Elliptic Curve Cryptography

Alexa Carr
CS 448A : Dr. Riley
Spring 2016

Contents

Figures

1 Abstract

In public key cryptography, elliptic curves are increasingly utilized because of their ability to provide maximal security with minimal key size. Elliptic curve cryptography is built upon the mathematical foundations of group theory. Despite intense levels of theoretical mathematical foundations, elliptic curves still contain imperfection. On the basis of a literature review, this report reviews the mathematical underpinnings of elliptic curves, the applications of elliptic curves in cryptography, and what makes some curves stronger or more useful than others.

2 Development of Public Key Cryptography

With the expansion of Public Key Cryptography, Elliptic Curve Cryptography may be one of the most secure systems presently available. In many cases it is replacing RSA as the encryption method of choice for key exchange and digital signatures. Rather than RSA's dependence upon mere multiplication and factorization of large prime numbers, ECC utilizes the algebraic structure of elliptic curves over finite fields, requiring expensive computation of the elliptic curve discrete logarithm function. Not only is ECC much more difficult to work with, but it achieves the same level of security as an RSA algorithm, with a much smaller key. According to CloadFlare, "...breaking a 228-bit RSA key requires less energy to than it takes to boil a teaspoon of water. Comparatively, breaking a 228-bit elliptic curve key requires enough energy to boil all the water on earth" [1]. Thus, if ECC was more widespread and industry supported, it could be an extremely viable replacement to the more taxing public key systems, e.g. RSA. Another representation of relative key sizes which provide the same amount of security can be seen in Figure 1:

Figure 1: Comparing Key Sizes

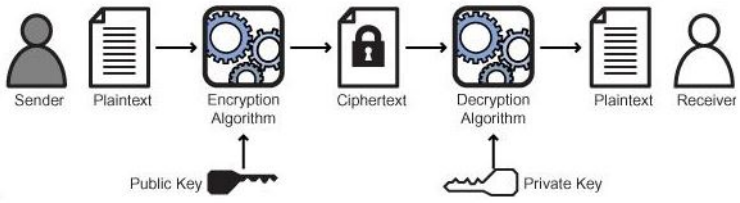
NIST guidelines for public key sizes for AES

ECC KEY SIZE (Bits)	RSA KEY SIZE (Bits)	KEY SIZE RATIO	AES KEY SIZE (Bits)
163	1024	1 : 6	
256	3072	1 : 12	128
384	7680	1 : 20	192
512	15 360	1 : 30	256

Asymmetric (public key) cryptography relies on a pair of keys which are mathematically interdependent. The system of public key cryptography is an advance from previous symmetric key cryptography systems because it does not rely on the prior exchange of a single encryption/decryption pair, during which

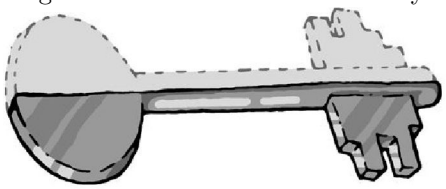
there is greater potential for key interference and compromised security. [2] The process of public key cryptography is illustrated in Figure 2.

Figure 2: Public Key Cryptography



In public key cryptography, Alice and Bob are assigned their pairs of keys from some certificate authority. Alice has a public key which is known to the world and can be used to encrypt a message being sent to her. Her private key is mathematically dependent upon her public key in an extremely computationally expensive way, so that it is infeasible to know Alice's private key without being Alice herself. When Bob sends Alice a message encrypted with Alice's public key, only Alice can decrypt it using her private key. Likewise, Alice can encrypt a message with Bob's public key, send it to Bob, and Bob can decrypt it using his private key.

Figure 3: Distinct but Related Keys



It is important to note that the key used for encryption is different than that used for decryption, though the encryption and decryption keys are still mathematically related. This is the primary distinction from symmetric key cryptography, which uses the same key for encryption and decryption. Multiple different systems of asymmetric cryptography have been developed since its

beginning in the 1970s.

