

Elliptic Curve Cryptography

Ashkan Hosseinzadeh Namin

**Research Centre for Integrated Microsystems
Electrical and Computer Engineering**



U N I V E R S I T Y O F
W I N D S O R

Outline

- **Available Public key Cryptographic Techniques**
 - Motivation
 - Introduction
 - Symmetric Key and Asymmetric key Cryptography
 - RSA Cryptosystem
 - El Gamal Cryptosystem
- **Elliptic Curve Cryptography**
 - Elliptic Curve Definition
 - Point Addition and scalar multiplication
 - Elliptic Curve cryptosystem
 - Elliptic Curve Discrete Logarithm Problem
 - Comparison between Public key Cryptosystems
- **Conclusions**

Motivation

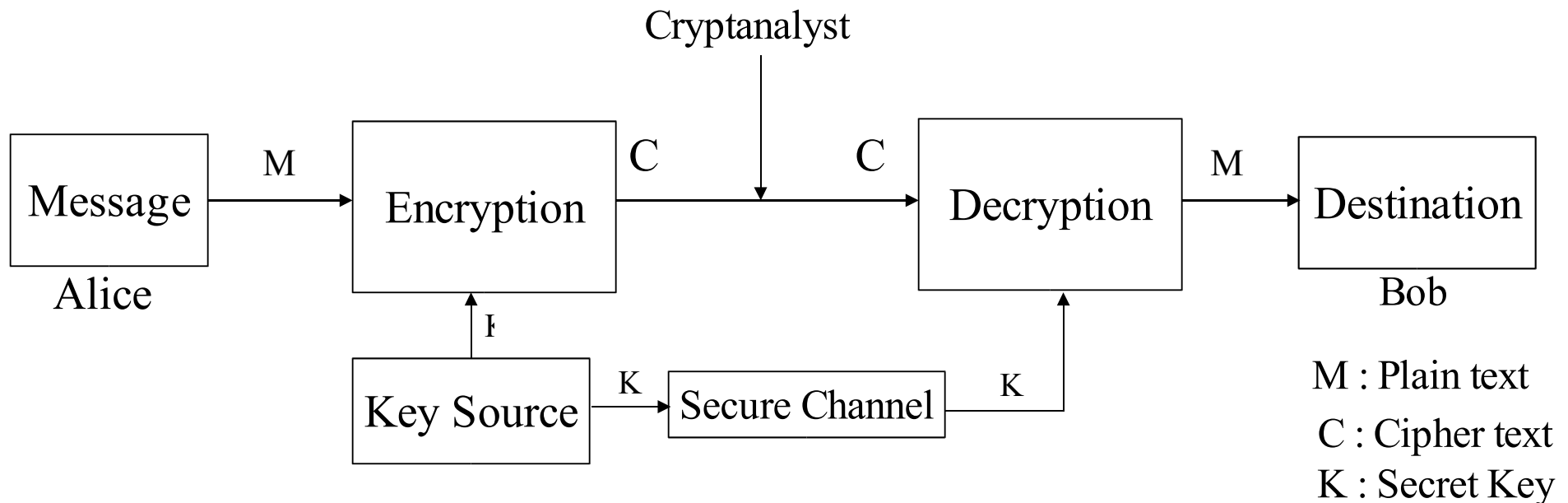
- **1976 Diffie and Hellman, first idea**
- **Efficient way to achieve secure data exchange between two unfamiliar parties**
- **RSA, El Gamal, ECC Cryptosystems**
- **With the same security level, smaller key size for ECC**
- **ECC implementations require less power, less memory**
- **Attractive for constrained devices like wireless devices and smart cards and handheld computers.**

Introduction

- **Cryptography : Greek word means Secret writing**
Scrambling of data so that only someone with the necessary **key** can unscramble it. Used for secure data transmission and storage.
- **Cryptanalysis** : Deals with the breaking of an encrypted data (scrambled data) to recover information
- **Two main categories of cryptography**
 - Symmetric Cryptography or Secret Key
 - Asymmetric Cryptography or Public Key

Symmetric Key Cryptography

- Alice and Bob agree on encryption method and a key
- Alice encrypts the message with the key and sends it to Bob
- Bob uses **the same key** to decrypt the message



Symmetric Key Cryptography

- **Advantages**

- High speed and high throughput
- Short key size (> 128 bits)
- Extensive history

