ĐẶNG MINH TUẤN

HỆ MẬT MÃ KHÓA CÔNG KHAI DỰA TRÊN ĐƯỜNG CONG ELLIPTIC

HÀ NỘI, 4-2016

Mở đầu

Tháng 3 năm 2016, Bộ Ngoại Giao Hoa Kỳ, đứng đầu là bộ trưởng John Kerry, đã dẫn một đoàn đại biểu tới các nước ASEAN trong đó có Việt Nam để thảo luận về phát triển Fintech và đặc biệt là về công nghệ Blockchain[1].Tháng 9 năm 2015, Ủy ban giao dịch hàng hoá tương lai Mỹ công bố, Bitcoin đã chính thức được đưa vào danh sách hàng hóa được phép giao dịch tại Mỹ[2].Công nghệ Blockchain và Bitcoin là công nghệ tiền số ra đời năm 2009 và ngày càng có nhiều quốc gia và các tổ chức, doanh nghiệp cho phép lưu hành và thanh toán bằng loại tiền số này trong không gian mạng Internet toàn cầu. Tháng 4-2016, giá trị thương mại của Bitcoin đã lên đến 6.5 tỷ USD [3]. Nền tảng cơ sở của Bitcoin chính là lý thuyết về mật mã mà cụ thể ở đây là hàm băm và lý thuyết về chữ ký số dựa trên Hệ mật đường cong Elliptic (ECC).

Bên cạnh việc sử dụng trong tiền số Bitcoin[4], ECC còn được ứng dụng rất nhiều trong thực tiễn ngành Công nghệ thông tin [1]. Các trang Web bảo mật https (http-secure) thường được dùng trong thanh toán điện tử hay ứng dụng riêng tư như gmail đều sử dụng các giao thức TLS (Transport Layer Security) mà trước đó là

[1]B. Cohen. (April 12, 2016) U.S. State Department Recommends Development of Blockchain and Distributed Ledgers to International Partners. [Online]. Available: http://www.nasdaq.com/article/us-state-department-recommendsdevelopment-of-blockchain-and-distributed-ledgers-to-international-partnerscm605334.

[2](21/09/2015) Bitcoin chính thức được Mỹ công nhận là hàng hoá. [Online]. Available: http://vnreview.vn/tin-tuc-kinh-doanh/-/view-content/content/1654484/bitcoin-chinh-thuc-duoc-My-cong-nhan-la-hang-hoa.

[3](15/4/2016) https://markets.blockchain.info/

[4]Địa chỉ ví Bitcoin được tính dựa trên khóa công khai của ECC với hàm băm bảo mật có độ dài 256-bit SHA256 và thuật toán Base58Encode dùng để chuyển số thành dạng 56 ký tự, RIPEMD (RACE Integrity Primitives Evaluation Message Digest) là một họ hàm băm bảo mật khác:

$$\begin{aligned}
\texttt{Version} &= 1(byte)\\
\texttt{KeyHash} &= \texttt{Version} + \texttt{RIPEMD}(\texttt{SHA256}(\texttt{PublicKeyEC}))\\
\texttt{CheckSum} &= \texttt{SHA256}(\texttt{SHA256}(\texttt{KeyHash}))\\
\texttt{BitcoinAddress} &= \texttt{Base58Encode}(\texttt{KeyHash} + \texttt{CheckSum})
\end{aligned}$$

SSL (Secure Socket Layer). Trong các giao thức này ECC được sử dụng để trao đổi khóa phiên. Các giao dịch remote access được sử dụng rất nhiều trong thế giới Unix, Linux là SSH (Secure SHell) cũng sử dụng ECC để trao đổi khóa. Ưu điểm của hệ mật sử dụng đường cong Elliptic (ECC) là có độ dài khóa nhỏ (160 bit tương đương với khóa độ dài 1024 Bit trong hệ mật RSA), do sử dụng độ dài khóa nhỏ nên tài nguyên phục vụ cho ECC thường nhỏ hơn rất nhiều, bên cạnh đó hiệu năng tính toán cũng được nâng cao rõ rệt. Hiện nay ECC đang là xu thế để thay thế RSA.

Cơ sở toán học của hệ mật ECC là nhóm giao hoán Abel các điểm nằm trên đường cong Elliptic. Ngoài việc đường cong Elliptic là cơ sở cho hệ mật ECC, hệ mật ID-Based, đường cong Elliptic (EC) còn là công cụ hữu hiệu để phân tích số nguyên ra thừa số nguyên tố [2, 3, 4], hoặc dùng để kiểm tra tính nguyên tố của số nguyên [3]. EC cũng là cơ sở để chứng minh định lý Fermat nổi tiếng đã tồn tại nhiều trăm năm qua.

Đường cong Elliptic là một trường hợp đặc biệt của phương trình Diophant. Lý thuyết về đường cong Elliptic (EC) rất phong phú và đồ sộ. Trong [5] tác giả Serge Lang đã phát biểu về phương diện học thuật: "*Có thể viết vô tận về đường cong Elliptic*". Các lý thuyết và khái niệm liên quan tới EC có thể liệt kê một số như dưới đây:

- Lý thuyết nhóm, vành, trường trong đại số trừu tượng [3, 6];
- Đa tạp Affine, đa tạp Jacobian và đa tạp xạ ảnh trong hình học đại số [7, 8];
- Điểm Torsion, Divisor, cặp song tuyến tính Weil, Tate-Lichtenbaum [3];
- Lý thuyết trường Galois, tự đồng cấu-ánh xạ Frobenius [3, 9];
- Lý thuyết Baker-Feldman, Baker-Tijdeman và lý thuyết Kummer [5];
- Số p-adic, Isogenies, hàm Sigma và hàm Zeta [5, 7, 10, 3, 4];
- Nhóm đối đồng điều, đối đồng điều Galois và đối đồng điều phi giao hoán (Topo đại số) [8, 4];
- Nhóm Mordell–Weil, Selmer và nhóm Shafarevich–Tate [8, 11];
- Phương pháp hình học và Tựa tuyến tính (Quasilinear) [12].

Với ý nghĩa to lớn cả về thực tiễn và học thuật, EC là nền tảng toán học quan trọng trong đại số hiện đại cũng như lý thuyết mật mã hiện đại. EC cũng là nền tảng quan trọng trong chính phủ điện tử và thương mại điện tử. Chính vì những điều này mà Chuyên đề "*Hệ mật mã khóa công khai dựa trên đường cong Elliptic*" được lựa chọn để trình bày báo cáo.

Với khối lượng kiến thức và khái niệm đồ sộ như đã liệt kê ở trên việc nghiên cứu và đào sâu về đường cong Elliptic gặp không ít khó khăn cho những người làm

Công nghệ thông tin (CNTT) mà toán học không phải là chuyên môn chính. Mục tiêu của chuyên đề này là tổng hợp những khái niệm và kiến thức cơ bản nhất của EC liên quan đến cơ sở toán học của Hệ mật dựa trên đường cong Elliptic. Đồng thời người viết cũng chứng minh lại một số định lý và bổ đề theo cách dễ hiểu hơn, tránh dùng đến các khái niệm quá phức tạp và xa lạ với chuyên ngành CNTT. Các phép toán của đường cong Elliptic được trình bày trong báo cáo phần lớn được cài đặt bằng phần cứng sử dụng công nghệ FPGA (Field-Programmable Gate Array) của Xilinx trong khuôn khổ Đề tài cấp Nhà nước KC01-18 (đã được nghiệm thu trong năm 2014) do người viết báo cáo làm chủ nghiệm đề tài, kết quả được công bố tại [13].

Phạm vi của chuyên đề cũng được giới hạn với những khái niệm và lý thuyết đủ cho các ứng dụng cơ bản của EC, các phát triển của EC thành hệ mật ID-Based, hoặc các ứng dụng về chữ ký số tập thể, chữ ký số nhóm, chữ ký ngưỡng, chứ ký ủy nhiệm, chứ ký số mù sẽ không được đề cập đến trong khuôn khổ của báo cáo này.

Báo cáo chuyên đề được kết cấu thành 02 chương, chương 1 trình bày các khái niệm, định nghĩa cơ bản về đường cong Elliptic (Phương trình của EC, nhóm cộng Abel các điểm trên đường cong, chứng minh định lý về nhóm...). Chương 2 trình bày về Hệ mật dựa trên đường cong Elliptic và một số ứng dụng trong mã hóa, xác thực chữ ký số, trao đổi khóa dự trên bài toán khó Logarithm rời rạc.

Hà Nội, ngày 15 tháng 04 năm 2016

Đặng Minh Tuấn[5]

[5]Đặng Minh Tuấn, Mobile: +84-98-868-6636, Email: tuanvietkey@gmail.com.

Mục lục

Kí hiệu

$\mathbb{N}$	tập hợp số tự nhiên
$\mathbb{N}^*$	tập hợp số tự nhiên khác 0
$\mathbb{Z}$	trường số nguyên
$\mathbb{Q}$	trường số hữu tỉ
$\text{char}(K)$	đặc số của trường K
$\equiv$	dấu đồng dư
∞	dương vô cùng (tương đương với $+\infty$)
gcd	ước số chung lớn nhất
deg	bậc của đa thức
det	định thức
RSA	Hệ mã công khai dựa trên bài toán phân tích ra thừa số nguyên tố do Rivest-Shamir-Adleman phát triển
EC	Đường cong Elliptic (Elliptic Curve)
ECC	Hệ mật dựa trên đường cong Elliptic (Elliptic Curve Cryptography)
ECDLP	Elliptic Curve Logarithm Problem
ECDH	Thuật toán Elliptic Curve Diffie–Hellman
ECDSA	The Elliptic Curve Digital Signature Algorithm
ECIES	The Elliptic Curve Integrated Encryption System
ECMQV	Elliptic Curve Menezes–Qu–Vanstone protocol

Chương 1

Kiến thức cơ sở về đường cong Elliptic

1.1 Tổng quan về đường cong Elliptic

Năm 250 sau Công nguyên, Diophant khi giải bài toán tìm số tầng của tháp các quả cầu mà khi trải ra mặt đất có thể xếp thành một hình vuông đã dẫn đến giải phương trình (y là số quả cầu trên 1 cạnh hình vuông; x là số tầng của tháp):

$$y^2 = 1^2 + 2^2 + 3^2 + \cdots + x^2 = \frac{x(x+1)(2x+1)}{6}$$

Phương trình $y^2 = x(x+1)(2x+1)/6$ là một dạng của đường cong Elliptic.