2.1 Diffie-Hellman

The first version of public key cryptography introduced in 1975 was the Diffie-Hellman system intended for use as a military secret system [2]. The Diffie-Hellman system problem is founded on the computationally difficult discrete logarithm problem: for a group G, $g \in G$, and h in the subgroup generated by g, find the integer m such that:

$$h = g^m$$

To make a system for message exchange from this discrete logarithm problem, the group G, and element g of order n is made known to the public. Bob has secret key $b < n$ and Alice has secret key $a < n$. Then Bob's message $B = g^b$ and Alice's message $A = g^a$. Bob sends B to A who can then compute $B^a = g^{ba}$ and Alice sends A to Bob who can then compute $A^b = g^{ab}$ [3]. Accordingly, Alice and Bob now share the same information without knowing each others' private keys a, b:

$$B^a = g^{ba} = g^{ab} = A^b$$

2.2 RSA

In 1977, the joint efforts of Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman produced the public key crypto-system known as RSA for confidentiality and digital signatures based on the computationally difficulty of factoring large numbers.

To establish the public key, two numbers must be found, e and n. The modulus n is generated by the multiplication of two, large, relatively prime numbers p and q. Find a number e which is relatively prime to $\Phi(n) = (p -$

$1)(q-1)$such that $1 < e < \Phi(n)$. The private key d is the inverse of e,or the d such that $ed = 1 \bmod(\Phi(n))$. The security of RSA is founded on the difficulty of finding the two prime factors p and q of n [4].

If Alice is sending a message to Bob using RSA, Bob and Alice both have knowledge of the public key (n, e). The message Alice is sending will be numerically represented with numbers less than n. With P, a section of the numeric plain text message, Alice calculates the cipher text C as:

$$C = P^e mod n$$

When Bob receives this message, he uses his private key d, calculates C^d, and retrieves the plain text P.

2.3 Depth of Trapdoors

The above described asymmetric cryptography systems exemplify the workings of public key crypto-systems to aid in the understanding of Elliptic Curve's Cryptography's relevance to other systems of its type. Elliptic Curve Cryptography is a variation of the discrete logarithm problem that uses a geometric group rather than a modular group. This allows the keys involved in the Elliptic Curve Crypto-system to be stored with less memory, thus making the keys easier to manage and compute[4].

Essential to all such public key crypto-systems is their classification as "Trapdoor Functions." That is, they are easy to use in one direction, but much more difficult in the other. For example, with the Diffie-Hellman system, it is easy to raise a shared message by a private number, but much more difficult to then take this exponential result and from it, determine which private number it was raised to, the knowledge of which would need to be had if one was to decipher a message. For RSA, it is easy to start with two prime numbers and then

multiply them to establish a field, but much more difficult to determine which two prime numbers were multiplied given the one large product n. However, as previously illustrated with the boiling water analogy, these two systems, though mathematically intensive, still have their limitations. They both require much more memory than do elliptic curve systems. Additionally, algorithms such as the Quadratic Sieve and the General Number Field Sieve have been developed in response to the problem of factoring large prime numbers [1].

Elliptic curves serve as a trap door with an even deeper falling distance. The crypto-system of an elliptic curve relies upon its symmetry. The 'addition' of two points along the curve results in a slope of intersecting or tangential points to give a third point which is then reflected over the x-axis and remains on the curve. This reflected point can then be combined with another point to obtain another point on the curve which is again reflected over the x-axis. This process can be repeated however many (privately known) times to arrive at a final point which is the cipher text [1]. Just as it is easy to multiply large primes in RSA but difficult to determine which two large primes were multiplied, it is easy to keep connecting and reflecting points along the Elliptic Curve but hard to determine how many times the points where reflected, which is necessary knowledge to recover the original plain text information.

3 Applications

In general, Public Key Cryptography is used in key exchange for integrity, signature schemes for authenticity, and encryption for confidentiality. Such cryptosystems include RSA, Diffie-Hellman, ECDH, DSA, and ECDSA.

Elliptic curves have been increasingly applied in a number of settings since their introduction in 1985 [5]. Elliptic Curves are advantageous because of the reduced key size required to provide the same level of security as relevant counterparts, such as RSA. Applications of Elliptic Curves include government communications, bitcoins, signatures of iMessages, encryption of DNS information, and browser authentication. Typically the ECC variation of an RSA or Diffie Hellman key exchange mechanisms or elliptic curve - based certificates operate much quicker and with less memory. A 256-bit Elliptic Curve Digital Signature Algorithm key is more than 20 times faster than a 2,048-bit RSA key [1].

