Initial Project Report:
Aspects of Elliptic Curve Cryptography

Blake Marcus Wanier

Advisor: Professor George Nakos

January 25, 2007

Introduction

Since the beginning of society, there has been a need for information to be passed swiftly and securely. To solve this problem people searched for ways to mask what they were communicating. Cryptology is the composition of cryptography and cryptanalysis, or the making and breaking of codes. One of the earliest uses of cryptography was in the Roman Empire where they simply shifted the alphabet, so that each letter stood for another letter a certain number of positions down the alphabet. While it would not be a secure cipher today, it was adequate for the day. In this project we outline the basic theory of a modern cipher by analyzing elliptic curve cryptography, and eventually we will study and implement Rene Schoof's algorithm [SE] which counts the number of points of an elliptic curve over a finite field.

A serious problem that arises from creating secure cryptosystems is the ability to communicate and manage the encryption and decryption keys. When dealing with symmetric ciphers, the decryption key can be easily derived when the encryption key is known. So the problem arises when trying to establish a key when there is no secure communication already established between the entities that want to interact. Up until recently, there were no asymmetric cryptosystems, in other words, there were no systems in which knowing the encryption key did not allow easy access to the decryption key. The ability to publish a key for which others could use to encrypt without giving away the decryption key solved the problem of communicating keys that are used for symmetric ciphers. Now it is possible to exchange keys for secure symmetric ciphers by using an asymmetric, or public key, cipher. Currently, methods in creating public encryption keys that are resistant to attack are based on mathematical problems that are believed to be computationally hard. Two such mathematical problems are the factorization of integers and the discrete logarithm problem. We will analyze the discrete logarithm problem further before analyzing several public key cryptosystems.

Goals

In this project we hope to accomplish several things, the first being an analysis of the discrete logarithm problem and factoring, and how it influences public key cryptography. From there we plan to move into elliptic curve cryptography, analyzing how the discrete logarithm problem changes, and the different problems that arise. Finally we hope to analyze Schoof's Algorithm, and discuss how it impacts elliptic curve cryptography.

Discrete Logarithm Problem

The discrete logarithm problem comes from the difficulty in finding the power to which to raise a generator of a cyclic group in order to find a specific element of that group. We begin by defining the group and a generator of that group:

Let n be a positive integer for which the group of invertible elements modulo n, $\mathbf{Z}_n^*$, is cyclic. Gauss proved that in this case

n= 2, 4, p^l, $2p^l$, where l=1, 2, …, and p is an odd prime

Let α be a generator of the cyclic group $\mathbf{Z}_n^*$:
$\langle \alpha \rangle = \mathbf{Z}_n^*$.

then we choose which element we wish to find the discrete logarithm of,

Let β be an element of $\mathbf{Z}_n^*$.

Then there is a unique a such that

$\alpha^a = \beta$ in $\mathbf{Z}_n^*$ where $1 \le a \le n-1$.

This a is the discrete logarithm of β to the base α or,

$a = \log_\alpha \beta$.

The discrete logarithm problem(DLP) is written as such:

Given a prime p, a generator α of $\mathbf{Z}_n^*$ and an element $\beta \in \mathbf{Z}_n^*$, find a such that
$\alpha^a \equiv \beta \bmod p$ where $0 \le a \le p-2$.

For small p it is possible to do an exhaustive search in a short time span, but as p grows large, then the difficulty of finding a is exponentially harder. It is the difficulty of finding a which provides the security for several modern asymmetric cryptosystems including the El Gamal cryptosystem.

The El Gamal Cryptosystem

Let p be a prime such that the DLP for $\mathbf{Z}_n^*$ is infeasible. Let α be a generator of $\mathbf{Z}_p^*$. Let a be an integer.

$K = \{(p, a, \alpha, \beta): \quad \alpha^a = \beta \bmod p\}$

In the cryptosystem (p, α, β) is the public key and (a) is the private key.
The encrypter chooses a random number $k \in \mathbf{Z}_{p-1}$.

Encryption consists of

$$\varepsilon_k(x,k) = (y_1, y_2),$$
$$y_1 = \alpha^k \bmod p,$$
$$y_2 = x\beta^k \bmod p,$$

where x is the message, y_1 is the header and y_2 the encrypted message.

