INTRODUCTION

Definitions, Acronyms and Abbreviations.

Addition Rule: An *addition rule* describes the addition of two elliptic curve points P_1 and P_2 to produce a third elliptic curve point P_3.

Basis: A representation of the elements of the finite field F_2^m

Binary polynomial: A polynomial whose co-efficient are in the Field F_2

Elliptic Curve: An *elliptic curve* is a set of points specified by two parameters a_2 and a_6, which are the elements of Field F_q. If q = p is an odd prime, p>3, the Weierstrass equation defined by the curve of the form $y^2=x^3+a_2x+a_6$. If q is a power of 2 (so that the field is F_2^m), then the Weierstrass equation defined by the curve is of the form $y^2=x^3+a_2x^2+a_6$

Elliptic Curve key Pair: Given particular elliptic curve parameters, an elliptic curve key pair consists of an elliptic curve private key and the corresponding elliptic curve public key.

Elliptic Curve private key: Given particular elliptic parameters, an elliptic curve private key, d is a statistically unique and unpredictable integer in the interval [1,n-1], where n is the prime order of the base point *P*.

Elliptic Curve public key: Given particular elliptic curve parameters and an elliptic curve private key d, the corresponding elliptic curve public key, *Q*, is the elliptic curve *point Q=dP,* where P is the base point.

Elliptic Curve Point: If E is an elliptic curve defined over a field F_q, then the *elliptic curve point P* is either

- A pair of field elements (x_p, y_p) (where x_p, y_p(element of F_q)) such that the values x =x_p and y = y_p satisfy the equation defining E or
- A special point O called the *point at infinity*.

Irreducible Polynomial: A binary polynomial *f(x)* is *irreducible* if it does not factor as a product of two binary polynomials, each of degree less than the degree of *f(x).*

Non Super singular Elliptic Curve: An elliptic curve of the form $y^2+xy=x^3+a_2x^2+a_6$. The attacks on curves of this form are fully exponential in time.

Order of a curve: The *order of an elliptic curve* E defined over the field F_q is the number of points on E, including the point at infinity. This is denoted by $\#E(F_q)$.

Order of a point: The *order of a point P* is the smallest positive integer n such that n*P*=O, the point at infinity.

Polynomial basis: A type of basis where by the elements of the field F_2^m are represented by polynomials.

Private key: In an asymmetric key system, that key of the entity's key pair which is known only by the entity.

Public key: In an asymmetric key system, that key of the entity's key pair which is publicly known.

Super Singular Elliptic Curve: An elliptic curve of the form $y^2+y=x^3+a_4x+a_6$.The curves of this form are easy to cryptanalyze.

Scalar multiplication: If *k* is a positive integer, then *kP* denotes the point by adding together *k* copies of the point *P*. The process of computing *kP* from *P* and *k* is called *scalar multiplication*.

Trace function:If *c* is an element of F_2^m, the *trace* of *c* is $Tr(c)= c+ c^{(2\wedge1)}+ c^{(2\wedge2)}+. . . + c^{(2\wedge(m-1))}$. It's a mapping from F_2^m to F_2.

x-coordinate: The *x-coordinate* of an elliptic curve point, *P* = (x_p ,y_p), is x_p

y-coordinate: The y-*coordinate* of an elliptic curve point, *P* = (x_p ,y_p), is y_p

Symbols and Notations

a mod *n*: The unique remainder *r*, 0<=*r*<=*n*-1, when integer *a* is divided by *n.* For e.g., 23 mod 7 = 2.

E: an elliptic curve.

E(F_q): The set of all points on an elliptic curve *E* defined over F_q and including the point at infinity O.

#*E*(F_q): If E is defined over F_q , then #*E*(F_q) denotes the number of points on the curve (including the point at infinity O). #*E*(F_q) is called the order of the curve *E*.

$F_2{}^m$: The finite field containing ${}_2{}^m$ elements, where 'm' is a positive integer.

F_p: The finite field containing 'p' elements where 'p' is a prime.

F_q: The finite field containing q elements, q shall be an odd prime number (p) or a power of 2,(2^m).

Tr: The trace function.

Introduction to Finite Field Theory

A finite field is a set **F** with a multiplicative and additive operation that satisfies the follow rule- associativity and commutativity for both addition and multiplication, existence of an additive identity 0 and a multiplicative identity 1, additive inverses and multiplicative inverses for everything except 0. The Field can over set of real numbers **R**, field of complex numbers **C**, the field **Z**/*p***Z** of integers modulo a prime number *p*. By referring to the "Order" of an element we mean the least positive integer modulo *p* that gives 1.