Năm 1637, nhà toán học và vật lý học người Pháp Pierre de Fermat công bố định lý Fermat cuối cùng khi viết trên lề bản copy công trình của Diophant: Phương trình sau đây là vô nghiệm:

$$x^n + y^n = z^n, \quad n > 2$$

Hơn ba thế kỷ, đã có rất nhiều nhà toán học cố gắng chứng minh định lý này xong đều thất bại, mãi cho đến năm 1994, Andrew Wiles, giáo sư trường Princeton đã gây một tiếng vang lớn trong cộng đồng toán học thế giới vào thời điểm đó khi sử dụng đường cong Elliptic có dạng $y^2 = x(x - a^n)(x + b^n)$ cùng với lý thuyết về Modul để chứng minh định lý Fermat cuối cùng.

Năm 1987, Trong [14], Lenstra đề xuất thuật toán phân tích số nguyên ra thừa số nguyên tố sử dụng đường cong Elliptic, đó là thuật toán tương đối nhanh, chạy với thời gian dưới hàm mũ và là thuật toán nhanh thứ 3 trong việc phân tích ra thừa số nguyên tố, sau phương pháp sàng đa thức toàn phương và phương pháp sàng trường số tổng quát.

Trong lĩnh vực mật mã, vào năm 1985, Victor S. Miller công bố bài báo đầu tiên về ứng dụng đường cong EC trong mật mã "*Use of Elliptic Curves in Cryptography*" [15] và sau đó là Neal Koblitz với "*Elliptic curve cryptosystem*" [16] vào năm 1987. Từ đó cho đến nay đã có rất nhiều công bố nghiên cứu về EC về lý thuyết và trong thực tiễn càng ngày ứng dụng ECC càng được sử dụng rộng rãi, và đã được đưa thành các tiêu chuẩn.

Một số tiêu chuẩn liên quan đến đường cong Elliptic:

IEEE 1363:	Tiêu chuẩn này bao gồm gần như tất cả các thuật toán về các hệ khóa công khai trong đó có ECDH, ECDSA, ECMQV và ECIES. Trong phần phụ lục có cả các thuật toán cơ bản về lý thuyết số liên quan đến hệ mật khóa công khai.
ANSI X9.62 và X9.63:	Các chuẩn này tập trung vào đường cong Elliptic và cụ thể về ECDSA trong X9.62 và ECDH, ECMQV và ECIES trong X9.63. Các chuẩn này cũng xác định khuôn dạng các dữ liệu và danh mục các đường cong khuyến cáo sử dụng.
FIPS 186.2:	Tiêu chuẩn của NIST cho chữ ký số, mô tả chi tiết về thuật toán DSA algorithm.
SECG:	Là tiêu chuẩn được biên soạn bởi nhóm các doanh nghiệp dẫn dắt bởi công ty Certicom, gần như là ánh xạ của các chuẩn ANSI nhưng được tiếp cận trên môi trường Web từ Website http://www.secg.org/
ISO 15946-2:	Tiêu chuẩn mô tả về ECDSA và ECIES (còn được gọi là ECIES-KEM).
RFC 3278:	"Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)" là khuyến nghị sử dụng thuật toán ECC trong mã hóa thông điệp văn bản.

1.2 Phương trình Weierstraß của đường cong Elliptic

Trong tài liệu này, đa phần số các đường cong Elliptic sẽ được nghiên cứu dưới dạng sau:

$$y^2 = x^3 + Ax + B, \tag{1.1}$$

Trong đó A và B là các hằng số. Các giá trị của x, y, A, B thường là các giá trị trên một trường nào đó, ví dụ như $\mathbb{R}$ (số thực), $\mathbb{Q}$ (số hữu tỷ), $\mathbb{C}$ (số phức), hoặc trường hữu hạn $\mathbb{F}_q$, với $q = p^n$ trong đó p là số nguyên tố với $n \geqq 1$. Nếu K là một trường có $a, b \in K$, khi đó ta nói đường cong Elliptic được định nghĩa trên trường K. Điểm (x, y) trên đường cong Elliptic với $(x, y) \in K$ được gọi là điểm K–Hữu tỷ. Dạng tổng quát phương trình Weierstrass của đường cong Elliptic sẽ được biểu diễn dưới dạng:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{1.2}$$

Trong đó $a_1, \cdots, a_6$ là các hằng số. Dạng (1.2) thường được sử dụng với các trường K có đặc số chap(K) bằng 2 hoặc 3. Khi K có chap(K) khác 2 có thể biến đổi (1.2) thành dạng sau:

$$\left(y + \frac{a_1x}{2} + \frac{a_3}{2}\right)^2 = x^3 + \left(a_2 + \frac{a_1^2}{4}\right)x^2 + \left(a_4 + \frac{a_1a_3}{2}\right)x + \left(\frac{a_3^2}{4} + a_6\right),$$

Có thể viết lại như sau:

$$y_1^2 = x^3 + a_2'x^2 + a_4'x + a_6',$$

Với $y_1 = y + a_1x/2 + a_3/2$ và với các hằng số a_2', a_4', a_6'. Khi K có chap(K) khác 3 có thể dùng phép thế $x_1 = x + a_2'/3$ và ta có:

$$y_1^2 = x_1^3 + Ax + B,$$

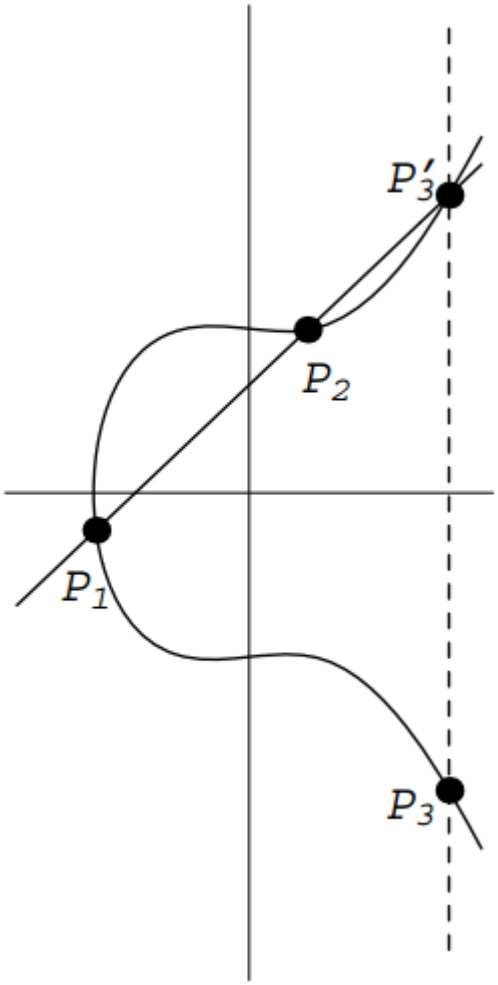
Trong đó A, B là các hằng số nào đó. Đường cong (1.1) có định thức $\Delta = -16(4A^3 + 27B)$. Đường cong này sẽ suy biến và không có đủ 3 nghiệm phân biệt khi $\Delta = 0$, trong tài liệu này chúng ta chỉ xét các đường cong có $\Delta \neq 0$.

1.3 Cộng các điểm trên đường cong Elliptic

Xét hai điểm $P_1 = (x_1, y_1)$ và $P_2 = (x_2, y_2)$ trên đường cong Elliptic $E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$. Phép cộng giữa hai điểm trên đường cong E được định nghĩa như sau:

$$P_3(x_3, y_3) = P_1(x_1, y_1) + P_2(x_2, y_2) \tag{1.3}$$

Trong đó $P_3(x_3, y_3) = -P_3'(x_3, y_3')$, điểm $P_3'(x_3, y_3')$ là giao điểm của đường cong E và đường thẳng đi qua P_1 và P_2. Vì 2 điểm $P_3(x_3, y_3)$ và $-P_3'(x_3, y_3')$ đều nằm trên đường cong E nên (x_3, y_3) và (x_3, y_3') phải thỏa mãn phương trình (1.2). Công thức để tính các giá trị (x_3, y_3) sẽ được chứng minh ở dưới đây.



Hình 1.1: Phép cộng trên đường cong Elliptic

Trong các các tài liệu cơ bản và nâng cao được tham chiếu nhiều về đường cong Elliptic như [3, 7, 8] người viết vẫn chưa thỏa mãn với các dẫn dắt và chứng minh công thức tổng quát cho các giá trị (x_3, y_3), do đó các công thức này sẽ được chứng minh chi tiết trong tài liệu này. Đường thẳng đi qua 2 điểm P_1 và P_2 có phương trình là:

$$y = \lambda x + \mu \tag{1.4}$$

Trong đó λ là hệ số góc của đường thẳng đi qua P_1, P_2. Ta có:

$$y_1 = \lambda x_1 + \mu \tag{1.5}$$

$$y_2 = \lambda x_2 + \mu \tag{1.6}$$
$$y_3' = \lambda x_3 + \mu \tag{1.7}$$

1.3.1 Trường hợp 2 điểm không trùng nhau $P_1 \neq P_2$

Từ (1.5) và (1.6) suy ra: $y_1 - y_2 = \lambda(x_1 - x_2)$, khi $P_1 \neq P_2$, nghĩa là $x_1 \neq x_2$ ta có công thức:

$$\lambda = \frac{y_1 - y_2}{x_1 - x_2} \tag{1.8}$$
$$\mu = y_1 - \lambda x_1 = y_1 - \frac{y_1 - y_2}{x_1 - x_2} \times x_1 = \frac{x_1 y_2 - x_2 y_1}{x_1 - x_2} \tag{1.9}$$

Tiếp theo thay y ở (1.4) vào phương trình (1.2) ta có:

$$(\lambda x + \mu)^2 + (a_1 x + a_3)(\lambda x + \mu) = x^3 + a_2 x^2 + a_4 x + a_6 \tag{1.10}$$

Từ đó dẫn đến phương trình $r(x) = 0$ với:

$$r(x) = x^3 + (a_2 - \lambda^2 - a_1\lambda)x^2 + (a_4 - 2\lambda\mu - a_3\lambda - a_1\mu)x + a_6 - \mu^2 - a_3\mu \tag{1.11}$$

Biết rằng $r(x)$ có 3 nghiệm phân biệt nên có thể viết:

$$\begin{aligned} r(x) &= (x - x_1)(x - x_2)(x - x_3) \\ &= (x^2 - (x_1 + x_2)x + x_1x_2)(x - x_3) \\ &= x^3 - (x_1 + x_2)x^2 + x_1x_2x - x_3x^2 + (x_1 + x_2)x_3x - x_1x_2x_3 \\ &= x^3 - (x_1 + x_2 + x_3)x^2 + (x_1x_2 + x_1x_3 + x_2x_3)x - x_1x_2x_3 \end{aligned} \tag{1.12}$$