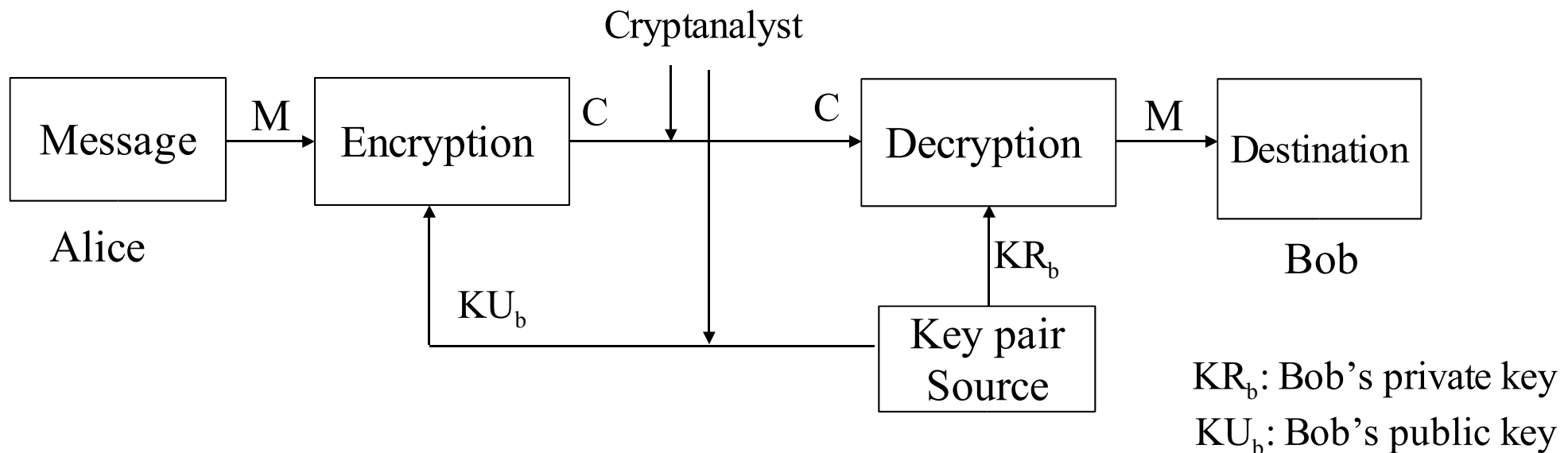
- **Disadvantages**

- The key must be remain secret at both ends
- In a large network there are many key pairs to be managed
(For n nodes $\frac{n \times (n - 1)}{2}$ keys are required).

2

Asymmetric Key Cryptography

- **Bob generates a key which makes public.**
- **Bob uses his public key to determine a second key which is his private key and keeps it secret.**
- **Alice uses Bob's **public key** to encrypt a message for him.**
- **Bob uses his **private key** to decrypt this message**



Asymmetric Key Cryptography

- **Advantages**

- Only the private key must be kept secret
- Easier key administration on a network
- In a large network, number of keys is smaller than symmetric-key scenario (n pairs of keys).

- **Disadvantages**

- Small throughput
- Larger Key size than symmetric-key encryption (160-1024 bits)
- Public key schemes have their security based on some hard mathematical problems
- Does not have as extensive a history (Started in 1970's)

Asymmetric Cryptography Techniques

- **The concept was introduced in 1976 by Diffie and Hellman as an algorithm for key exchange (they didn't come up with a practical cryptographic system)**
- **Today three different cryptographic systems are considered both secure and efficient.**
 - RSA (Based on integer factorization system)
 - El Gamal (based on discrete logarithm system)
 - Elliptic Curve (based on elliptic curve discrete logarithm problem)
- **All these cryptographic systems rely on the difficulty of a mathematical hard problem for their security and modular arithmetic plays a central role in their implementations.**

Asymmetric Cryptography Timeline

- **In 1978, L.M Adleman, R.L. Rivest and A. Shamir propose the RSA encryption method as the first public key algorithm. This algorithm is currently the most widely used.**
- **In 1985, Taher El Gamal proposed the discrete logarithm problem. In 1991 Schnorr discovered a variant Gamal's work which offers more efficiency. U.S government Digital Signature Algorithm is based on this technique.**
- **In 1985, Neil Koblitz and Victor Miller independently proposed the Elliptic Curve Cryptosystem (ECC). ECC is the strongest public key cryptographic system known today.**