Multiplicative generators of finite field are those elements in F_q^*, $q = p^f$ which have maximum order. It is seen that the order of any *a*(element of) F_q^* divides q-1.

Every finite field has a generator. If *g* is a multiplicative generator of F_q, then g^j is also a generator if and only if gcd(j,q-1)=1. In particular, there are phi(q-1) different generators in the multiplicative generators of F_q^* .

As an example, let us investigate generators of F_{19}^*.

Let = = denote the congruence symbol.

We check if 2 is a generator in the given prime field.

2^1= =2 mod 19

2^2= =4 mod 19

2^3= =8 mod 19

2^4= =16 mod 19

2^5= =13 mod 19

2^6= =7 mod 19

2^7= =14 mod 19

2^8= =9 mod 19

2^9= =18 mod 19

2^{10}= =17 mod 19

2^{11}= =15 mod 19

2^{12}= =11 mod 19

2^{13}= =3 mod 19

2^{14}= =6 mod 19

2^{15}= =12 mod 19

2^{16}= =5 mod 19

2^{17}= = 10 mod 19

2^{18}= =1 mod 19

We see it gives the sequence

2,4,8,16,13,7,14,9,18,17,15,11,3,6,12,5,10,1

We observe that the set contains all the elements of the prime field. It is also seen that 2 has maximum order and is hence a generator in the given prime field.

If we obtain one generator in the prime field, it is easy to find the other generators.

We observe

gcd(3,9-1)=1

gcd(5,9-1)=1

gcd(7,9-1)=1

Hence the other generators in F_9^* are

2^3 mod9 = 8

2^5mod 9 = 5

2^7 mod 9 = 2

If we take 3 and test if it is a generator in F_9^*

4^1 mod 9 = 4

4^2 mod 9 = 7

4^3 mod 9 = 1

4^4 mod 9 = 4

4^5 mod 9 = 7

4^6 mod 9 = 1

4^7 mod 9 = 4

4^8 mod 9 = 7

4^9 mod 9 = 1

We see that 4 has order 3.

The tests to find a generator in F_p, the IEEE P1363 standards specify the following method. We find the factorization of p-1

$p-1= q_1*q_2*\ldots*q_n$, the prime factors of p-1.

Then we find

$Order_1 = g^{(p-1)/q}{}_1$

$Order_2 = g^{(p-1)/q}{}_2$

$Order_3 = g^{(p-1)/q}{}_3$

$Order_4 = g^{(p-1)/q}{}_4$

$Order_5 = g^{(p-1)/q}{}_5$

. . . .

. . . .

$Order_n = g^{(p-1)/q}{}_n$

For (1<=i<=n), if any of $Order_i$ = 1, the given element is not a generator. We test another random element of F_p^* until we find a generator.

Since there are phi(p-1) distinct generators in F_p^* finding one won't be so hard.

Diffie-Hellman key exchange over F_p

Let Alice and Bob be two legitimate parties attempting secure communication. They agree upon a publicly known generator in F_q^*.

- Alice chooses a random integer *a* between 0 and q-1 which is her private key.
- Bob chooses a random integer *b* between 0 and q-1 which is his private key.
- Alice computes $A_{pub} = g^a \bmod p$ which is her public key and sends it to Bob.
- Bob computes $B_{pub} = g^b \bmod p$ which is his public key and sends it to Alice.
- Alice computes $A_{shared} = (B_{pub})^a = (g^b)^a \bmod p$ which is her shared secret.
- Bob computes $B_{shared} = (A_{pub})^b = (g^a)^b \bmod p$ which is his shared secret.

We illustrate the scenario using a table.

g = Preferably a generator in F_p^*

a= Random number generated by Alice, private key of Alice.

b= Random number generated by Bob, private key of Bob.

ALICE	BOB
g^a mod p →	→ g^a mod p
g^b mod p ←	← g^b mod p
$A_{shared} = (B_{pub})^a = (g^b)^a$ mod p	$B_{shared} = (A_{pub})^b = (g^a)^b$ mod p

Discrete Log Problem: If G is a finite group, b is an element in G and y is an element in G which is the power of b, then the discrete log problem of y to the base b is any integer x such that $b^x = y$

Diffie-Hellman Assumption: It is computationally infeasible to compute g^{ab} knowing only g^a and g^b.