Đồng nhất các hệ số x^2 của $r(x)$ ở 2 phương trình (1.11) và (1.12) ta có: $x_1 + x_2 + x_3 = -(a_2 - \lambda^2 - a_1\lambda)$ từ đây có thể tính được x_3 theo công thức sau:

$$x_3 = \lambda^2 + a_1\lambda - a_2 - x_1 - x_2 \tag{1.13}$$

Đến đây cần phải tính tiếp giá trị y_3, lúc này x_3 đã tính xong nên có thể coi là hằng số, có thể viết lại (1.2) thành dạng sau:

$$y^2 + (a_1x_3 + a_3)y - (x_3^3 + a_2x_3^2 + a_4x_3 + a_6) = 0 \tag{1.14}$$

Phương trình bậc 2 này có 2 nghiệm là:

$$y_3, y_3' = \frac{-(a_1x_3 + a_3) \pm \sqrt{\Delta}}{2 \times 1} \tag{1.15}$$

Cộng 2 nghiệm này ta sẽ có: $y'_3 + y_3 = -a_1x_3 - a_3$, mặt khác do y'_3 nằm trên đường thẳng P_1, P_2 nên $y'_3 = \lambda x_3 + \mu$. Từ đây có thể tính được y_3 theo công thức [1]:

$$y_3 = -\lambda x_3 - \mu - a_1x_3 - a_3 \tag{1.16}$$

Thay μ từ (1.9) ta có thể tính y_3 dưới dạng sau:

$$y_3 = \lambda(x_1 - x_3) - y_1 - a_1x_3 - a_3 \tag{1.17}$$

1.3.2 Trường hợp 2 điểm trùng nhau $P_1 = P_2$

Khi này $x_1 = x_2$ và $y_1 = y_2$ do đó công thức tính λ ở (1.9) không sử dụng được vì xuất hiện phép chia số 0. Trong trường hợp này λ chính là hệ số góc của đường thẳng tiếp tuyến đường cong E tại P_1 hay P_2. Hệ số góc của tiếp tuyến của E chính là đạo hàm $\frac{dy}{dx}$, sử dụng các quy tắc lấy đạo hàm của tích, đạo hàm của hàm số hợp và lấy đạo hàm 2 vế của phương trình (1.2) theo dx ta có:

$$\begin{aligned}
\frac{d(y^2 + a_1xy + a_3y)}{dx} &= \frac{d(x^3 + a_2x^2 + a_4x + a_6)}{dx} \\
\frac{d(y^2)}{dx} + \frac{d(a_1xy)}{dx} + \frac{d(a_3y)}{dx} &= 3x^2 + 2a_2x + a_4 \\
\frac{d(y^2)}{dy} \times \frac{dy}{dx} + a_1(\frac{d(xy)}{dx}) + a_3\frac{dy}{dx} &= 3x^2 + 2a_2x + a_4 \\
2y\frac{dy}{dx} + a_1(y + x\frac{dy}{dx}) + a_3\frac{dy}{dx} &= 3x^2 + 2a_2x + a_4 \\
(2y + a_1x + a_3)\frac{dy}{dx} &= 3x^2 + 2a_2x + a_4 - a_1y \\
\frac{dy}{dx} &= \frac{3x^2 + 2a_2x + a_4 - a_1y}{2y + a_1x + a_3}
\end{aligned}$$

Như vậy với điểm $P_1(x_1, y_1)$ ta có:

$$\lambda = \frac{3x_1^2 + 2a_2x_1 + a_4 - a_1y_1}{2y_1 + a_1x_1 + a_3} \tag{1.18}$$

Trong tất cả các trường hợp điểm P_3 là tổng của 2 điểm P_1, P_2 sẽ là điểm có tọa độ là:

$$P_3(x_3, y_3) = (\lambda^2 + a_1\lambda - a_2 - x_1 - x_2, \lambda(x_1 - x_3) - y_1 - a_1x_3 - a_3) \tag{1.19}$$

[1]Ghi chú: Các tác giả H. Cohen và G. Frey trong cuốn "*Handbook of Elliptic and Hyperelliptic Curve Cryptography*" [7] trình bày diễn giải về cách tính y_3 dựa trên sự đối xứng qua trục x là không chính xác và thiếu rõ ràng đối với đường cong Elliptic dạng tổng quát, cách giải thích của người viết bằng cách giải phương trình bậc 2 ở đây có thể coi là đúng đắn và rõ ràng hơn.

Với đường cong E dạng (1.1), khi đó $a_1 = a_3 = a_2 = 0$ và P_3 sẽ được tính theo công thức:

$$P_3(x_3, y_3) = (\lambda^2 - x_1 - x_2, \lambda(x_1 - x_3) - y_1) \tag{1.20}$$

Trong trường hợp $P_1 = P_2$, (1.18) sẽ được biến đổi thành:

$$\lambda = \frac{3x_1^2 + a_4}{2y_1} \tag{1.21}$$

1.4 Nhân vô hướng các điểm trên đường cong Elliptic

Với $n \in \mathbb{N} \setminus \{0\}$ định nghĩa phép nhân vô hướng của điểm P nằm trên đường cong E là phép cộng n lần chính bản thân điểm P:

$$P \mapsto nP = \underbrace{P + P + \cdots + P}_{n \text{ lần}} = Q$$

Để tối ưu phép nhân vô hướng, có thể sử dụng phương pháp *Nhân đôi-và-cộng*, đầu tiên biểu diễn số n dưới dạng: $n = n_0 + 2n_1 + 2^2n_2 + \cdots + 2^m n_m$ với $[n_0 \dots n_m] \in \{0, 1\}$, sau đó áp dụng thuật toán:

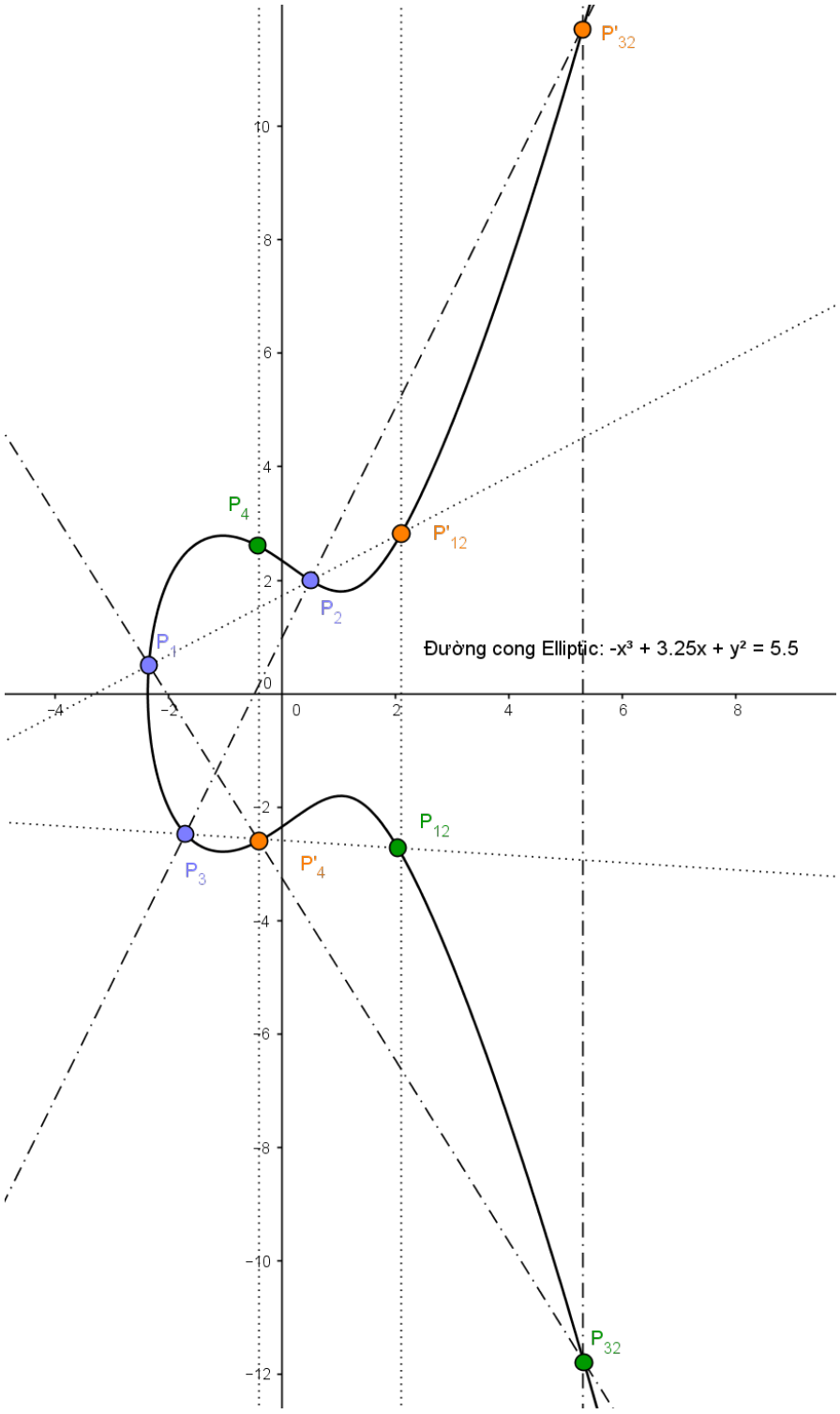
Thuật toán 1.1 Phương pháp Nhân đôi-và-cộng

```
1: Q ← 0
2: for i = 0 to m do
3:    if n_i = 1 then
4:       Q ← CộngĐiểm(Q,P)
5:    end if
6:    P ← NhânĐôi(P)
7: end for
8: return Q
```

Ngoài phương pháp *Nhân đôi-và-cộng*, có thể sử dụng phương pháp *Trượt-cửa-sổ*. Các phương pháp này cho phép nhân vô hướng một cách tối ưu.

Lưu ý của người viết:

- Không tồn tại phép nhân 2 điểm trên đường cong E, có nghĩa là không tồn tại $P \times Q$ với $P, Q \in E$.
- Không tồn tại thuật toán chia vô hướng $Q : n$. Biết rằng $Q = nP$, bài toán tìm số n là bài toán Logarithm rời rạc sẽ được đề cập tới ở chương sau. Đây



Hình 1.2: Ví dụ về tính chất kết hợp trên đường cong Elliptic

là bài toán khó, thông thường phải thử lần lượt $n = 1, 2, \ldots, n-1$ phép cộng điểm P, cho đến khi tổng bằng Q, tuy nhiên có một số thuật toán tối ưu hơn

để tìm n nhưng vẫn không thể giải được bài toán này trong thời gian đa thức vì thế dựa vào độ khó này có thể xây dựng ra hệ mật đường cong Elliptic với các giao thức cho mã hóa, xác thực và trao đổi khóa.