RSA

- **Bob chooses two primes p and q and calculates $n=p \times q$**
- **Bob chooses e with $\gcd(e, (p-1) \times (q-1))=1$**
- **Bob calculates d with $d \times e=1 \pmod{(p-1) \times (q-1)}$**
- **Bob makes n and e public, and keeps p, q, d secret**

- **Alice encrypts m as $c=m^e \pmod{n}$**
- **Bob decrypts by calculating $m=c^d \pmod{n}$**

- **$m = c^d = m^{(d \times e)} = m^{(1)} = m \pmod{n}$**

RSA

- **RSA security relies on the difficulty of the Integer Factorization problem**
- **Integer Factorization problem :**
given a large prime number $n=p \times q$ factor n into it's prime numbers
- RSA efficiency rests on the speed of performing exponentiation modulo n .
- Up to 2003 the largest RSA modulus factored is a 530 bit binary number.

El Gamal

- Bob chooses prime p and a primitive root α and makes them public
- Bob also chooses a secret integer A and calculates $B = (\alpha)^A \pmod p$
- Bob public key is (p, α, B) and his private key is A
- Alice chooses a random integer k and calculates $K = (\alpha)^k$
- Alice encrypts m as $C_1 = \alpha^k, C_2 = B^k \times m \pmod p$
- Bob decrypts by calculating $C_2 \times (C_1)^{-A}$
- $m = C_2 \times (C_1)^{-A} = B^k \times m \times (\alpha^k)^{-A} = (\alpha^A)^k \times m \times (\alpha^k)^{-A} = m \pmod p$

El Gamal

- **El Gamal security relies on the difficulty of the Discrete Logarithm problem.**
- **Discrete Logarithm problem :**
Given pair g and y and prime number p such that $y = g^x \pmod{p}$
determine integer x
- El Gamal efficiency rests on the speed of performing modular exponentiation modulo p .
- Up to 2003 the largest DLP solved is a 397 bit binary number.



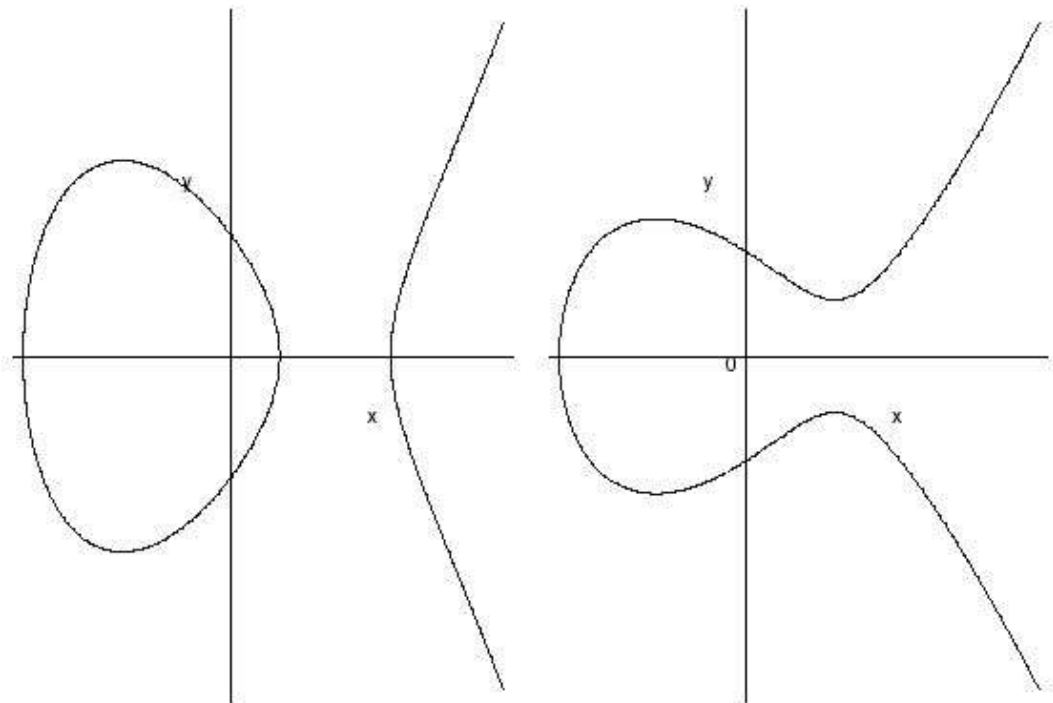
Elliptic Curve Definition

- An Elliptic Curve is the graph of equation of the form

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

(we assume that the curve has no multiple roots $4a^3+27b^2 \neq 0$)