Algorithm for binary exponentiation modulo *m*:

Let $n_0,n_1,. . . , n_{k-1}$ denote the binary digits of n, i.e. $n=n_0+ 2n_1 +. . .+2^{k-1}n_{k-1}$.{$n_j$=0 or 1;0<= j <= k-1)

Step1 :Set *a* = 1.

Step2: Compute $b_1=b^2$ mod *m*. If n_0=1 (*a*<-b) else *a* remains unchanged.

Step3: Compute $b_2={b_1}^2$ mod *m*. If n_1=1(multiply *a* by b_1 mod *m*) else keep *a* unchanged.

Step4: Compute $b_3={b_2}^2$ mod *m*. If n_2=1(multiply *a* by b_2 mod *m*) else keep *a* unchanged.

. . .

. . .

Step n: At the j^th step we have computed b_j= = $b^{(2^\wedge j)}$ mod *m*. If n_j=1(multiply *a* by b_j mod *m*), else keep *a* unchanged. After the (k-1)st step we have the desired result *a*= = b^n mod *m*.

An introduction to arithmetic in polynomial basis

Let $f(x) = x^m + f_{m-1} x^{m-1} +. . . + f_2x^2 +f_1x+f_0$, where f_i(element of) F_2 for i=0 to m-1) be an irreducible polynomial of degree m over F_2 , i.e., $f(x)$ cannot be factored into two polynomials over F_2, each of degree less than m. $f(x)$ is called the reduction polynomial. The finite field F_2^m is comprised of all polynomials over F_2 of degree less than m:
$F_2^m = \{a_{m-1} x^{m-1} +. . . + a_2x^2 +a_1x+a_0$, a_i(element of)$\{0,1\}\}$.
The elements of F_2^m, $a_{m-1} x^{m-1} +. . . + a_2x^2 +a_1x+a_0$,can be represented using the binary string (a_{m-1} …, a_2,a_1,a_0,). Thus, the elements of F_2^m can be represented by a set of binary strings of length m. The multiplicative identity element 1 is represented by the bit string (000. . .01), while the zero element is represented by the bit string (000. . . 00).

Field addition in polynomial basis:
(a_{m-1} …, a_2,a_1,a_0,) + (b_{m-1} …, b_2,b_1,b_0,) = (c_{m-1} …, c_2,c_1,c_0,), where $c_i = a_i$ (xor) b_i , i.e. the field addition is performed componentwise.

Field multiplication modulo an irreducible polynomial in polynomial basis:
(a_{m-1} …, a_2,a_1,a_0,) . (b_{m-1} …, b_2,b_1,b_0,) = (r_{m-1} …, r_2,r_1,r_0,), where the polynomial $r_{m-1} x^{m-1} +. . . + r_2x^2 +r_1x+r_0$ is the remainder when the polynomial (a_{m-1} …, a_2,a_1,a_0,) . (b_{m-1} …, b_2,b_1,b_0,) is divided by $f(x)$ over F_2. This method of representing F_2^m is called the *polynomial basis representation* and $\{1,x,x^2,. . . ,x^{m-1}\}$ is called a polynomial basis of F_2^m over F_2. Let $F^*{}_2^m$ denote the set of all non-zero elements in F_2^m . There exists at least one element g in F_2^m such that any non-zero element of F_2^m can be expressed as a power of g. Such an element g is called a generator of primitive element in F_2^m. The multiplicative inverse of $a=g^i$ (element of) $F^*{}_2^m$ is $a^{-1}=g(^{2\wedge m)-1-i}$ where 2^m represents 2^m.
E.g. The finite field F_2^4 using a polynomial basis representation.
Take $f(x) = x^4+x+1$ over F_2. Then the elements of F_2^4 are:
(0000)
(1000)
(0100)
(1100)
(0010)
(1010)
(0110)

(1110)

(0001)

(1001)

(0101)

(1101)

(0011)

(1011)

(0111)

(1111).

As examples of field arithmetic, we have:

(1101) + (1001) = (0100) and (1101).(1001) = (1111) since

$(x^3+x^2+1)(x^3+1) = x^6+x^5+x^2+1$

$=(x^4+x+1)(x^2+x)+(x^3+x^2+x+1)$

$=x^3+x^2+x+1 \bmod f(x)$.