U.S. standards in place to govern Elliptic Curve Cryptography include TLS for secure browsers, S/MIME, CMS for secure email, IPSec, X509 certificates, FIPS, Digital Signature Standard for NIST [5].

Elliptic Curve Cryptography involves a combination of elliptic curve methods with other cryptographic methods. For example, Elliptic Curve Diffie Hellman (ECDH) is often used for key exchange and Elliptic Curve Digital Signature Algorithm for digital signatures.

4 Elliptic Curves

Elliptic curves can be represented by the following equation:

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

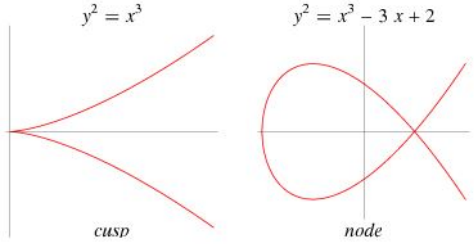
The points described by this Weierstrass form of the elliptic curve equation together with a point at infinity, denoted 0 comprise an elliptic curve. When graphed, elliptic curves are symmetric about the x-axis. The different variations of the curves seen in Figure 4 are the result of varied a and b values.

Figure 4: Elliptic Curves in Weierstrass Form

For the curve to always be defined, $4a^3 + 27b^2 \neq 0$. This restriction excludes the case that there will be a point or an intersection in the left looping portion of the graphical curve.

Including the point at infinity yields the following definition of elliptic curves:

Figure 5: Singular Curves



$$\{(x, y) \in R^2 \mid y^2 = x^3 + ax + b, 4a^3 + 27b^2 \neq 0\} \cup \{0\}$$

Elliptic curves are useful because the group that can be defined over them. For a further definition of the group structure of Elliptic Curves see Mathematical Foundations. In terms of the elliptic curve, the group operation of addition is defined as follows for three, aligned nonzero points in any order:

$$P + Q + R = 0$$

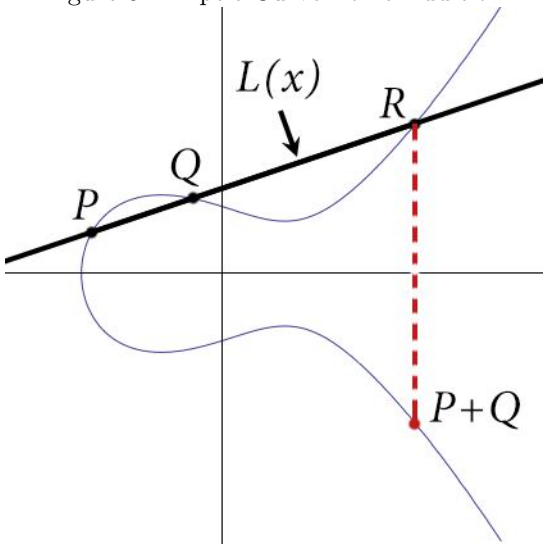
The identity element $e = 0$, the point at infinity. The inverse of a point P is the one symmetric about the x axis. This is an associative and commutative operation, making this an abelian group. Because of such:

$$P + Q + R = 0 \;\Rightarrow\; P + Q = -R$$

This relationship can be easily seen in the graph of an elliptic curve. The addition of points depicted in Figure 6 is the foundation for elliptic curve operations and elliptic curve cryptography.

Given two non-zero, points $P = (x_p, y_p)$ and $Q = (x_q, y_q)$ the sum of $P + Q$

Figure 6: Elliptic Curve Point Addition



is a geometric operation of connecting two points P and Q by a line l. This involves finding the slope of l that intersects both P and Q, and the third point R that is the intersection of l with the elliptic curve E at another point distinct from P and Q. Algebraically, this sum can be computed by first finding the slope of the the line l that intersects the first two point P and Q on E. Once the slope m is found it is used to formulate l. Then the third intersection of l with E will be the point R, whose reflection across the x axis, $-R$ will be used as the P of the next iteration. First, to find the slope m:

(a) If $x_P = x_Q$, and $y_P \neq y_Q$ or $y_P = y_Q = 0$ then $P + Q = 0$, where 0 is the identity element and point at infinity, so $P = -Q$. Geometrically, this is the case that P and Q are either directly symmetric each other about the x axis or else both on the x axis. Either way, the line l is a vertical line and there is no third point of intersection R.