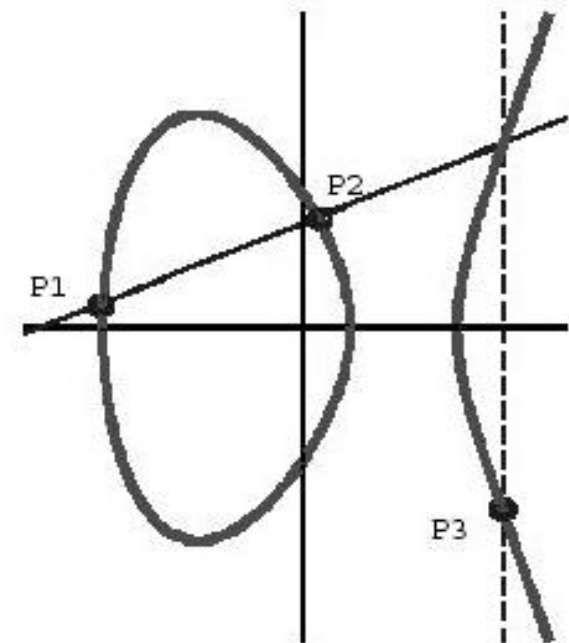
When we are working with real Numbers graph E has one of the two possible forms (it can have one real root or three real roots).



Elliptic Curve Point Addition

- if (x,y) satisfy the elliptic curve equation then $p=(x,y)$ is a point on the elliptic curve
- Suppose P_1 and P_2 are both points on the elliptic curve then $P_1 + P_2$ is always another point on the elliptic curve which is defined as

Draw a line through P_1 and P_2 (if $P_1 = P_2$ take the Tangent line). The line intersects the curve in a third Point. Reflect that point through the x-axis to find $P_3 = P_1 + P_2$



Elliptic Curve Point Addition

- For curve $y^2 = x^3 + ax + b$
- Point Addition $P(x_1, y_1) \neq Q(x_2, y_2)$

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

$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1} \right) \times (x_1 - x_3) - y_1$$

- Point Doubling $P(x_1, y_1)$

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1} \right)^2 - 2x_1$$

$$y_3 = \left(\frac{3x_1^2 + a}{2y_1} \right) \times (x_1 - x_3) - y_1$$

Elliptic Curve Scalar Multiplication

- **Scalar multiplication is the dominant computation part of ECC**
- **It computes $k \times P$ for a given point P and integer k .**
- **$Q = k \times P = (P + P + \dots + P)$ (($k-1$) addition)**
- **There are different methods for speeding up this process, The most common one is the **Binary Method** (also called **Double and Add Method**)**

```

For i = 0 to n-1
  If  $b_i=1$  then  $Q = Q + P$ 
   $P = P + P$ 
End

```

$$K = \sum_{i=0}^{n-1} b_i * 2^i$$

$$b_i = 0,1$$

Elliptic Curve & Finite Field

- Elliptic curve calculations are usually defined over finite field

The finite field is prime field $GF(P)$

The elements are $\{0,1,2,\dots,p-1\}$

all operations are modulo p

The finite field is a binary polynomial field $GF(2^m)$

The elements are binary polynomials

all operations are modulo 2

$$\mathbf{x} = \mathbf{a}_{m-1}\mathbf{X}^{m-1} + \mathbf{a}_{m-2}\mathbf{X}^{m-2} + \dots + \mathbf{a}_1\mathbf{X} + \mathbf{a}_0 \quad \mathbf{a}_i = \{0,1\}$$

Defining the curve over Binary Field will speed up the calculations

Elliptic Curve Cryptosystem

- **Bob chooses the curve E and point P on the curve**
- **Bob chooses integer d and calculates $Q=d \times P$ and makes it public**
- **Alice maps the plaintext m to point M on curve**
- **Alice chooses a random integer k**
- **Alice encrypts M as $C_1=k \times P$, $C_2= M + k \times Q$**
- **Bob decrypts by calculating $M=C_2 - d \times k \times P$**
- **$M = C_2 - d \times k \times P = M + k \times Q - d \times k \times P = M + k \times Q - d \times Q = M$**

(Elliptic curves, points on them and mapping formats are standardized)

Elliptic Curve Discrete Logarithm Problem

- **Elliptic Curve security relies on the difficulty of the Elliptic Curve Discrete Logarithm problem (ECDLP)**
- **Elliptic Curve Discrete Logarithm problem:**
- ECDLP is the inversion to scalar multiplication and defined as
Given points Q and P , find the integer k such that $Q = K \times P$
- ECC efficiency rests on the speed of calculating $k \times P$ for some integer k and a point P on the curve.