1.5 Nhóm (+) của các điểm trên đường cong Elliptic

Xét đường cong Elliptic E được định nghĩa bởi phương trình $y^2 = x^3 + Ax + B$. Xét 3 điểm nằm trên đường cong E là P_1, P_2, P_3 lần lượt có các tọa độ là $(x_1, y_1), (x_2, y_2), (x_3, y_3)$.

Để các điểm trên đường cong Elliptic tạo thành nhóm (+), "điểm vô cùng" (∞) sẽ được thêm vào đường cong, kí hiệu là $\mathcal{O}$, điểm này sẽ nằm ở trên cùng và dưới cùng của trục y. Một trong những thuộc tính quan trọng nhất của đường cong Elliptic là tồn tại nhóm các điểm với phép cộng nằm trên đường cong.

Định lý 1.5.1. *Phép cộng với các điểm P, P_1, P_2, P_3 trên đường cong E thỏa mãn các tính chất của nhóm:*

1. *(Giao hoán): $P_1 + P_2 = P_2 + P_1$;*
2. *(Điểm đơn vị): $P + \infty = P$;*
3. *(Điểm nghịch đảo): Tồn tại P' của P sao cho $P + P' = \infty$;*
4. *(Kết hợp): $(P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$.*

Chứng minh. (*1. Tính chất giao hoán*) của phép cộng 2 điểm P_1, P_2 là hiển nhiên từ công thức tính tọa độ của điểm tổng, các giá trị có giao hoán thì giá trị tính bởi công thức này cũng không thay đổi, hoặc về mặt hình học đường thẳng đi qua P_1, P_2 dù có xuất phát từ P_1 hay P_2 thì đều như nhau và cùng cắt đường cong E tại một điểm chung duy nhất.

(*2. Điểm đơn vị*) không cần phải chứng minh vì nó xuất phát từ định nghĩa. Có thể lý giải rõ hơn về điểm (∞) và cách định nghĩa phép cộng trên E (theo quan điểm của người viết) như sau: Khi đường cong E không suy biến, nó sẽ cắt một đường thẳng được định nghĩa bởi phương trình (1.4) ở 3 điểm, thực vậy theo các phép biến đổi ở mục 1.3.1 (trang 11) phương trình (1.11) $r(x)$ sẽ có 3 nghiệm phân biệt. Mặt khác E đối xứng qua trục x do phương trình (1.1) có thành phần y^2 nên luôn tồn tại hai giá trị $y, -y$ thỏa mãn (1.1), cũng do tính đối xứng này nên đường cong E sẽ cắt các đường thẳng song song với với trục y ở 2 điểm, vì nếu cắt thêm 1 điểm nữa thì sẽ phải cắt thành 4 điểm do tính đối xứng, điều này là mẫu thuẫn vì phương trình bậc 3 chỉ có tối đa 3 nghiệm. Ở trường hợp này nếu cộng 2 điểm nằm trên đường song song trục y sẽ không tìm được điểm thứ 3 do vậy $P_1 + P_2$ sẽ

không tồn tại. Chính vì để nhóm các điểm trên E có tính đóng bắt buộc chúng ta phải định nghĩa thêm điểm ∞ coi như là điểm thứ 3 nằm trên đường cong E, và nó sẽ nằm ở vô cực ở 2 đầu trục y.

Tiếp theo, có thể lý giải (của người viết[2]) về phép định nghĩa phép cộng 2 điểm trên E như sau. Để thỏa mãn tính chất tồn tại điểm đơn vị theo định nghĩa về nhóm G với mọi giá trị $a \in G$ tồn tại $e \in G$ sao cho

$$a \bullet e = e \bullet a = a \tag{1.22}$$

Xét điểm P trên đường cong E, khi đó cần tính $P + \infty$, dễ thấy điểm này chắc chắn phải nằm trên cùng đường thẳng song song trục y vì nếu không sẽ cắt E ở 2 điểm nữa cùng với ∞ tạo thành 4 điểm và điều này là phi lý. Nếu đã nằm trên đường song song y thì nó sẽ cắt E ở điểm đối xứng qua trục x, nếu coi điểm cắt này là tổng $P + \infty$ thì như vậy sẽ tồn tại $P + \infty = P'$ và điều kiện (1.22) sẽ không được thỏa mãn. Vì vậy sẽ phải định nghĩa điểm tổng không phải là giao trực tiếp với E mà phải lấy là điểm đối xứng để đối xứng của điểm đối xứng sẽ quay lại chính P và ta có $P + \infty = P$.

(*3.Điểm nghịch đảo*) Cũng từ nhận xét rằng luôn tồn tại 2 điểm P, P' nằm trên cùng đường thẳng song song với trục y và sẽ cắt đường cong E ở điểm ∞, coi 2 điểm này là nghịch đảo của nhau và sẽ luôn có: $P + P' = \infty$.

(*4. Tính chất kết hợp*) Chứng minh $(P_1 + P_2) + P_3 = P_1 + (P_2 + P_3)$ khác hẳn với 3 điều kiện khác về nhóm, và nó đặc biệt phức tạp. Có 2 cách chứng minh điều này là dùng phương pháp hình học hoặc đại số. Có thể tham khảo chứng minh bằng hình học qua các tài liệu [3], [17] và [4], tuy nhiên chứng minh bằng phương pháp hình học tương đối khó hiểu với một số định lý trong không gian xạ ảnh. Dưới đây là một số điểm được sử dụng để chứng minh bằng phương pháp đại số. Trước tiên tính các điểm tổng sau:

$$\begin{aligned} P_{12} &= P_1 + P_2 \\ P_{32} &= P_3 + P_2 \\ P_{123} &= (P_1 + P_2) + P_3 = P_{12} + P_3 \\ P_{321} &= (P_3 + P_2) + P_1 = P_{32} + P_1 \end{aligned}$$

Từ các công thức tính giá trị tọa độ (x, y) của điểm tổng, có thể dễ dàng biến

[2]Khi bắt đầu nghiên cứu về đường cong Elliptic, luôn có 2 câu hỏi mà người viết không thể tìm thấy trong nhiều tài liệu kể cả những cuốn kinh điển về Elliptic:

1. Tại sao phải chọn ∞ làm điểm trung hòa.
2. Tại sao $P_1 + P_2 = P_3$ mà P_3 không nằm trên đường thẳng đi qua P_1, P_2 mà phải là điểm đối xứng của giao điểm qua trục x.

Trong chuyên đề này cả 2 câu hỏi đều đã được người viết lý giải một cách rõ ràng và đầy đủ.

đổi tính điểm:

$$\lambda_{12} = \frac{(y_2 - y_1)}{(x_2 - x_1)} \tag{1.23}$$

$$\lambda_{32} = \frac{(y_2 - y_3)}{(x_2 - x_3)} \tag{1.24}$$

$$x_{12} = \lambda_{12}^2 - x_1 - x_2 = \frac{(y_2 - y_1)^2}{(x_2 - x_1)^2} - x_1 - x_2 \tag{1.25}$$

$$y_{12} = \lambda_{12}(x_1 - x_{12}) - y_1 = -\lambda_{12}^3 + (2x_1 + x_2)\lambda_{12} - y_1 \tag{1.26}$$

$$= -\frac{(y_2 - y_1)^3}{(x_2 - x_1)^3} + \frac{(2x_1 + x_2)(y_2 - y_1)}{x_2 - x_1} - y_1$$

$$x_{32} = \lambda_{32}^2 - x_3 - x_2 = \frac{(y_2 - y_3)^2}{(x_2 - x_3)^2} - x_3 - x_2 \tag{1.27}$$

$$y_{32} = \lambda_{32}(x_3 - x_{32}) - y_3 = -\lambda_{32}^3 + (2x_3 + x_2)\lambda_{32} - y_3 \tag{1.28}$$

$$= -\frac{(y_2 - y_3)^3}{(x_2 - x_3)^3} + \frac{(2x_3 + x_2)(y_2 - y_3)}{x_2 - x_3} - y_3$$

$$\lambda_{123} = \frac{(y_{12} - y_3)}{(x_{12} - x_3)} \tag{1.29}$$

$$\lambda_{321} = \frac{(y_{32} - y_1)}{(x_{32} - x_1)} \tag{1.30}$$

Cuối cùng cách tính tọa độ của các điểm P_{123}, P_{321} theo tọa độ 3 điểm $P_1(x_1, y_1), P_2(x_2, y_2), P_3(x_3, y_3)$ được biểu diễn dưới các dạng công thức từ (1.31) đến (1.34).

$$x_{123} = \frac{\left(-\frac{(y_2 - y_1)^3}{(x_2 - x_1)^3} + \frac{(2x_1 + x_2)(y_2 - y_1)}{x_2 - x_1} - y_1 - y_3\right)^2}{\left(\frac{(y_2 - y_1)^2}{(x_2 - x_1)^2} - x_1 - x_2 - x_3\right)^2} - \frac{(y_2 - y_1)^2}{(x_2 - x_1)^2} + x_1 + x_2 - x_3 \tag{1.31}$$

$$y_{123} = \frac{\left(-\frac{(y_2 - y_1)^3}{(x_2 - x_1)^3} + \frac{(2x_1 + x_2)(y_2 - y_1)}{x_2 - x_1} - y_1 - y_3\right)^3}{\left(\frac{(y_2 - y_1)^2}{(x_2 - x_1)^2} - x_1 - x_2 - x_3\right)^3} \tag{1.32}$$

$$+ \frac{\left(\frac{2(y_2 - y_1)^2}{(x_2 - x_1)^2} - 2x_1 - 2x_2 + x_3\right)\left(-\frac{(y_2 - y_1)^3}{(x_2 - x_1)^3} + \frac{(2x_1 + x_2)(y_2 - y_1)}{x_2 - x_1} - y_1 - y_3\right)}{\frac{(y_2 - y_1)^2}{(x_2 - x_1)^2} - x_1 - x_2 - x_3}$$

$$+ \frac{(y_2 - y_1)^3}{(x_2 - x_1)^3} - \frac{(2x_1 + x_2)(y_2 - y_1)}{x_2 - x_1} + y_1$$

$$x_{321} = \frac{\left(-\frac{(y_2-y_3)^3}{(x_2-x_3)^3} + \frac{(2x_3+x_2)(y_2-y_3)}{x_2-x_3} - y_3 - y_1\right)^2}{\left(\frac{(y_2-y_3)^2}{(x_2-x_3)^2} - x_3 - x_2 - x_1\right)^2} - \frac{(y_2-y_3)^2}{(x_2-x_3)^2} + x_3 + x_2 - x_1 \tag{1.33}$$

$$y_{321} = \frac{\left(-\frac{(y_2-y_3)^3}{(x_2-x_3)^3} + \frac{(2x_3+x_2)(y_2-y_3)}{x_2-x_3} - y_3 - y_1\right)^3}{\left(\frac{(y_2-y_3)^2}{(x_2-x_3)^2} - x_3 - x_2 - x_1\right)^3} \tag{1.34}$$
$$+ \frac{\left(\frac{2(y_2-y_3)^2}{(x_2-x_3)^2} - 2x_3 - 2x_2 + x_1\right)\left(-\frac{(y_2-y_3)^3}{(x_2-x_3)^3} + \frac{(2x_3+x_2)(y_2-y_3)}{x_2-x_3} - y_3 - y_1\right)}{\frac{(y_2-y_3)^2}{(x_2-x_3)^2} - x_3 - x_2 - x_1}$$
$$+ \frac{(y_2-y_3)^3}{(x_2-x_3)^3} - \frac{(2x_3+x_2)(y_2-y_3)}{x_2-x_3} + y_3$$