I.e. x^3+x^2+x+1 is the remainder when $(x^3+x^2+1).(x^3+1)$ is divided by f(x). The multiplicative identity is (0001). $F^*{}_2{}^m$ can be generated by one element g = x. The powers of g are:

g^0 = (0001)

g^1 = (0010)

g^2 = (0100)

g^3 = (1000)

g^4 = (0011)

g^5 = (0110)

g^6 = (1100)

g^7 = (1011)

g^8 = (0101)

g^9 = (1010)

g^{10} = (0111)

g^{11} = (1110)

g^{12} = (1111)

g^{13} = (1101)

g^{14} = (1001).

Now let us consider Elliptic curves over $F_2{}^m$. We illustrate the working process by means of an example

Example: An elliptic curve over $F_2{}^4$. Consider the field $F_2{}^4$ generated by the root g = x of the irreducible polynomial:

f(x)=x^4+x+1.

The powers of g are:

g^0 = (0001)

g^1 = (0010)

g^2 = (0100)

g^3 = (1000)

g^4 = (0011)

g^5 = (0110)

g^6 = (1100)

g^7 = (1011)

g^8 = (0101)

g^9 = (1010)

g^{10} = (0111)

g^{11} = (1110)

g^{12} = (1111)

g^{13}= (1101)

g^{14} = (1001)

g^{15} = g^0 = (0001).

Consider the non super-singular elliptic curve over $F_2{}^4$ with the defining equation:

$y^2+xy=x^3+g^4x^2+1$.

Here, a=g^4 and b=1. The notation for this equation can be expressed as follows, since the multiplicative identity is (0001):

(0001) y^2+(0001)xy = (0001)x^3+(0011)x^2+(0001).

Then the solution over $F_2{}^4$ to the equation of the elliptic curve are:

(0,1)

(1,g^6)

(1,g^{13})

(g^3,g^8)

(g^3,g^{13})

(g^5,g^3)

(g^5,g^{11})

(g^6,g^8)

(g^6,g^{14})

(g^9,g^{10})

(g^9,g^{13})

(g^{10},g^1)

(g^{10},g^8)

$(g^{12},0)$

(g^{12},g^{12}) .

The group $E(F_{2^4})$ has 16 points (including the point at infinity O). The following are the examples of the group operation.

1. Let $P_1 = (x_1,y_1) = (g^6,g^8)$, $P_2 = (x_2,y_2) = (g^3,g^{13})$, $P_{1+}\ P_2 = (x_3, y_3)$. where:

$m=(y_1+y_2)/(x_1+x_2) = (g^8,g^{13})/ (g^6,g^3) =g$

$x_3=m^2+m+x_1+x_2+a = g^2+ g+ g^6+ g^3+ g^4 = 1$,

$y_3=m(x_1+x_3) +x_3+y_1 = g(g^6+1)+1+g^8 = g^{13}$.

2. If $2P_1 = (x_3, y_3)$, then:

$m = x_1+(y_{1/}\ x_1) = g^6+(g^8/g^6) = g^3$,

$x_3=m^2+m+a = g^6+ g^3+ g^4 = g^{10}$,

$y_3=x_1^2+(m+1)x_3 = g^{12}+(g^3+1)g^{10} = g^8$.

We can find

$mP = \underbrace{P + P + \ldots + P}_{\text{(m times)}}$ using the equation above.

An element P of any group is said to have order m if

$mP = \underbrace{P + P +\ldots+ P}_{\text{(m times)}} = O$

but m`P (not equal) O for all integers 1<= m`< m. If such an ‘m’ exists, P has finite order.

Elliptic curves over F_{p^f}

We are especially interested in non-super singular elliptic curves

$E (F_{2^m})$ defined by parameters a_2, $a_6 \in F_{2^m}$, a_6(not equal) 0, is the set of solutions (x,y), $x \in F_{2^m}$, $y \in F_{2^m}$, to the equation $y^2 + xy = x^3 + a_2x^2 +a_6$ of characteristic 2 with the point at infinity O. Let $q =p^f$, Hasse theorem tells us q + 1- 2*Sqrt (q) < = #E (F_q)<= q + 1+2 *Sqrt(q) , which ensures that the number of points on E is close to the field size, for a large field.