(b) Else if $x_P \neq x_Q$, then there is a positive real distance between P and Q,

so the slope m of l between P and Q is:

$$m = \frac{y_Q - y_p}{x_Q - x_P}$$

(c) Else if $x_P = x_Q$ and $y_P = y_Q \neq 0$, then P and Q are the same point on the curve E, so the the tangency point of l at P and Q is:

$$m = \frac{3x_P^2 + a}{2y_P}$$

Then using this slope, the point $R = (x_R, y_R)$ is determined as follows:

$$x_R = m^2 - x_P - x_Q$$

$$y_R = y_P + m(x_R - x_Q) = y_Q = m(x_R - x_Q)$$

Then because $P + Q = -R$,

$$(x_P, y_P) + (x_Q, y_Q) = (x_R, -y_R)$$

Once the sum $P+Q$ is found, then $P+Q$ is again added to another point on E to find a new sum. Considering in particular the third case where $P = Q$ and it is the point of tangency that determines the line l, this process of repeatedly adding point on E to themselves to find new points on E can be summarized by

$$mP = P + P + \cdots + P$$

where m is the number of times P is added to itself [3]. In this process, it is easy to move in the forward direction and continue adding P to itself m times to arrive at some final point $R \in E$, but it is very difficult to be given R and

from it derive the original m to recover the first P. This backwards process of elliptic curves is even more difficult than factoring the product of large prime numbers, and as such, can provide more security with less bits.

This variation of elliptic curves has been discussed in Weierstrass form. Other forms of elliptic curves, can be shown to be equivalent to the Weierstrauss form.

5 Variations of Elliptic Curves

Alternative representations of elliptic curves include Hessian curves, Edwards curves, Twisted curves, Jacobian curves, and Montgomery curves. Each has benefits to particular situations, i.e. each can apply the group operation of its particular format, thus saving time and memory. In particular, Curve 25518 is considered here, as it has been increasingly applied of late.

Of the next three forms, each has its own advantages related to computational ease and use of the group law formula. Three particular advantages can be considered: prime order, completeness, and the ADD function. These advantages can be summarized as follows [6]:

Figure 7: Comparing Features of of Curve Forms

curve model	prime order possible	simple completeness	supports ADD function
Weierstrass	✓	✗	✓
twisted Edwards	✗	✓	✓
Montgomery x-only	✗	✓	✗

Prime order is useful because it is backwards compatible and supports more popular standardized curves. Completeness - use of a group law formula that is compatible with all possible inputs - is useful because it makes implementation more compact and allows constant time operations. The ADD function is useful because it provides the foundation for ECDSA signature verification.

5.1 Weierstrass

Weierstrass models can be used to define curves over large prime fields. Weierstrass curves are considered for their advantages in allowing prime ordered Fields and supporting the ADD function. As seen before, the Weiserstrass form of elliptic curves requires the choice of a and b such that $4a^3 + 27b^2 \neq 0$ in the following equation.

$$W_{a,b} : y^2 = x^3 + ax + b$$

Contrastingly, Edwards and Montgomery Curves cannot be used for prime order curves.

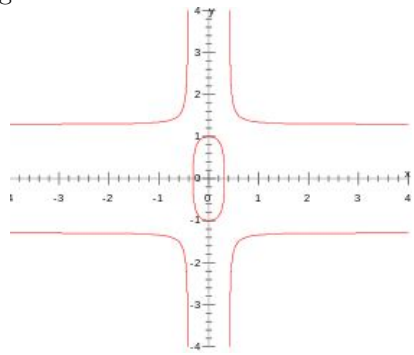
5.2 Edwards Curves

Twisted Edwards curves are considered for their advantages in providing simple completeness and supporting the ADD function. In Edwards Curves, the parameters needing to be chosen are a and d such that a, d are distinct non-zero elements the Field over which the curve is defined.

$$E_{a,d} : ax^2 + y^2 = 1 + dx^2y^2$$

Twisted Edwards curves have the following geometric form:

Figure 8: Twisted Edward's Curve Form



By this curve form, the same kind of operations can be performed, where all are supported by a geometric group.