Cần phải chứng minh rằng $P_{123} = P_{321}$ điều này có nghĩa cần phải chứng minh:

$$dx = x_{123} - x_{321} = 0 \tag{1.35}$$
$$dy = y_{123} - y_{321} = 0 \tag{1.36}$$

Triển khai vế trái của (1.35), sẽ được một phân số mà tử số và mẫu số bao gồm tổng cộng 1446 thành phần, tương tự vế trái của (1.36) có tất cả 10081 thành phần có dạng $n_k x_1^{i_1} x_2^{i_2} x_3^{i_3} y_1^{j_1} y_2^{j_2} y_3^{j_3}$, trong đó số mũ $i_1, i_2, i_3, j_1, j_2, j_3$ nằm trong khoảng $[0..12]$ và n_k là hệ số của thành phần biểu thức trên. Có thể kiểm tra tính đúng đắn của (1.35) và (1.36) bằng phần mềm Maple với các ràng buộc sau:

$$y_1^2 = x_1^3 + Ax_1 + B \tag{1.37}$$
$$y_2^2 = x_2^3 + Ax_2 + B \tag{1.38}$$
$$y_3^2 = x_3^3 + Ax_3 + B \tag{1.39}$$

■

1.6 Đường cong Elliptic trên trường hữu hạn $\mathbb{F}_q$

1.6.1 Trường hữu hạn $\mathbb{F}_q$

Các ứng dụng về mật mã của đường cong Elliptic đa số chỉ sử dụng các đường cong trên trường hữu hạn.

Xét $\mathbb{F}_q$ là một trường hữu hạn (hữu hạn số phần tử số nguyên dương):

$$\mathbb{F}_q = \{0, 1, 2 \ldots, q-1\}$$

q là một số nguyên tố hoặc có dạng $q = p^m$ với p là một số nguyên tố và m là một số nguyên dương. Khi này p được gọi là đặc số $char(q) = p$ và m là bậc mở rộng của $\mathbb{F}_q$.

Trong thực tế và đặc biệt trong các thiết bị phần cứng [18], người ta thường sử dụng trường hữu hạn $\mathbb{F}_{2^m}$. Khi đó phép cộng trong trường này đơn giản chỉ là phép toán XOR (Exclusive OR). Nhiều tài liệu cho thấy làm việc với $\mathbb{F}_{2^m}$ hiệu quả hơn 40% so với làm việc với trường $\mathbb{F}_q$. Nhóm thực hiện Đề tài cấp Nhà nước KC01.18 do người viết làm hủ nhiệm đề tài đã cài đặt toàn bộ các phép toán về đường cong Elliptic trên trường $\mathbb{F}_{2^m}$ cho Chip Spartan 6 của Xilinx cho bài toán xác thực và trao đổi khóa phiên trong thiết bị VPN IPSec.

Trường $\mathbb{F}_{2^m}$ thường được biểu diễn dưới dạng tổ hợp tuyến tính của các vector gồm m phần tử $\{\alpha_0, \alpha_1, \ldots, \alpha_{m-1}\}$, mọi phần tử $\alpha \in \mathbb{F}_{2^m}$ đều có thể được biển diễn dưới dạng:

$$\alpha = a_0\alpha_0 + a_0\alpha_0 + \ldots + a_{m-1}\alpha_{m-1}, \quad a_i \in \{0, 1\}$$

Có nhiều phương pháp để xây dựng cơ sở của $\mathbb{F}_{2^m}$: đa thức cơ sở và cơ sở chuẩn tắc. Các thuật toán để thực hiện các phép toán trên EC có thể tìm thấy trong [19].

1.6.1.1 Đa thức cơ sở

Xét đa thức $f(x) = x^m + f_{m-1}x^{m-1} + \ldots + f_2x^2 + f_1x + f_0$ (với $f_i \in \mathbb{F}_2, i = 0, \ldots, m-1$) là một đa thức bất khả quy bậc m trên trường $\mathbb{F}_2$, nghĩa là không thể phân tích $f(x)$ thành các đa thức thừa số khác có bậc nhỏ hơn m. $f(x)$ gọi là đa thức rút gọn. Trường hữu hạn $\mathbb{F}_{2^m}$ sẽ là tập tất cả các đa thức trên F_2 có bậc nhỏ hơn hoặc bằng m.

$$F_{2^m} = \{a_{m-1}x^{m-1} + \ldots + a_2x^2 + a_1x + a_0 : a_i \in \{0, 1\}\}$$

Các phần tử $(a_{m-1}x^{m-1} + \ldots + a_2x^2 + a_1x + a_0)$ thường được biểu diễn dưới dạng chuỗi bit $(a_{m-1} \ldots a_1a_0)$ có độ dài là m.

Các phép toán trong trường $\mathbb{F}_{2^m}$:

- Phép cộng:
  $(c_{m-1} \ldots c_1c_0) = (a_{m-1} \ldots a_1a_0) + (b_{m-1} \ldots b_1b_0), \quad c_i = a_i \oplus b_i.$
- Phép nhân:
  $(r_{m-1} \ldots r_1r_0) = (a_{m-1} \ldots a_1a_0).(b_{m-1} \ldots b_1b_0)$
  Trong đó $(r_{m-1}x^{m-1} + \ldots + r_1x + r_0) = (a_{m-1}x^{m-1} + \ldots + a_1x + a_0) \times (b_{m-1}x^{m-1} + \ldots + b_1x + b_0) \mod f(x)$.

Đa thức rút gọn $f(x)$ thường có dạng sau:

- Trinomial basis (TPB):
  $f(x) = x^m + x^k + 1, \quad 1 \leq k \leq m-1.$
- Pentanomial basis (PPB):
  $f(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1, \quad 1 \leq k_1 < k_2 < k_3 \leq m-1.$

1.6.1.2 Cơ sở chuẩn tắc

$\mathbb{F}_{2^m}$ sử dụng cơ sở có dạng $\{\beta, \beta^{2^2}, \ldots, \beta^{2^{m-1}}\}$ với $\beta \in \mathbb{F}_{2^m}$, khi đó mỗi phần tử $a \in \mathbb{F}_{2^m}$ đều có dạng $a = \sum_{i=0}^{m-1} a_i \beta^{2^i}, a_i \in \{0,1\}$ và cũng được biểu diễn dưới dạng chuỗi bit $(a_0 a_1 \ldots a_{m-1})$ có độ dài là m. Với cơ sở này phép bình phương sẽ thực hiện rất đơn giản chỉ bằng cách quay bit. Các phép toán trong trường $\mathbb{F}_{2^m}$:

- Phép cộng:
  $(c_0 c_1 \ldots c_{m-1}) = (a_0 a_1 \ldots a_{m-1}) + (b_0 b_1 \ldots b_{m-1}), \quad c_i = (a_i + b_i) \mod 2.$
- Phép bình phương:

$$a^2 = \left(\sum_{i=0}^{m-1} a_i \beta^{2^i}\right)^2 = \sum_{i=0}^{m-1} a_i \beta^{2^{i+1}} = \sum_{i=0}^{m-1} a_{i-1} \beta^{2^i} = (a_{m-1} a_0 a_1 \ldots a_{m-2})$$

- Phép nhân:
  Xét $p = Tm + 1$ và $u \in \mathbb{F}_p$ là phần tử bậc T, định nghĩa chuỗi $F(1), F(2), \ldots, F(p-1)$ ta sẽ có:

$$F(2^i u^j \mod p) = i, \quad 0 \leq i \leq m-1, 0 \leq j \leq T-1.$$
$$(c_0 c_1 \ldots c_{m-1}) = (a_0 a_1 \ldots a_{m-1}).(b_0 b_1 \ldots b_{m-1})$$
$$c_l = \begin{cases} \sum_{k=1}^{p=2} a_{F(k+1)+l} b_{F(p-k)+l}, & \text{nếu } T \text{ chẵn.} \\ \sum_{k=1}^{m/2} (a_{k+l-1} b_{m/2+k+l-1} + a_{m/2+k+l-1} b_{k+l-1}) & \\ + \sum_{k=1}^{p=2} a_{F(k+1)+l} a_{F(k+1)+l}, & \text{nếu } T \text{ lẻ.} \end{cases}$$

1.6.2 Tổng số điểm của đường cong Elliptic trên trường hữu hạn $\mathbb{F}_q$

E là đường cong Elliptic trên trường $\mathbb{F}_q$, bởi vì cặp (x, y) với $x, y \in \mathbb{F}_q$ là hữu hạn do đó nhóm $E(\mathbb{F}q)$ cũng sẽ là nhóm hữu hạn. Các giá trị x, y là các số nguyên, dễ dàng nhận thấy không phải với mọi giá trị x đều tìm được giá trị nguyên y bởi vì không phải bao giờ y^2 cũng là một số nguyên dương. Câu hỏi đặt ra là số điểm

của của đường cong Elliptic trên trường $\mathbb{F}_q$ là bao nhiêu? Xác định số điểm trên đường cong E nhằm xác định không gian khóa của hệ mật.

Sau đây là phần trình bày về việc tính tổng số điểm của đường cong Elliptic trên trường hữu hạn $\mathbb{F}_q$.

Bổ đề 1.6.1. *M và N là hai ma trận 2×2. $M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}, N = \begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix}$, với các số nguyên a, b ta có:*

$$\det(aM + bN) = a^2 \det(M) + b^2 \det(N) + ab(\det(M + N) - \det(M) - \det(N)) \quad (1.40)$$

Chứng minh. Theo định nghĩa về định thức ta có:

$$\begin{aligned} \det(M) &= m_{11}m_{22} - m_{21}m_{12}, \quad \det(N) = n_{11}n_{22} - n_{21}n_{12} \\ \det(aM + bN) &= \det \begin{pmatrix} am_{11} + bn_{11} & am_{12} + bn_{12} \\ am_{21} + bn_{21} & am_{22} + bn_{22} \end{pmatrix} \\ &= (am_{11} + bn11)(am_{22} + bn_{22}) - (am_{21} + bn_{21})(am_{12} + bn_{12}) \\ &= a^2(m_{11}m_{22} - m_{21}m_{12}) + b^2(n_{11}n_{22} - n_{21}n_{12}) + \\ &+ ab(m_{11}n_{22} + n_{11}m_{22} - m_{21}n_{12} - n_{21}m_{12}) \\ &= a^2 \det(M) + b^2 \det(N) + ab(m_{11}n_{22} + n_{11}m_{22} - m_{21}n_{12} - n_{21}m_{12}) \end{aligned} \quad (1.41)$$

Khi $a = b = 1$ áp dụng công thức ở trên ta có:

$$\det(M + N) = \det(M) + \det(N) + (m_{11}n_{22} + n_{11}m_{22} - m_{21}n_{12} - n_{21}m_{12}) \quad (1.42)$$

Nhân cả 2 vế (1.42) với ab sau đó trừ vào (1.41) sẽ được kết quả (1.40) là điều cần phải chứng minh. ■

Định nghĩa 1.6.2. *Điểm n-xoắn (Torsion): Cho đường cong Elliptic E được định nghĩa trên trường K, cho n là số nguyên dương, tập các điểm Torsion $E[n]$ là tập các điểm trên đường cong có tính chất như sau:*

$$E[n] = \left\{ P \in E(\overline{K}) \mid nP = \infty \right\} \quad (1.43)$$

$\overline{K}$ là đóng đại số của K.

Định nghĩa 1.6.3. *Divisor: Cho đường cong Elliptic E được định nghĩa trên trường K, với mỗi điểm $P \in E(\overline{K})$ định nghĩa một ký hiệu hình thức (formal symbol) $[P]$, khi đó divisor D sẽ là:*

$$D = \sum_j a_j[P_j], \quad a_j \in \mathbb{Z}$$

f là một hàm trên E mà khác 0, khi đó divisor của f sẽ là:

$$div(f) = \sum_{P \in E(K)} ord_P(f)[P] \in Div(E)$$

$f = u_P^r g$, với $r \in \mathbb{Z}$ và $g(P) \neq 0, \infty$, $u(P) = 0$, định nghĩa bậc của f tại P là:

$$ord_P(f) = r.$$

Định nghĩa 1.6.4. *Cặp Weil: là một ánh xạ từ 2 điểm trong nhóm các điểm Torsion thành giá trị bậc thứ n của đơn vị:*

$$e_n : E[n] \times E[n] \to \mu_n, \quad \mu_n = \{x \in \overline{K} \mid x^n = 1\}$$

Bổ đề 1.6.5. *Với mọi tự đồng cấu bất khả tách α trên E, và với mọi $S, S_1, S_2, T, T_1, T_2 \in E[n]$ ta có:*

$$\begin{aligned} e_n(\alpha(S), \alpha(T)) &= e_n(S,T)^{\deg(\alpha)} \\ e_n(T,T) &= 1 \\ e_n(S_1 + S_2, T) &= e_n(S_1, T) e_n(S_2, T) \\ e_n(S, T_1 + T_2) &= e_n(S, T_1) e_n(S, T_2) \end{aligned}$$

Chứng minh có thể xem trong [3].

Giả thiết $\{T_1, T_2\}$ là cơ sở của $E[n]$, mỗi phần tử trong $E[n]$ đều có thể biểu diễn dưới dạng tổ hợp tuyến tính $m_1 T_1 + m_2 T_2$. α là một tự đồng cấu trong $E[n]$, n là một số nguyên không chia hết bởi $char(K)$. Tồn tại các số $a, b, c, d \in \mathbb{Z}$ sao cho:

$$\alpha(T_1) = aT_1 + cT_2, \quad \alpha(T_2) = bT_1 + dT_2$$

Do đó mỗi tự đồng cấu α đều có thể được biểu diễn bởi ma trận 2×2:

$$\alpha_n = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Bổ đề 1.6.6. *α là một tự đồng cấu trong $E[n]$, n là một số nguyên không chia hết bởi $char(K)$ khi đó* $\det(\alpha_n) \equiv \deg(\alpha) \mod n$.

Chứng minh. Đặt $\zeta = e_n(T_1, T_2)$, theo bổ đề 1.6.5 ta có:

$$\begin{aligned} \zeta^{\deg(\alpha)} &= e_n(\alpha(T_1), \alpha(T_2)) = e_n(aT_1 + cT_2, bT_1 + dT_2) \\ &= e_n(T_1, T_1)^{ab} e_n(T_1, T_2)^{ad} e_n(T_2, T_1)^{cb} e_n(T_2, T_2)^{cd} \\ &= \zeta^{ad-bc} = \zeta^{\det(\alpha_n)} \end{aligned}$$

■

Nếu α, β là 2 tự đồng cấu trên E, và a, b là các số nguyên thì tự đồng cấu $a\alpha + b\beta$ được định nghĩa như sau:

$$(a\alpha + b\beta)(P) = a\alpha(P) + b\beta(P)$$

Bổ đề 1.6.7. $\deg(a\alpha + b\beta) = a^2 \deg \alpha + b^2 \deg \beta + ab(\deg(\alpha + \beta) - \deg \alpha - \deg \beta)$

Chứng minh. Biểu diễn các tự đồng cấu α, β bằng các ma trận α_n, β_n (với một số cơ sở trong $E[n]$), theo đó $a\alpha + b\beta$ sẽ được biểu diễn bằng $a\alpha_n + b\beta_n$. Áp dụng công thức (1.40) ta có:

$$\det(a\alpha_n + b\beta_n) = a^2 \det(\alpha_n) + b^2 \det(\beta_n) + ab(\det(\alpha_n + \beta_n) - \det(\alpha_n) - \det(\beta_n))$$

Theo bổ đề 1.6.6 chúng ta sẽ có:

$$\deg(a\alpha + b\beta) = a^2 \deg(\alpha) + b^2 \deg(\beta) + ab(\deg(\alpha + \beta) - \deg(\alpha) - \deg(\beta))$$

■

Định lý 1.6.8. *(Hasse) Nếu E là đường cong Elliptic trên trường $\mathbb{F}_q$, và $\#E(\mathbb{F}q)$ là tổng số điểm trên đường cong đó thì:*

$$q + 1 - 2\sqrt{q} \leq \#E(\mathbb{F}q) \leq q + 1 + 2\sqrt{q}. \tag{1.44}$$

Chứng minh. Trước tiên xét ánh xạ Probenius được định nghĩa như sau:

$$\begin{aligned} \phi_q : \overline{\mathbb{F}}_q &\longrightarrow \overline{\mathbb{F}}_q, \\ x &\mapsto x^q \end{aligned}$$

Có thể viết một cách khác:

$$\phi_q(x, y) = (x^q, y^q), \quad \phi_q(\infty) = \infty$$

Khi thay các giá trị x^q, y^q vào phương trình (1.1) dễ thấy (x, y) cũng nằm trên đường cong E. Ánh xạ ϕ_q là một tự đồng cấu và có thể biểu diễn bằng hàm đa thức hữu tỷ có bậc là q. Đạo hàm của x^q là qx^{q-1} sẽ bằng 0 bởi vì $q = 0$ trong trường $\mathbb{F}_q$. Do đạo hàm bằng 0 nên ϕ_q là khả tách (separable).

Bởi vì ϕ_q là tự đồng cấu trong E do đó $\phi_q^2 = \phi_q \circ \phi_q$ cũng là tự đồng cấu và ϕ_q^n cũng là tự đồng cấu trong E. Phép nhân với -1 cũng là tự đồng cấu do đó tổng $\phi_q^n - 1$ là đồng cấu trong E.

ϕ_q là khả tách (separable) nhưng $\phi_q - 1$ sẽ là bất khả tách do đó bậc của nó sẽ bằng số phần tử của hạch $\phi_q - 1$ có nghĩa là số điểm trên đường cong E sẽ là:

$$\#E(\mathbb{F}_q) = \deg(\phi_q - 1)$$

Với các số nguyên r, s, áp dụng bổ đề 1.6.7 ta có:

$$\deg(r\phi_q - s) = r^2 \deg(\phi_q) + s^2 \deg(-1) + rs(\deg(\phi_q - 1) - \deg(\phi_q) - \deg(-1))$$

Bởi vì $\deg(-1) = 1$ và $\deg(\phi_q) = q$ nên:

$$\deg(r\phi_q - s) = r^2 q + s^2 + rs(\deg(\phi_q - 1) - q - 1))$$

Đặt $a = -(\deg(\phi_q - 1) - q - 1) = q + 1 - \#E(\mathbb{F}_q)$, bởi vì $\deg(r\phi_q - s) \geq 0$ suy ra $r^2 q + s^2 + rsa \geq 0$ hay với mọi r, s ta có:

$$q\left(\frac{r}{s}\right)^2 - a\left(\frac{r}{s}\right) + 1 \geq 0$$

Do đó $\Delta = a^2 - 4q \leq 0$ hay là $|a| \leq 2\sqrt{q}$ cũng có nghĩa là $|q + 1 - \#E(\mathbb{F}_q)| \leq 2\sqrt{q}$ và đó là điều phải chứng minh. ■

Tham khảo thêm về cách tính số điểm trên đường cong E có thể xem trong [20].

Chương 2

Hệ mật dựa trên đường cong Elliptic và một số ứng dụng

2.1 Bài toán Logarithm rời rạc

Định nghĩa 2.1.1. Bài toán Logarithm rời rạc trên đường cong Elliptic (ECDLP): *Cho đường cong E trên trường hữu hạn $\mathbb{F}_q$, điểm $P \in E(\mathbb{F}_q)$ với bậc n ($nP = \mathcal{O} = \infty$) và điểm $Q \in E(\mathbb{F}_q)$, tìm số nguyên $k \in [0, n-1]$ sao cho $Q = kP$. Số nguyên k được gọi là Logarithm rời rạc của Q với cơ sở P, và thường được viết là $k = \log_P Q$.*

Bất kỳ một hệ mật khóa công khai nào cũng phải sử dụng một bài toán khó để xây dựng hàm một chiều. Ý nghĩa một chiều ở đây có nghĩa là tính thuận thì dễ (thuật toán giải trong thời gian đa thức) và tính ngược thì khó (thuật toán giải với thời gian không phải là đa thức - thường là hàm mũ hoặc nửa mũ). Các tham số của Hệ mật dựa trên đường cong Elliptic (ECC) cần phải được lựa chọn cẩn thận để tránh được các tấn công đối với bài toán ECDLP. Thuật toán vét cạn để giải bài toán ECDLP là lần lượt tính thử các điểm $P, 2P, 3P, ...$ cho đến khi điểm mới tính được đúng bằng điểm Q. Trong trường hợp xấu nhất sẽ phải cần đến n bước thử, trung bình thường là $n/2$ là đạt được điểm Q, do đó cần phải chọn n đủ lớn

để bài toán vét cạn là không khả thi ($n \geq 2^{160}$).

Thuật toán tốt nhất hiện nay để tấn công bài toán ECDLP là sự kết hợp của thuật toán Pohlih-Hellman và Pollard's rho, thuật toán này có thời gian tính là $O(\sqrt{p})$, với p là ước số nguyên tố lớn nhất của n do đó phải chọn số n sao cho nó chia hết số nguyên tố p lớn nhất có $\sqrt{p}$ đủ lớn để giải bài toán này là không khả thi.

Trong phần tiếp theo, một số phương pháp tấn công bài toán Logarithm rời rạc sẽ được trình bày, đa số các phương pháp này có thể áp dụng được cho một nhóm bất kỳ. Chi tiết có thể tham khảo trong [3, 8, 21].

Cho G là nhóm các điểm trên đường cong E. $P, Q \in G$ là các điểm trên đường cong E, chúng ta cần giải bài toán $kP = Q$, N là bậc của G.

2.1.1 Phương pháp bước nhỏ, bước lớn

Phương pháp này do Shanks đề xuất và được H. Cohen mô tả trong [22].

Thuật toán 2.1 Phương pháp bước nhỏ, bước lớn

1: Chọn $m \geq \sqrt{N}$ và tính mP.
2: Tính và lưu trữ danh sách iP với $0 \leq i < m$
3: Tính $Q - jmP$ với $j = 0, 1, \ldots, m-1$
4: **if** $iP = Q - jmP$ **then**
5: $\quad k = i + jm(\mod N)$
6: **end if**
7: Quay về bước 3

Dễ dàng nhận thấy $Q = iP + jmP$ hay $Q = (i + jm)P$ từ đó $k = i + jm$. Điểm iP được tính bằng cách cộng thêm P vào $(i-1)P$ và giá trị này được gọi là bước nhỏ. $Q - jmP$ được tính bằng cách cộng thêm mP vào $Q - (j-1)mP$ và giá trị này được gọi là bước lớn.

2.1.2 Phương pháp Pollard's ρ và λ

Phương pháp này do Pollard đề xuất trong [23].

Định nghĩa hàm $f : G \to G$ một cách ngẫu nhiên $P_{i+1} = f(P_i)$ với P_0 cũng được chọn một cách ngẫu nhiên. Bởi vì G là tập hữu hạn do đó sẽ có các chỉ số $i_0 < j_0$ mà $P_{i_0} = P_{j_0}$, từ đó ta có:

$$P_{i_0+1} = f(P_{i_0}) = f(P_{j_0}) = P_{j_0+1}$$

Tương tự sẽ có $P_{i_0+l} = P_{j_0+l}$ với $l \geq 0$, từ đó chuỗi P_i là chuỗi tuần hoàn với chu

kỳ là $j_0 - i_0$. Hàm biểu diễn chuỗi P_i thường giống chữ cái Hi Lạp ρ và đó là lý do tại sao phương pháp này có tên là phương pháp ρ.

Hàm f được chọn như sau: Chia tập G thành s tập con không trùng nhau $S_1, S_2, \ldots, S_s$ có kích thước tương đương nhau, s thường được chọn là 20, chọn $2s$ số ngẫu nhiên $a_i, b_i \mod N$. Đặt:

$$M_i = a_i P + b_i Q$$

Và định nghĩa:

$$f(g) = g + M_i, \quad g \in S_i$$

Biểu diễn P_j dưới dạng $P_j = u_j P + v_j Q$, khi $P_{i_0} = P_{j_0}$ ta có:

$$\begin{aligned} u_{j_0} P + v_{j_0} Q &= u_{i_0} P + v_{i_0} Q \\ (u_{i_0} - u_{j_0}) P &= (v_{j_0} - vx`_{j_0}) Q \\ k &= (v_{j_0} - vx`_{j_0})_1 (u_{i_0} - u_{j_0}) \pmod N \end{aligned}$$

Phương pháp này cũng tương tự như phương pháp trên cần $\sqrt{N}$ bước, tuy nhiên không gian lưu trữ sẽ nhỏ hơn.

2.1.3 Phương pháp Pohlig-Hellman

Pohlig và Hellman đề xuất phương pháp này trong [24].

Nếu có thể phân tích bậc N của G thành các thừa số nguyên tố thì có thể viết:

$$N = \prod_i q_i^{e_i}$$

Ý tưởng của phương pháp này là tìm $k \pmod{q_i^{e_i}}$ với mỗi i, sau đó áp dụng định lý đồng dư Trung Hoa để tính $k \pmod N$. Coi q là số nguyên tố và q^e là lũy thừa e của q được chia hết bởi N, viết k dưới dạng sau:

$$k = k_0 + k_1 q + k_2 q^2 + \cdots, \quad 0 \le k_i < q$$

Lý giải thuật toán 2.2 như sau:

$$\begin{aligned} \frac{N}{q} Q &= \frac{N}{q} \left(k_0 + k_1 q + \cdots\right) P \\ &= k_0 \frac{N}{q} P + \left(k_1 + k_2 q + \cdots\right) NP = k_0 \frac{N}{q} P \end{aligned}$$

Bởi vì $NP = \infty$ và từ đây có thể tìm được k_0. Tiếp theo:

$$Q_1 = Q - k_0 P = \left(k_1 q + k_2 q^2 + \cdots\right) P$$

$$\frac{N}{q^2}Q_1 = \frac{N}{q}\left(k_1 + k_2 q + \cdots\right)P$$
$$= k_1\frac{N}{q}P + \left(k_2 + k_3 q + \cdots\right)NP = k_1\frac{N}{q}P$$

Từ đó tìm được k_1, tương tự như vậy chúng ta sẽ tìm được $k_2, k_3 \ldots$ Thuật toán sẽ dừng khi $e = r+1$, khi đó N/q^{e+1} không còn là số nguyên nữa và chúng ta không thể nhân Q_e với một số hữu tỷ.

Thuật toán 2.2 Phương pháp Pohlig-Hellman

1: Tính $T = \left\{ j\left(\frac{N}{q}P\right)\right\} \mid \quad 0 \le j \le q-1$.
2: Tính $\frac{N}{q}Q$. Đó là phần tử $k_0\left(\frac{N}{q}P\right)$ của T.
3: **if** $e = 1$ **then**
4: Nhảy đến bước 15.
5: **end if**
6: $Q_1 \leftarrow Q - k_0 P$
7: Tính $\frac{N}{q^2}Q_1$. Đó là phần tử $k_1\left(\frac{N}{q}P\right)$ của T.
8: **if** $e = 2$ **then**
9: Nhảy đến bước 15.
10: **end if**
11: Lần lượt tính được các giá trị $k_0, k_1, \ldots, k_{r-1}$ và $Q_0, Q_1, \ldots, Q_{r-1}$
12: $Q_r \leftarrow Q_{r-1} - k_{r-1}q^{r-1}P$
13: Xác định k_r sao cho $\frac{N}{q^{r+1}}Q_r = k_r\left(\frac{N}{q}P\right)$
14: **if** $e = r+1$ **then**
15: $k = k_0 + k_1 q + k_2 q^2 + \cdots + k_{e-1}q^{e-1}(\mod q^e)$
16: Stop.
17: **end if**
18: Quay về bước 11.

2.1.4 Phương pháp tấn công MOV

Tấn công MOV là tên viết tắt của các tác giả Menezes, Okamoto, và Vanstone [25], sử dụng cặp Weil để chuyển đổi bài toán Logarithm rời rạc trong $E(\mathbb{F}_q)$ thành bài toán Logarithm rời rạc trong $\mathbb{F}_{q^m}^{\times}$. Bởi vì giải bài toán Logarithm rời rạc trong trường hữu hạn sẽ dễ dàng và nhanh hơn giải Logarithm rời rạc trong nhóm các điểm trên đường cong Elliptic. Chọn m sao cho:

$$E[N] \subseteq \mathbb{F}_{q^m}^{\times}$$

Bởi vì tất cả các điểm trong $E[N]$ đều có tọa độ trong $\overline{\mathbb{F}}q = \cup_{j\geq 1}\mathbb{F}q^j$, nên m tồn tại. Theo định nghĩa về cặp Weil và các thuộc tính của cặp song tuyến tính:

$$\zeta_2 = e_N(Q, T_1) = e_N(kP, T_1) = e_N(P, T_1)^k = \zeta_1^k$$

Thuật toán 2.3 Tấn công MOV

1: Chọn điểm ngẫu nhiên $T \in E(\mathbb{F}_{q^m})$.
2: Tính bậc M của T.
3: Cho $d = \gcd(M, N)$ và cho $T_1 = (M/d)T$ có nghĩa là T_1 có bậc là d, chia hết bởi N, do đó $T_1 \in E[N]$.
4: Tính các cặp Weil $\zeta_1 = e_N(P, T_1)$ và $\zeta_2 = e_N(Q, T_1)$. Cả hai $\zeta_1, \zeta_2 \in \mu_d \subseteq \mathbb{F}_{q^m}^{\times}$.
5: Giải bài toán Logarithm rời rạc $\zeta_2 = \zeta_1^k$ trong $\mathbb{F}_{q^m}^{\times}$, sẽ tính được $k \pmod N$.
6: Lặp lại với điểm ngẫu nhiên T cho đến khi bội số chung nhỏ nhất của các số d là N, từ đó xác định được $k \pmod N$.

2.2 Tham số của hệ mật ECC

Các tham số của hệ mật ECC cần được lựa chọn kỹ càng để tránh các tấn công như MOV, trong quá trình lựa chọn hệ ECC cần phải đạt được một số tiêu chí được mô tả trong chuẩn [26].