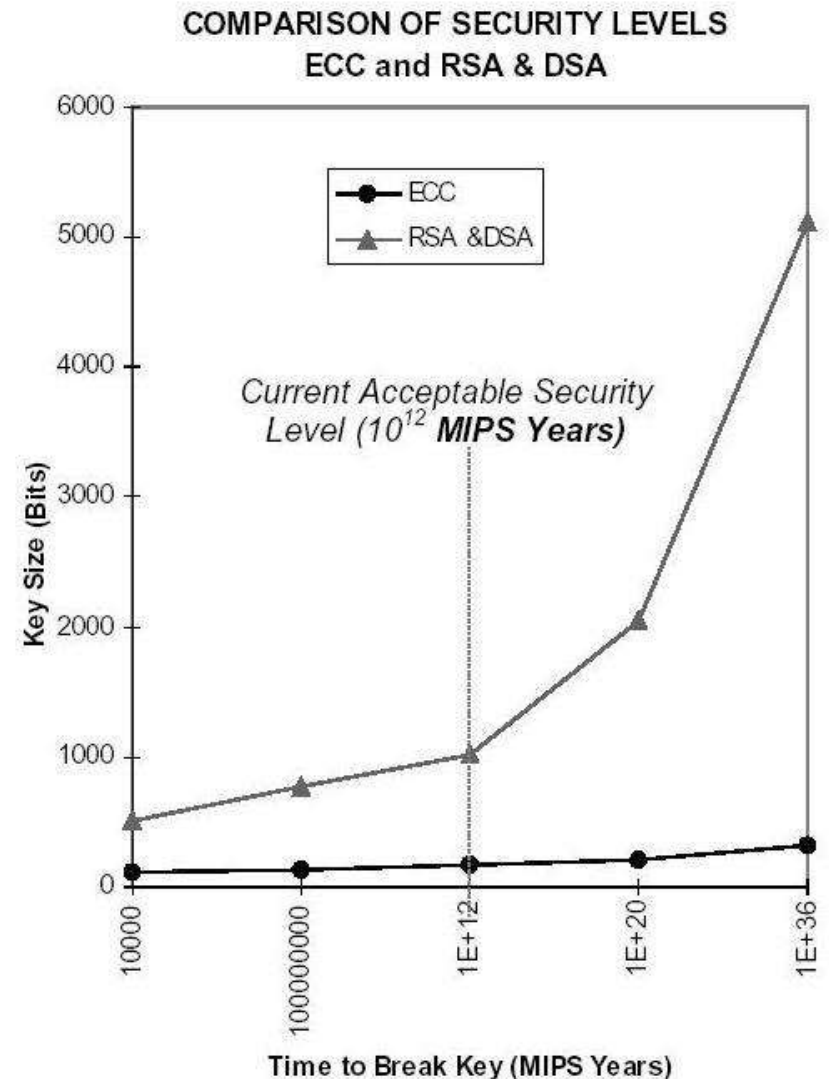
Up to 2003 the largest ECDLP solved is a 109 bit prime field binary number.



Comparison between Public Key Cryptosystems

Secure system : It is generally accepted that 10^{12} MIPS years represents reasonable security at this time.

MIPS year : computing time of one year on a machine capable of performing one million instructions per second.



Comparison between Public Key Cryptosystems

- To achieve reasonable security **today**, RSA and DSA (El Gamal) should employ a 1024 modulus, while a 160 bit modulus should be sufficient for ECC.
- The security gap between the systems grows as the key size grows. For example a 300 bit ECC provides the same security as a 2000 bit RSA or DSA
- Shorter keys reduce storage space for keys and faster computation speed which makes ECC suitable for constrained applications where computational power and bandwidth is limited.

Conclusions

- Information security through public key cryptography is required for electronic transactions for unfamiliar parties
- Three different approaches are RSA, El Gamal and ECC
- ECC offers the highest security (strength per bit)
- Security gap between systems grows as the key size grows
- ECC is suitable for constrained applications such as smart cards, tokens, wireless communication devices