5.3 Montgomery Curves

Montgomery curve are considered for their advantages in providing simple completeness. For Montgomery Curves, parameters needing to be chosen are A and B which are elements of the Field over which the curve is defined. Also, $B(A^2 - 4) \neq 0)$

$$M_{A,B} : By^2 = x^3 + Ax + x$$

Across varying security levels of 128-bits, 192-bits, and 256-bits, performance of the twisted Edwards and Montgomery models are very similar, and the fastest among them is 1.20 times faster than the Weierstrass model. [6]

Overall, Weierstrass curves over prime fields offer backwards compatibility but operate slower, whereas twisted Edwards curves over composite fields offer a time advantage.

6 Mathematical Foundations of Elliptic Curves

6.1 Group Structure

The definition of a group over an elliptic curve is the foundation of elliptic curves cryptography because the identity element and the inverse of all elements of groups are unique. This uniqueness is what provides the security of elliptic curves. Groups must have an associated binary operation that satisfies the properties of closure, associativity, existence of an identity element, an inverse for every element in the group. Such an operation is called addition and denoted as such "+". If the addition is also commutative, then the group is Abelian [7]. The properties of binary group operations are summarized as follows for a group G:

1. Closure: $\forall\ a, b \in G, a + b \in G$
2. Associativity: (a + b) + c = a + (b + c)
3. Identity: $\exists\ e\ :\ \forall\ a \in G,\ a + e = e + a = a$
4. Inverse: $\forall\ a \in G\ \exists\ a^{-1}\ :\ a + a^{-1} = e$
5. (Abelian Groups) Commutativity: $\forall\ a, b \in G,\ a + b = b + a$

The definition of an elliptic curve over a group has the identity 0 which is the point at infinity. Because of an elliptic curve's symmetric geometry, for all points P on the elliptic curve E there exists an inverse $-P$ which is the reflection of P over the x-axis. The group operation is that of 'adding' points on the curve to themselves to obtain a line which intersects the curve in another location and is reflected over the x-axis to obtain another point for addition.

By applying the group operation n times, the process of elliptic curve cryptography can be represented as $Q = kP$. If P and Q are known, the problem of

finding k is the computationally difficult discrete logarithm problem for elliptic curves [7].

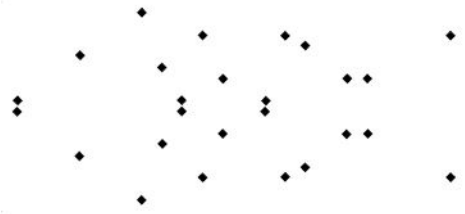
6.2 Subgroups of Finite Fields

Elliptic curves are most often restricted to finite fields, a set of finitely numbered elements of prime order, denoted F_p. Both addition and multiplication are closed over fields with unique identity elements and unique inverse elements. Rather than being a continuous curve of real elements, elliptic curves over finite fields comprise a set of disjoint points.

Because addition of multiples of P results in a multiple of P, the multiples of P form a cyclic subgroup of the group of the elliptic curve. As such, a point P is a generator of its subgroup, because addition of P to its multiples k-number of times will produce all the elements of the cyclic subgroup of order k. The order of a subgroup or of its generator P is the smallest possible k such that $kP = 0$. It is important to note that by Lagrange's theorem, the order k of a subgroup divides the order q of the elliptic curve group with q elements. The co-factor of the subgroup is the integer $h = q/k$. Accordingly, $k(hP) = k((\frac{q}{k})P) = qP = 0$ [7].

Because elliptic curves are defined over finite fields, it is important to note that points of an elliptic curve do not comprise a smooth curve but rather a set of discrete points.

Figure 9: Discrete Points of Elliptic Curves over Finite Fields



As seen in Figure 9, the discrete points of an elliptic curve maintain the symmetry of the curve. Accordingly while images of smooth curves with infinitely many points do aid in the understanding of the geometrically-founded group operations of elliptic curves, in reality the curves are defined over fields of finitely many points.

6.3 Elliptic Curves defined over Fields

Most elliptic curves are defined over prime fields, particularly when used for Internet security. Particular prime shapes allow faster modular arithmetic. Such a shape is the Edwards curve developed in 2007, or its generalized form, the twisted Edwards curve. Highlighted properties of this curve include its efficient arithmetic and compatibility with the Montgomery model. One drawback of the twisted Edwards curves is its inability to be of prime order of rational points in the base field, making it incompatible with the commonly used prime-order Weierstrass curves. Such consideration of prime order are key elements considered when determining curve security.

When defined over prime fields F_p with $p > 3$, the order of the field contributes to the modular arithmetic used in the group law where points are evaluated p. Different kinds of primes contribute to different levels of efficiency and security because of certain primes ability to be reduced via modular reduction.

Pseudo-Mersenne primes are primes of the form $p - s^\alpha - \gamma$, where α is a security parameter and multiple of 64, γ is an integer less than 2^32 [8]. In choosing fields over which an elliptic curve will be defined, it is advisable to choose one in which the group law can be computed efficiently. This limits the number of different fields, particularly to those with finite order q and a prime characteristic. If $k = F_q$ is the field over which the elliptic curve E is defined, then Hasse's theorem estimates the cardinality of the elliptic curve group E_k to

be near the order k:

$$|q + 1 - |E_{F_q}|| \leq 2\sqrt{q}$$

The discrete logarithm problem involved solving for n in the equation $\beta = \alpha^n p$. In regards to an elliptic curve, this n represents the number of times the group law is applied to some given starting point P, which would be the α used in the description of the discrete logarithm problem. For cyclic group $G =< \alpha >$ with cardinality p, the difficulty of the solving the discrete logarithm problem varies according to the representation of the group. For any generator α of an integer field Z_n, solving the discrete logarithm problem with the Euclidean Algorithm is relatively easy [9].

Accordingly, it is the order of the group E_k that contributes to the difficulty of solving the elliptic curve discrete logarithm problem, and thus the security of the elliptic curve. For groups of large finite order, square root attacks which could provide a backdoor to groups of lesser order no longer suffice.

6.3.1 Variations of Galois Fields

Galois fields, or finite fields are particularly well suited for cryptography. In particular, the GF fields that are most used include Prime fields $GF(p)$ with algebra modulo prime p and Binary Extension fields $GF(2^m)$ with algebra modulo an irreducible polynomial $F(t)$.

For any Galois Field, there are levels of computational difficulty for different types of operations. Easiest operations include addition, multiplication, and inversion. Slightly more difficult operations include Point Add and Point Double. More difficult is point multiplication. Finally most difficult operations involve elliptic curve protocol such as ECD

Prime fields of order p have $p - 1$ integer elements and support basic operations such as addition, subtraction, multiplication, division, and inversion. They

support algorithms including reduction technique such as Reduced Radix and Montgomery, multiplication techniques such as Comba multipliers and sometimes Karatsuba, and inversions such as Euclid's Algorithm. Because of their functionality with these operations and algorithms involving optimized integer arithmetic, Prime Galois Fields are favorably used for implementing software. The larger the prime field is, the more difficult it becomes for standard computers to handle, and points have to be represented by multiple words. To abate this problem, Prime Extension fields of the form $GF(p^q)$ can instead be used. Prime extension fields remove the need for elements of a field to be divided among multiple words and carries propagated. Prime Extension Fields make possible fast inversion algorithms, but reduction can be more complicated than it is when performed over other fields.

Binary Extension fields have 2^m polynomial elements over $GF(2)$ and support the basic operations of addition, subtraction, multiplication, division, and inversion. Binary Extension fields are often used because they support binary finite field math. This involves addition modulo 2 which can be easily represented in hardware via XOR gates without the need for a carry propagation. They support almost inverse, double and add, Montgomery scalar multiplication, and Frobenius expansion.

Because of their applicability to different algorithms, prime and binary fields will be used for different platforms. In particular, prime fields are more applicable for classical computer use because of the relative ease of performing algebra in prime fields. [10]

Ten different curve have been recommended by NIST, including 5 prime fields F_p where p is 192, 224, 256, 384, and 521 bits, and 5 binary fields F_2^m where m is 163, 233, 283, 409, and 571.

7 Attacks

Despite the heightened security offered by the relatively small key sizes of elliptic curves, there are still a number of weaknesses involved in the use of Elliptic Curves. These include domain attacks, backdoors, quantum computing attacks, and side-channel attacks.

7.1 Domain Attacks

In particular, binary fields F_2^m where m is not prime should be avoided as they are susceptible to the Weil descent attack. Prime fields F_p where the order is itself p should be avoided because points can be mapped to the additive group of F_p.

7.2 Backdoors

Backdoors involved the deliberate inclusion of known solutions to a trapdoor function, so that information encrypted with the backdoored-function can be recovered by the party who has knowledge of the backdoor. Because of the unexpected release of memos which used Dual Elliptic Curve Deterministic Random Bit Generator (DualECDRBG), it has been speculated that the NSA influenced the inclusion of such a backdoor in the DualECDRBG [6].

A pseudorandom generator involves the choice of a known trapdoor with which one knows future and possibly past generator outputs. It has been shown that "backdoored PRGs are equivalent to public-key encryption schemes with pseudorandom ciphertexts". An example of psuedorandom sabotage is the "backdoored NIST Dual EC PRG," where a saboteur selects two elliptic curve points P and Q and given d can generate a prediction of future elements given the group structure of elliptic curves and the fact that $d = d \log_Q P$ [11].

7.3 Quantum Computing

Quantum computing involves the use of qubits rather than binary digit bits, making it possible to solve some mathematically intese problems more quickly than regular binary-based computers Because of the smaller key size of elliptic curves needed to provide the same level of security provided by the larger keys of RSA, attacks involving the use of Shor's Algorithm are better suited to elliptic curves than RSA.

7.4 Side-Channel

Side-channel attacks involve intentional misuse of physically implemented factors of cryptosystems. Technical information relayed by such factors as differential power analysis and timing analysis can be used as a method of attack. Such an attack can be contrasted with a brute force attack, where lesser technical strategy is involved. A fault attack, which is a variation of a side-channel attack, involves the introduction of conditions for which the code is unprepared to respond, in order to gain information about the internal state of the system. Use of smart cards are particularly prone to fault attacks.

8 Conclusion

Ultimately, the choice of many different factors contributes to the applicability of elliptic curves to different situations. These include the form of the curve equation, the type of field over which the curve is defined, the parameters involved in the equation, the key size of the crypto-system, and the system's having been guarded against different potential attacks. By making informed decisions regarding these different factors, elliptic curve cryptography can be maximized. In doing so, elliptic curves can continue to play an integral role in the growing realm of public key cryptography.

References

[1] Nick Sullivan. A (relatively easy to understand) primer on elliptic curve cryptography. *CloudFlare*, October 2013.

[2] Phil Zimmerman and Adam Back. *Introduction to Cryptography.* Network Associates, 1990-1999. excerpt of book for definition of public key crypto.

[3] Joseph H. Silverman. *An Introduction to the Theory of Elliptic Curves.* Brown University and NTRU Cryptosystems, Inc., July 2006. presentation on public key crypto and ECs for Diffie Hellman and geometric EC explanation.

[4] TutorialsPoint. Public key encryption. *Tutorials Point*, 2016.

[5] Kristin Lauter. *Elliptic Curve Cryptography and Applications.* Microsoft Research, July 2013. applications and governing standards.

[6] Craig Costello. *A brief discussion on selecting new elliptic curves.* Microsoft Research, June 2015. Details differences in Weierstrass and Edwards curves with implications for NIST algorithm generation. Suggests use of full length primes. In particular we want: Weierstrass and Edwards equations to return the smallest constant c such that $2^{2s} - c$ is prime and equivalent to 3 mod 4.

[7] Andrea Corbellini. Elliptic curve cryptography: a gentle introduction, 2015.

[8] Joppe W. Bos. *Selecting Elliptic Curves for Cryptography: An Efficiency and Security Analysis*, July 2008.

[9] Andreas Enge. *Elliptic Curves and Their Applications to Cryptography: An Introduction*, September 1999.

[10] Tom St Denis Dana Neustadter. *Elliptic Curves over Primary and Binary Fields in Cryptography.* Elliptic, 2008.

[11] Dodis Yevgeniy. A formal treatment of backdoored pseudorandom generators. *Lecture Notes in Computer Science*, 9056, April 2015.