Decryption consists of

$$d_k(y_1, y_2) = y_2 \left(y_1{}^a\right)^{-1} \bmod p.$$

this works since

$$y_1 = \alpha^k \bmod p,$$
$$y_2 = x\beta^k \bmod p,$$
$$d_k(y_1, y_2) = y_2 \left(y_1{}^a\right)^{-1} \bmod p = x\beta^k \left(\alpha^{k^a}\right)^{-1} \bmod p = x\alpha^{ak}\left(\alpha^{ka}\right)^{-1} \bmod p = x \bmod p\,.$$

Elliptic Curves

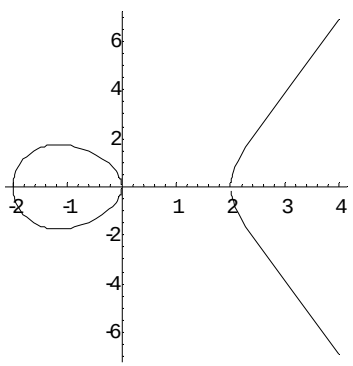
One of the difficulties of previous public key cryptosystems is the ability to find unique groups which can be used for encryption. While elliptic curves are a well studied area of mathematics, it was not until the mid 1980s when V. Mitler [Mi] and N Kolbitz [Ka], working independently found a solution for this problem. Elliptic curves are solution sets for certain polynomials, and can be defined over $\mathbf{Z}_p$. While not a recent discovery, their use in cryptology have proved to be quite useful. For simplicity we will analyze elliptic curves over the **R**, the set of real numbers, first.

A non singular elliptic curve is the set of solutions $(x, y) \in \mathbf{R}x\mathbf{R}$ to the following equations:

Let $a, b \in \mathbf{R}$ such that $4a^3 + 27b^2 \neq 0$.

$$y^2 = x^3 + ax + b$$

and the point O, also called the point at infinity.



Now that we have defined what an elliptic curve *E* is, we need to define an to make *E* an abelian group. The identity element of *E* is defined as *O*.

Let $P, Q \in E$, where $P = (x_1, y_1), \quad Q = (x_2, y_2)$.

We then have three cases which must be considered

$1)\, x_1 \neq x_2$
$2)\, x_1 = x_2 \quad y_1 = -y_2$
$3)\, x_1 = x_2 \quad y_1 = y_2$.

In the first case we look at the line *L* which intersects *E* at the two points *P* and *Q*. It is clear that *L* also intersects *E* at a third point R'. We define *P*+*Q*=*R*, where *R* is the reflection of R' about the x-axis.

The elliptic curve would not be very useful if we could not define *R* algebraically as well as graphically, so in order to do so we begin by analyzing the equation for *L*.

$$y = \lambda x + v\,,$$

where

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

is the slope and

$$v = y_1 - \lambda x_1 = y_2 - \lambda x_2\,.$$

We then solve for the intersections of *L* with *E* by substituting $y = \lambda x + v$ into the equation for *E*

$$y^2 = x^3 + ax + b$$
$$x^3 - \lambda^2 x^2 + (a - 2\lambda v)x + b - v2 = 0$$

When we solve for x in this equation we will get the three x-coordinates of $E \cap L$, and since we already know two of them are real from *P* and *Q*, we know the third root is real as well. Furthermore we know that the sum of the three roots must be equal to

$$-(-\lambda^2) = \lambda^2,$$
$$x_1 + x_2 + x_3 = \lambda^2$$
$$x_3 = \lambda^2 - x_1 - x_2$$

Now that we have solved for the x-coordinate of R', we let y_3 be the y-coordinate of R', and compute it by using λ.

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} = \frac{-y_3 - y_1}{x_3 - x_1}$$
$$y_3 = \lambda(x_1 - x_3) - y_1$$

We now have a simple formula to solve for $R = (x_3, y_3)$, where

$$x_3 = \lambda^2 - x_1 - x_2 ,$$
$$y_3 = \lambda\left(x_1 - x_3\right) - y_1 ,$$
$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} .$$

For the second case, we simply define

$(x, y) + (x, -y) = O$, the point at infinity.