Let P(x_1, y_1) and Q(x_2, y_2) be two points on E. Then R(x_3, y_3)= P(x_1, y_1) + Q(x_2, y_2), where

$x_3=m^2+m+x_1+x_2+a_2$; $m=(y_1+y_2)/(x_1+x_2)$

$y_3=m(x_1+x_3) +x_3+y_1$; $m=(y_1+y_2)/(x_1+x_2)$

For double point, 2P

$2*(x_1, y_1) = (x_3, y_3)$

$x_3=m^2+m+a_2$; $m=x_1+(y_1/x_1)$

$y_3=x_1^2+(m+1)*x_3$; $m=x_1+(y_1/x_1)$

Analog of Diffie-Hellman key exchange over Elliptic Curve

Let E be the elliptic curve over $F_{q,}$ q =2^n being large. A reasonably secure implementation should have n>250 bits. Let Alice and Bob be two legitimate parties attempting secure communication. They choose a point B(element of)E to serve as the base point. For maximum security, the order of the base point should be near the field size. ‘B’ is a fixed publicly known point on E whose order is very large (either ‘n’ or a large prime divisor of ‘n’).

Lets ‘A’ represent Alice and ‘B’ represent Bob. They choose their private key to be a random bit pattern of the size of ‘q’. Let ‘a’ be Alice’s private key and ‘b’ be Bob’s private key. Alice computes

A_{pub} = aB(element of) F_q

Bob computes his public key as

B_{pub} = bB(element of) F_q

Alice takes B_{pub} and computes A_{shared} = a*B_{pub} =a(bB) (element of) F_q

The arithmetic’s is done modulo a prime polynomial. Similarly, Bob takes A_{pub} and computes B_{shared} = b* A_{pub} =b(aB) (element of) F_q

We illustrate the scenario using a table.

B = Base Point

a= Random number generated by Alice, private key of Alice.

b= Random number generated by Bob, private key of Bob.

ALICE	**BOB**
aB →	aB
bB ←	bB
a(bB) = abB	b(aB) = baB

Implementing cryptosystems over elliptic curves has the advantage that it provides greater strength against cryptanalysis when compared to conventional public key cryptosystems of the same key size. Because of **Joux** and **R. Lercier** and their contribution in Improvements to the general Number Field Sieve for discrete logarithms in prime fields, finding discrete logarithms over F_p, *p* a 110-digit prime has been shown possible in the year 2001 taking three weeks on an unique 525MHz quadri-processors Digital Alpha Server 8400 computer. In the year 2002, it was shown that calculating discrete logarithms over $F(2^{607})$ was feasible, using **Coppersmith's** index calculation method. However, till date there are no sub-exponential algorithms to solve the Elliptic Curve Discrete Log problem (ECDLP). One of the best attacks till date against the ECDLP is Teske's method and the algorithm is fully exponential in time.

BIBLIOGRAPHY

1. ANSI X9.62-199X, *Public Key Cryptography* for the Financial Services Industry.

2. Joseph.H.Silverman and John Tate*, Rational Points on Elliptic Curves, Undergraduates Texts in Mathematics*, Springer

3. Neal Koblitz, *A Course In Number Theory and Cryptography*, Springer, Second edition, 1994.

4. William Stallings, *Cryptography and Network Security*, Pearson Education, Third Edition.

5. John.B.Fraleigh, *A First Course in Abstract Algebra*, Narosa Publishing House, Third Edition.

6. Rudolf Lidl, Harald Niederreiter, ***Finite Fields-Encyclopedia of*** *Mathematics and its applications*, Cambridge University Press.

7. Alfred.J.Menezes,Ian.F.Blake,XuHongGao,Ronald.C.Mullin,Scott.A.Vanstone,Tomik Yaghoobian, *Application of Finite Fields*, Kluwer Academic Publishers

8. Alfred J. Menezes, Paul C. van Oorschot and Scott A.Vanstone, *Handbook of Applied Cryptography*, CRC press.

9. Kolman, Busby, Ross, *Discrete Mathematical Structures*, Prentice Hall India, Third Edition, 1996.

10. Tom Apostol**,** *Introduction to Analytical Number Theory*, Springer International, Student edition, 1989.

11. Bruce Schneier, *Applied Cryptography*, Wiley Publications, Second edition, 2001.

12. Ivan Niven, Herbert S.Zuckerman, *An Introduction to the Theory of Numbers,* Wiley Eastern Limited.

-Compiled by Sarad A.V