compute the shared secret or the private keys. Now that Alice and Bob have obtained the shared secret, they can exchange data with a symmetric encryption such as AES.

4 Conclusions

ECC is considered to be more secure compared to RSA and probably it is going to be the next generation for encryption methods. This doesn't mean that ECC is absolutely secure and that it doesn't have flaws, for example quantum computing is considered as a real treat for ECC (more than it is for RSA), but the creation of a quantum computer able to break it is still not near.

References

[1] https://www.youtube.com/watch?v=vnpZXJL6QCQ

[2] https://www.youtube.com/watch?v=zTt4gvuQ6sY&t=2s&ab_channel=IntroductiontoCryptographybyChristofPaar

[3] https://en.wikipedia.org/wiki/Elliptic-curve_cryptography

[4] https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman