RSA, ECC, & Pairing-based Cryptography

A basic introduction from a math perspective

Ning Shang

For filling the void in the cc

December, 2021

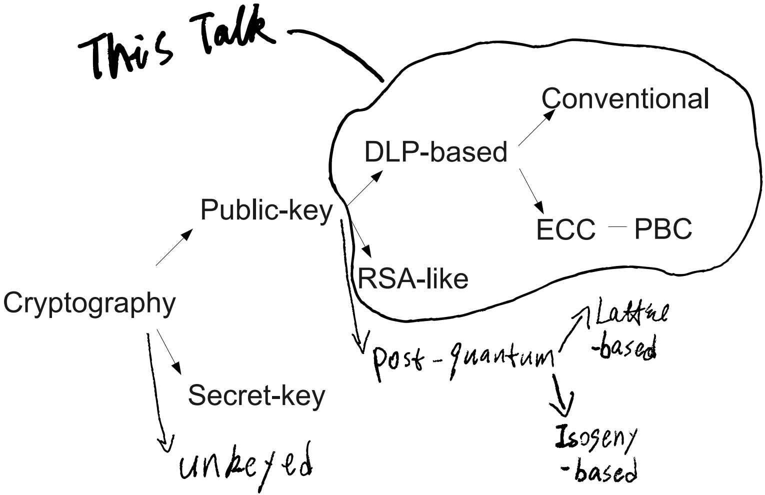
Chapter One: Overview of Cryptography

Cryptography: Overview

What is cryptography?

Study of mathematics techniques related to aspects of information security, e.g., how to hide information, data integrity, identification and authentication.

Cryptography: Overview (cont'd)



Chapter Two:
RSA (Factoring-based Crypto)

Rivest-Shamir-Adleman

Setup/KeyGen:

- Choose two large primes p and q.
- Let $n = p \cdot q$.
- Randomly choose e such that $1 < e < n - 1$ and $\gcd(e, \phi(n)) = 1$.
  Here $\phi(n) = (p-1)(q-1)$ is the Euler's totient function.
- Compute d such that $e \cdot d \equiv 1 \pmod{\phi(n)}$.
- The values p and q are never revealed.

RSA public key: (n, e).
RSA private key: d.

Remark: for RSA to work, factorization of the modulus n should be hard.

RSA Encryption (Basic Scheme)

Plaintext message: $M \in \{1, 2, \ldots, n-1\}$.
Encryption: $C := E(M) = M^e \pmod{n}$.
Decryption: $M := D(C) = C^d \pmod{n}$.

RSA Signatures (Basic Scheme)

Message to sign: $M \in \{1, 2, \ldots, n-1\}$.
Sign: signature $\sigma = D(M) = M^d \pmod{n}$.
Verify: check $M = E(\sigma) = \sigma^e \pmod{n}$.

Remark: The scheme shown above is not secure. In practice, one should sign the hash of the message instead of the message itself.

Chapter Three:
Elliptic Curve Crypto

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM) [Lenstra, 1987]

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM)   [Lenstra, 1987]
- Primality test (ECPP)   [Adleman and Huang, 1987]

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM) [Lenstra, 1987]
- Primality test (ECPP) [Adleman and Huang, 1987]
- Key management schemes [Bertino et al., 2008]

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM) [Lenstra, 1987]
- Primality test (ECPP) [Adleman and Huang, 1987]
- Key management schemes [Bertino et al., 2008]
- Hash function construction [Charles et al., 2007]

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM) [Lenstra, 1987]
- Primality test (ECPP) [Adleman and Huang, 1987]
- Key management schemes [Bertino et al., 2008]
- Hash function construction [Charles et al., 2007]
- Zero-knowledge proofs

Elliptic Curves in Cryptography

Application of elliptic curves in cryptography

- Public-key encryption and signature algorithms (ECC)
- Integer factorization method (ECM) [Lenstra, 1987]
- Primality test (ECPP) [Adleman and Huang, 1987]
- Key management schemes [Bertino et al., 2008]
- Hash function construction [Charles et al., 2007]
- Zero-knowledge proofs
- And so on

A Mathematician Quote

"It is possible to write endlessly on elliptic curves. (This is not a threat.)" — Serge Lang

We Follow The Wise

We are not going to talk about everything ...

Focus of this talk: ECC

Elliptic curve cryptography: use elliptic curves as an approach to public-key cryptography

Elliptic Curve

$$E : y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_q,\ q \text{ odd } 3 \nmid q$$

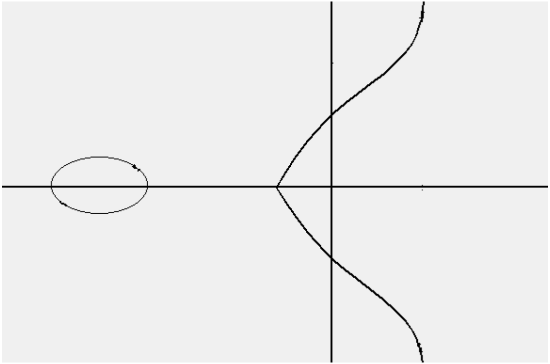


Figure: An example of elliptic curve over $\mathbb{R}$

Review: Group

A group is a set G together with a binary operation $\circ$ that satisfies

1. closure: $a, b \in G \Rightarrow a \circ b \in G$
2. associativity: for any $a, b, c \in G$, have $(a \circ b) \circ c = a \circ (b \circ c)$
3. identity: there exists an identity element i such that $a \circ i = i \circ a = a$, for any $a \in G$
4. inverse: for any element $a \in G$, there is an element $b \in G$ such that $a \circ b = b \circ a = i$.

Review: Group (cont'd)

Abelian Group

A group G is called **abelian** if $a \circ b = b \circ a$, for any $a, b \in G$.

Examples of abelian groups

- $(\mathbb{Z}, +)$
- $(\mathbb{F}_p^\times, \cdot)$
- $(E(\mathbb{F}_q), +)$: elliptic curve over finite field

Finite abelian groups are convenient for cryptography, because the group law behaves well!

No need to worry about things like $a + b \neq b + a$

Discrete Logarithm Problem

Discrete logarithm problem (DLP)

$G = \langle \alpha \rangle$: finite cyclic group of order N with generator α; written multiplicatively. $\beta = \alpha^n$ for some $n \in \{0, 1, 2, \ldots, N-1\}$.
DLP: given α, β, find n.

Discrete Logarithm Problem (cont'd)

Example

$G = (\mathbb{Z}/(1152921504606847363))^{\times} = \langle\alpha\rangle, \alpha = \overline{12345678}$.
Given $n = 640511947003800044$, it is easy to compute
$\beta = \alpha^n = \overline{243069074995667794}$: $O(\log N)$.
If only $\alpha = \overline{12345678}$ and $\beta = \overline{243069074995667794}$ are given, how to recover n?

Discrete Logarithm Problem (cont'd)

Bad news: this is in general a hard problem.
Best generic methods take time $O(\sqrt{N})$ – Shank's BSGS, Pollard's ρ and λ methods.

Discrete Logarithm Problem (cont'd)

Good news: we can use it to do cryptography!

Discrete Logarithm Based Cryptography

Example: Diffie-Hellman(-Merkle) key exchange protocol

Enables two parties to share a common secret (e.g. an symmetric encryption key) over an insecure communications channel.
How it works:

1. Alice and Bob agree on a finite cyclic group G and a generator α.
2. Alice chooses a secret integer m, computes $\beta = \alpha^m$, and sends β to Bob.
3. Bob chooses a secret integer n, computes $\gamma = \alpha^n$, and sends γ to Alice.
4. Alice computes $\gamma^m = \alpha^{nm}$; Bob computes $\beta^n = \alpha^{mn}$; this information is used as their shared secret.

Groups Suitable for DLP Based Cryptography

Requirements

- Intractability of DLP:
  - group order is desired to be "almost prime": to resist the Pohlig-Hellman attack.
  - No "easy" transformation into another group where DLP can be solvable in less time: to resist attacks like MOV, GHS, ...
- Compact representation of group elements.
- Efficient group operations.

Group Suitable for DLP Based Cryptography (cont'd)

Candidates:

- Multiplicative groups of finite fields $\mathbb{F}_q^\times$.
- Class groups of orders in number fields.
- Abelian varieties over finite fields, e.g., Jacobians of algebraic curves, **ELLIPTIC CURVES**.
- Others.

Elliptic Curve Group

Elliptic curve $E : y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_q$, q odd, $3 \nmid q$

$\mathbb{F}_q$-rational points of E,

$$E(\mathbb{F}_q) := \{(x, y) \in (\mathbb{F}_q)^2 : y^2 = x^3 + ax + b\} \cup \{\infty\}$$

Elliptic Curve Group

Elliptic curve $E : y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_q$, q odd, $3 \nmid q$

$\mathbb{F}_q$-rational points of E,

$$E(\mathbb{F}_q) := \{(x, y) \in (\mathbb{F}_q)^2 : y^2 = x^3 + ax + b\} \cup \{\infty\}$$

Important fact about $E(\mathbb{F}_q)$

$E(\mathbb{F}_q)$ is a finite abelian group
NON-TRIVIAL! Do Not Try This at Home!

Group operation in $E(\mathbb{F}_q)$ is often written as addition $(+)$
Scalar multiplication: $[m]P := \underbrace{P + P + \ldots + P}_{m \text{ times}}$

Elliptic Curve Diffie-Hellman(-Merkle) key exchange

Rewrite Diffie-Hellman key exchange in language of elliptic curves.

1. Alice and Bob agree on an elliptic curve $E(\mathbb{F}_q)$ and a point $P \in E(\mathbb{E}_q)$ as base point.
2. Alice chooses a secret integer m, computes $Q = [m]P$, and sends Q to Bob.
3. Bob chooses a secret integer n, computes $R = [n]P$, and sends R to Alice.
4. Alice computes $[m]R = [nm]P$; Bob computes $[n]Q = [mn]P$; this information is used as their shared secret.

Group Operation in $E(\mathbb{F}_q)$

$$E : y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_q,\ q \text{ odd},\ 3 \nmid q$$

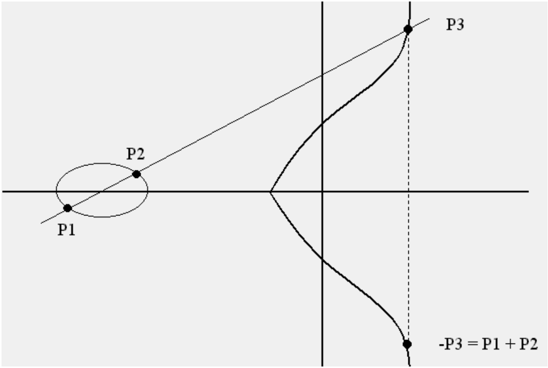


Figure: Illustration of points addition (courtesy of garykessler.net)

Group Law for $E(\mathbb{F}_q)$

Input: $P_1(x_1, y_1), \quad P_2(x_2, y_2)$
Output $Q(x, y) = P_1 + P_2$

Case $P_1 \neq P_2$: I, 3M/S

$$\lambda = (y_2 - y_1)/(x_2 - x_1)$$
$$x = \lambda^2 - x_1 - x_2$$
$$y = (x_1 - x)\lambda - x - y_1$$

Case $P_1 = P_2$: I, 4M/S

$$\lambda = (3x_1^2 + a)/(2y_1)$$
$$x = \lambda^2 - 2x_1$$
$$y = (x_1 - x)\lambda - x - y_1$$

Elliptic curve $E(\mathbb{F}_q)$ vs. $\mathbb{F}_p^\times$

Reason to replace "conventional" choice of the multiplicative group $\mathbb{F}_p^\times$?

Elliptic curve $E(\mathbb{F}_q)$ vs. $\mathbb{F}_p^\times$

Reason to replace "conventional" choice of the multiplicative group $\mathbb{F}_p^\times$?

EFFICIENCY

At same level of security, crypto schemes implemented with elliptic curves run faster than their $\mathbb{F}_q^\times$ counterparts, with much shorter key lengths.

Elliptic curve $E(\mathbb{F}_q)$ vs. $\mathbb{F}_p^\times$ (cont'd)

n: bit length of q
N: bit length of p

Complexity of best known attacks to DLP

- Elliptic curve $E(\mathbb{F}_q)$:

$$C_{\mathrm{ec}} = 2^{n/2}$$

- Multiplicative group $\mathbb{F}_q^\times$:

$$C_{\mathrm{conv}} = \exp(1.92 N^{1/3} (\log(N \log 2))^{2/3})$$

Elliptic curve $E(\mathbb{F}_q)$ vs. $\mathbb{F}_p^\times$ (cont'd)

Crude Estimate

$$C_{\mathrm{ec}} = C_{\mathrm{conv}} \Longrightarrow n = 4.91 N^{1/3} (\log(N \log 2))^{2/3}$$

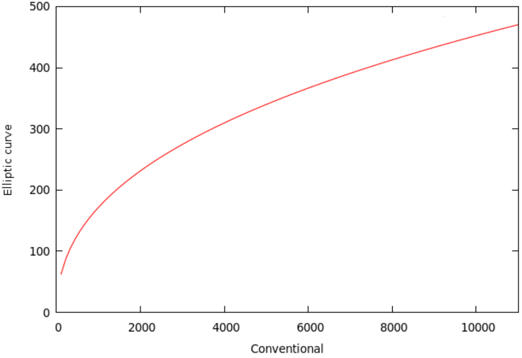


Figure: Elliptic curve vs. $\mathbb{F}_p^\times$ key sizes (in bits) for similar security level

Elliptic Curve $E(\mathbb{F}_q)$ vs. $\mathbb{F}_p^\times$ (cont'd)

Estimate above tells us

bit-strength	size q (bits)	size p (bits)
87	173	1024
117	233	2048
157	313	4096
209	417	8192

The estimate is at best crude, but it at least gives us some ideas about the low-costness of ECC over conventional public-key cryptosystems.

Elliptic Curve vs. RSA

Similar arguments apply to cryptosystems based on RSA.

Elliptic Curve vs. RSA

Similar arguments apply to cryptosystems based on RSA.

In practice, the performance comparison relies on implementation.

Commercial Time

In the commercial cryptographic literature,
1024-bit RSA $\approx$ 160-bit ECC [Lenstra and Verheul, 2001].

DSA vs. ECDSA

q: order of prime field $\mathbb{F}_q$
r: order of subgroup of $\mathbb{F}_r^\times$
n: order of subgroup of $E(\mathbb{F}_p)$
h: cofactor such that $|E(\mathbb{F}_p)| = n \cdot h$

The Digital Signature Standard (FIPS PUB 186-3) recommends

DSA

size q	1024	2048	2048	3072
size r	160	224	256	256

ECDSA

size n	161-223	224-255	256-383	384-511	≥ 512
max h	2^{10}	2^{14}	2^{16}	2^{24}	2^{32}

In Real World

Elliptic curve cryptography

- Elliptic curve based public-key cryptography is part of NSA Suite B.
- ECC has been written into various industrial standards eg. Digital Signature Standard (FIPS PUB 186-3).
- ECC has been integrated in many software and hardware (eg. latest version of OpenSSL, NSS v3.8, Firefox).
- A number of companies dedicated to ECC (eg. Certicom)
- Numerous white papers, presentations and publications (of course)

Chapter Four: Pairing-Based Crypto

Beyond Efficiency: Bilinear Maps

Many useful cryptographic protocols and applications, eg. IBE, short signatures, aggregate signatures, require use of a bilinear map

$$e : G_1 \times G_2 \to G_T$$

Bilineality

For all $P_1 \in G_1$, $P_2 \in G_2$, $m, n \in \mathbb{Z}$, $e([m]P_1, [n]P_2) = e(P_1, P_2)^{mn}$

Non-degeneracy

For $\mathcal{O} \neq P_1 \in G_1$, there exists $P_2 \in G_2$ such that $e(P_1, P_2) \neq 1$

Beyond Efficiency: Bilinear Maps (cont'd)

For certain kinds of elliptic curves, there exist efficient implementation of bilinear maps – the elliptic curve pairings (eg. Weil pairings, Tate pairings, Ate pairings, ...)

$$e : G_1 \times G_2 \to G_T$$

G_1 and G_2 are subgroups of $E(\mathbb{F}_q)$, G_T is a subgroup of $\mathbb{F}_{q^k}^{\times}$ for some small integer k. $|G_1| = |G_2| = |G_T| = r$.