In the third case we are adding *P* to itself and assume that $y_1 \neq 0$. In this case we need to define *L* to be the line tangent to *E* at *P*. While most of the analysis is Identical to case 1, the slope needs to be calculated through implicit differentiation

$$y^2 = x^3 + ax + b$$
$$2y\frac{dy}{dx} = 3x^2 + a$$
$$\lambda = \frac{dy}{dx} = \frac{3x^2 + a}{2y}$$

therefore in the case of $P = (x_1, y_1)$

$$x_3 = \lambda^2 - x_1 - x_2 ,$$
$$y_3 = \lambda(x_1 - x_3) - y_1 ,$$
$$\lambda = \frac{3x_1^2 + a}{2y_1}.$$

By combining the three cases we can summarize addition to:

Let $P = (x_1, y_1)$ then $-P = (x_1, -y_1)$. If $Q = (x_2, y_2)$ and $Q \neq -P$, then $P + Q = (x_3, y_3)$, where

$$x_3 = \lambda^2 - x_1 - x_2 ,$$
$$y_3 = \lambda(x_1 - x_3) - y_1 ,$$
$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, \text{if } \mathrm{P} \neq \mathrm{Q} \\ \dfrac{3x_1^2 + a}{2y_1}, \text{if } \mathrm{P} = \mathrm{Q} \end{cases}.$$

Elliptic Curve Cryptography

Elliptic curves are used for cryptography because of the difficulty of the elliptic curve DLP. While generic algorithms apply, the index calculus algorithm has no adaptation, effectively eliminating one of our most powerful tools. Before we can look more in depth at the cryptosystems, we need to modify the addition on elliptic curves for $\mathbf{Z}_p$.

Let $P=(x_1,y_1)$ and $Q=(x_2,y_2)$ if $x_1=x_2$ and $y_1=-y_2$ then $P+Q=O$ otherwise $P+Q=(x_3,y_3)$, where

$$x_3=\lambda^2-x_1-x_2,$$
$$y_3=\lambda(x_1-x_3)-y_1,$$
$$\lambda=\begin{cases}(y_2-y_1)(x_2-x_1)^{-1}, \text{if } \mathrm{P}\neq\mathrm{Q}\\ (3x_1^2+a)(2y_1)^{-1}, \text{if } \mathrm{P}=\mathrm{Q}\end{cases}.$$

and $P+O=O+P=P$ for all $P\in E$.

If E is small enough, it is possible to calculate all of the elements of E using a brute force method. After we determine the elliptic curve, we need to calculate the order. The brute force method is extremely slow when p is a large prime, and there are several theorems which allow us to approximate the order, but our most powerful tool for doing this is Schoof's algorithm. Schoof's algorithm computes $|E|$ with a running time of $\mathrm{O}(\log p)^8$, and is efficient for primes p up to several hundred digits. After we have determined $|E|$ we can easily see if the elliptic curve is cyclic, because if the order of a group is prime, then the group is cyclic.

Now that we have established that $\mathbf{Z}_p$ is a group, and we can determine relatively easily if it is a cyclic group, we can look at the El Gamal cryptosystem, which translates nicely to the elliptic curve.

The first thing we need to do is to change the encryption and decryption from multiplicative to additive notation.

Let α be a generator of the cyclic elliptic curve E, and let a be the private key.

$$\beta=a\alpha$$

Given a plain text to be encrypted x, and a secret number k $0<k\leq|E|-1$, encryption is as follows

$$\varepsilon_k(x,k)=(k\alpha,\ x+k\beta)$$

and decryption is

$$d_k(y_1, y_2) = y_2 - ay_1 .$$

It is important to note that *x* must be an element of *E*, and the encoding of *x* onto *E* is not trivial when E is over $\mathbf{Z}_p$.

References

[Ga] Garret, Paul “Making, Breaking Codes: Introduction to Cryptology” Prentice Hall, 2000.

[Ko] N. Koblitz, *Elliptic curve cryptosystems*, in *Mathematics of Computation* 48, 1987, pp. 203–209.

[Mi] V. Miller, *Use of elliptic curves in cryptography*, CRYPTO 85, 1985.

[Sc] Schoof, René(I-ROME2) Counting points on elliptic curves over finite fields. (English summary) Les Dix-huitièmes Journées Arithmétiques (Bordeaux, 1993). J. Théor. Nombres Bordeaux 7 (1995), no. 1, 219--254.

[St] Stinson, Douglas “Cryptography: Theory and Practice” 1956 Chapman & Hall/CRC, 2002.