Elliptic curve pairings are so far the only known efficient implementation of bilinear maps suitable for cryptography.

Miller's algorithm for Tate pairings

Algorithm 1 Basic Miller's algorithm

Input: $P \in G_1 \subseteq E(\mathbb{F}_{q^k})$, $Q \in G_2 \subseteq E(\mathbb{F}_{q^k})$, where r is the order of P

Output: $e(P, Q)$

$T \leftarrow P, f \leftarrow 1$
for $i = \lfloor \lg(r) \rfloor - 1$ to 0 **do**
    $f = f^2 \cdot l_{T,T}(Q)/v_{2T}(Q)$
    $T = 2T$
    if the i-th bit (from right) of r is 1 **then**
        $f = f \cdot l_{T,P}(Q)/v_{T+P}(Q)$
        $T = T + P$
    end if
end for
$f \leftarrow f^{(p^k-1)/r}$
return f

Here $l_{A,B}(Q)$ and $v_{A+B}(Q)$ are the "line" and "vertical" functions, resp.

Elliptic Curves Suitable for PBC

"Pairing-friendly curves"

- Present additional mathematical structures.
- Provide additional crypto hardness assumptions.
- Must satisfy all requirements for (regular) ECC.
- Have low density in all elliptic curves suitable for ECC

Applications of PBC

You cannot do (or do better) the following things without PBC.

- Tripartite key exchange [Joux, 2000]
- Identity-based encryption (IBE) [Boneh and Franklin, 2003], attributed-based encryption (ABE)
- Short signature scheme [Boneh et al., 2001]
- Aggregate signature scheme with different signing keys [Boneh et al., 2003]
- Efficient non-interactive zero-knowledge proofs [Groth and Sahai, 2008]
- Broadcast encryption scheme [Boneh and Waters, 2006]


The
END

References I

Adleman, L. and Huang, M. (1987).
Recognizing primes in random polynomial time.
In *STOC '87: Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 462–469, New York, NY, USA. ACM.

Bertino, E., Shang, N., and Wagstaff, Jr., S. (2008).
An efficient time-bound hierarchical key management scheme for secure broadcasting.
IEEE Trans. Dependable Secur. Comput., 5(2):65–70.

Boneh, D. and Franklin, M. (2003).
Identity-based encryption from the weil pairing.
SIAM J. Comput., 32(3):586–615.

Boneh, D., Gentry, C., Lynn, B., and Shacham, H. (2003).
Aggregate and Verifiably Encrypted Signatures from Bilinear Maps.
In *EUROCRYPT*, pages 416–432.

Boneh, D., Lynn, B., and Shacham, H. (2001).
Short Signatures from the Weil Pairing.
In *ASIACRYPT '01: Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security*, pages 514–532, London, UK. Springer-Verlag.

Boneh, D. and Waters, B. (2006).
A fully collusion resistant broadcast, trace, and revoke system.
In *ACM Conference on Computer and Communications Security*, pages 211–220.

Charles, D., Goren, E., and Lauter, K. (2007).
Cryptographic hash functions from expander graphs.
Journal of Cryptology.

References II

Groth, J. and Sahai, A. (2008).
Efficient Non-interactive Proof Systems for Bilinear Groups.
In *EUROCRYPT*, pages 415–432.

Joux, A. (2000).
A One Round Protocol for Tripartite Diffie-Hellman.
In *ANTS-IV: Proceedings of the 4th International Symposium on Algorithmic Number Theory*, pages 385–394, London, UK. Springer-Verlag.

Lenstra, A. K. and Verheul, E. R. (2001).
Selecting cryptographic key sizes.
Journal of Cryptology, 14:255–293.

Lenstra, Jr., H. W. (1987).
Factoring integers with elliptic curves.
Ann. of Math. (2), 126(3):649–673.