Định nghĩa 2.2.1. *Tham số hệ mật $D = (q, FR, S, a, b, P, n, h)$ là một tập hợp gồm:*

1. *Bậc của trường $\mathbb{F}_q$ là q.*
2. *Phương pháp biểu diễn trường FR (field representation) được sử dụng cho các phần tử của $\mathbb{F}_q$.*
3. *S là mầm được sử dụng trong trường đường cong Elliptic được tạo ra một cách ngẫu nhiên.*
4. *Hai hệ số $a, b \in \mathbb{F}_q$ được dùng để định nghĩa đường cong E trên $\mathbb{F}_q$ (nghĩa là $y^2 = x^3 + ax + b$).*
5. *P là một điểm có bậc nguyên tố n và gọi là điểm cơ sở $P = (x_P, y_P) \in E(\mathbb{F}_q)$.*
6. *Đồng hệ số $h = \#E(\mathbb{F}_q)/n$.*

2.3 Trao đổi khóa

Trong các mục còn lại, chuyên đề sẽ đề cập đến một số thuật toán ứng dụng trong trao đổi khóa, mã hóa và ký số cơ bản. Chuẩn do công ty Certicom xây dựng

[26] mô tả chi tiết việc triển khai ứng dụng ECC. Tác giả D. Hankerson [18] phân tích việc triển khai ECC bằng phần mềm, trong khi đó tác giả L. Cao [27] phân tích thực hiện các giao thức cơ bản của ECC bằng phần cứng.

2.3.1 Trao đổi khóa Diffie–Hellman ECDH

Năm 1998, Laurie và cộng sự đề xuất giao thức trao đổi khóa dựa trên ECC [28]. Sau đó giao thức này đã được đưa vào các tiêu chuẩn ANSI X9.42, ANSI X9.63 và IEEE P1363.

Hai bên A và B cần tạo khóa phiên bí mật trao đổi trong một kênh truyền công khai, hai bên cùng thỏa thuận điểm cơ sở P trên E. Bên A tạo khóa bí mật d_A và gửi giá trị d_AP cho bên B, ngược lại bên B tạo khóa bí mật d_B nhân với P sau đó gửi lại cho A. Khi đó khóa phiên của bên A sẽ là $K_A = d_Ad_BP$, và của bên B sẽ là $K_B = d_Bd_AP$. Dễ dàng nhận thấy $K_A = K_B$, khóa này chỉ riêng hai bên A và B có thể tính được. Xem sơ đồ dưới đây:

Bên A		Bên B
	$\xrightarrow{d_AP}$	
d_A		d_AP
	$\xleftarrow{d_BP}$	
d_BP		d_B
$K_A = d_Ad_BP$		$K_B = d_Bd_AP$

Đánh giá bảo mật: Để tìm được khóa chia sẻ K_A hoặc K_B, Hacker buộc phải tìm được cả 2 khóa bí mật d_A, d_B, trong khi chỉ có thể bắt được thông tin trên đường truyền là d_AP và d_BP, khi biết P, Hacker buộc phải giải bài toán Logarithm rời rạc $d_A = \log_P(d_AP)$ và $d_B = \log_P(d_BP)$ và đây là bài toán khó không giải được trong thời gian đa thức.

2.3.2 Tạo khóa bí mật chia sẻ ECMQV

Tên đầy đủ của giao thức là Elliptic Curve Menezes-Qu-Vanstone. Thuật toán đã được đưa vào trong các chuẩn ANSI X9.63, IEEE 1363-2000, và ISO/IEC 15946-3. Theo các tiêu chuẩn này điểm cơ sở được ký hiệu là G thay vì là P như thường gặp. Lược đồ này thường được sử dụng khi các bên A và B có cặp khóa công khai và bí mật cố định, tương ứng là (a, aG) và (c, cG).

Bên A sinh cặp số ngẫu nhiên (b, bG) và bên B tương ứng sinh cặp số ngẫu (d, dG), và trao đổi 2 cặp nay cho nhau giá trị bG và dG. Kí hiệu hàm $x : E \to \mathbb{N}$, lấy giá trị x của một điểm trên đường cong E.

Thuật toán 2.4 Tạo khóa bí mật chia sẻ ECMQV

INPUT: Các tham số của hệ mật (K, E, q, h, G), các số a, b, aG, bG, cG và dG.
OUTPUT: Khóa bí mật chia sẻ Q (chia sẻ với với đối tượng có khóa công khai cG).

1: $n \leftarrow \lceil log_2(\#k) \rceil / 2$.
2: $u \leftarrow (x(bG) (\text{mod } 2^n) + 2^n$.
3: $s \leftarrow b + ua ((\text{mod } q)$.
4: $v \leftarrow (x(dG) (\text{mod } 2^n) + 2^n$.
5: $Q \leftarrow s(dG + v(cG))$.
6: **if** $Q = \infty$ **then**
7: Quay lại bước 1.
8: **end if**
9: Trả về khóa Q.

Bên B có thể tính ra cùng số Q bằng cách thay (a, b, c, d) trong thuật toán trên bằng (c, d, a, b). Bên A sẽ có các giá trị u_A, v_A, s_A và bên B sẽ có u_B, v_B, s_B. Dễ dàng nhận thấy [10]:

$$\begin{aligned}
u_A &= v_B \\
u_B &= v_A \\
Q_A &= s_A(dG + v_A(cG)) = s_A(d + v_A c)G \\
&= s_A(d + u_B c)G = s_A s_B G \\
Q_B &= s_B(bG + v_B(aG)) = s_B(b + v_B a)G \\
&= s_B(b + u_A a)G = s_B s_A G \\
Q_A &= Q_B = Q
\end{aligned}$$

Đánh giá bảo mật: Để hack được khóa chia sẻ, Hacker cần phải tính được các giá trị a, b, c, d, muốn vậy Hacker phải giải các bài toán Logarithm rời rạc $a = \log_G(aG)$, $b = \log_G(bG)$, $c = \log_G(cG)$, $d = \log_G(dG)$. Đây là các bài toán khó không thể giải được trong thời gian đa thức.

2.4 Xác thực - chữ ký số

2.4.1 ECDSA(The Elliptic Curve Digital Signature Algorithm)

Năm 1999, ECDSA (The Elliptic Curve Digital Signature Algorithm) đã được phê duyệt thành tiêu chuẩn của ANSI (ANSI X9.62-1998 ECDSA, phiên bản mới nhất là X9.62-2005), năm 2000 ECDSA cũng được IEEE và NIST phê duyệt thành

tiêu chuẩn FIPS PUB 186-4 (DSS - Digital Signature Standard), phiên bản mới nhất ban hành 7-2013. ISO năm 2002 cũng ban hành tiêu chuẩn ISO/IEC 15946-2:2002 trong đó có phần dành riêng về ECDSA. Mô tả chi tiết về ECDSA có thể tìm thấy trong [29].

Người ký sẽ chọn số d làm khóa bí mật và tạo ra khóa công khai là $Q = dP$, sử dụng hàm băm H để tạo ra giá trị tóm lược văn bản e của văn bản m. Chữ ký số sẽ là cặp (r, s) được tính theo thuật toán 2.5.

Thuật toán 2.5 Sinh chữ ký số ECDSA

INPUT: Tham số D = (q, FR, S, a, b, P, n, h), khóa bí mật d, thông điệp m.
OUTPUT: Chữ ký số (r, s).

1: Chọn ngẫu nhiên $k \in [1, n-1]$,
2: $R \leftarrow kP = (x_1, y_1)$ và chuyển đổi $\bar{x_1} \leftarrow x_1$.
3: $r \leftarrow \bar{x_1} \pmod n$.
4: **if** $r = 0$ **then**
5: Nhảy đến bước 1:
6: **end if**
7: $e \leftarrow H(m)$.
8: $s \leftarrow k^{-1}(e + dr) \pmod n$.
9: **if** $s = 0$ **then**
10: Nhảy đến bước 1:
11: **end if**
12: Trả về (r, s)

Người xác thực chữ ký nhận được văn bản m' và chữ ký số (r, s) của người ký, sẽ tính giá trị tóm lược e' của văn bản nhận được là m' và áp dụng thuật toán 2.6 để xác định sự phù hợp của chữ ký số với văn bản nhận được, từ đó có thể khẳng định văn vản có do đúng người ký ký hay có sự giả mạo từ người khác hoặc văn bản có bị sửa đổi hay bị lỗi do đường truyền hay không.

Thuật toán 2.6 Xác thực chữ ký số ECDSA

INPUT: Tham số D = (q, FR, S, a, b, P, n, h), khóa công khai $Q = dP$, thông điệp nhận được m', chữ ký (r, s).
OUTPUT: Chữ ký hợp lệ hoặc không hợp lệ.

1: Kiểm tra r và s có phải là những số nguyên nằm trong khoảng $[1, n-1]$. Nếu không trả về return("Chữ ký không hợp lệ").
2: $e' \leftarrow H(m')$.
3: $w \leftarrow s^{-1} \pmod n$.
4: $u_1 \leftarrow e'w \pmod n$ và $u_2 \leftarrow rw \pmod n$.
5: $R' \leftarrow u_1P + u_2Q$.
6: **if** $R' = \infty$ **then**
7: return("Chữ ký không hợp lệ").
8: **end if**
9: Chuyển đổi x_1 của $R' \rightarrow$ số nguyên $\bar{x_1}$.
10: $r' \leftarrow \bar{x_1} \pmod n$.
11: **if** $r' = r$ **then**
12: return("Chữ ký hợp lệ").
13: **else**
14: return("Chữ ký không hợp lệ").
15: **end if**

Chứng minh tính đúng đắn của thuật toán: cần phải chứng minh rằng nếu $m' = m$ hay $e = e'$ thì $r' = r$. Thực vậy:

$$\begin{aligned} w &= s^{-1} = k(e + dr)^{-}1 \\ R' &= u_1P + u_2Q = (u_1 + u_2d)P = (e' + rd)wP \\ &= (e' + rd)s^{-1}P = k(e' + rd)(e + rd)^{-1}P \end{aligned}$$

Nếu $e = e'$ ta sẽ có $R' = k(e+rd)(e+rd)^{-1}P = kP = R$ là điều cần phải chứng minh.

Đánh giá bảo mật: Để giả mạo được chữ ký, Hacker cần phải tìm được giá trị k và khóa bí mật d, để tìm được 2 giá trị này Hacker buộc phải giải 2 bài toán Logarithm rời rạc $k = \log_P R$ và $d = \log_P Q$ và đây đều là 2 bài toán khó, chưa giải được trong thời gian đa thức.

2.4.2 Chữ ký số ElGamal

Dựa trên lược đồ ký số do ElGamal đề xuất năm 1984 [30], phiên bản sửa đổi đã được đưa vào thành chuẩn về chữ ký số DSS (Digital Signature Standard) trong FIPS 186 [31].

Định nghĩa hàm f như sau:

$$f : E(\mathbb{F}_{||}) \rightarrow \mathbb{Z}$$

Có thể chọn hàm $f(x, y) = x$, trong đó x là số nguyên $0 \leq x < q$. Cặp khóa bí mật và công khai của người ký là $(x, Y) \mid Y = xP$. N là bậc của điểm P thường là số nguyên tố lớn.

Thuật toán 2.7 Sinh chữ ký số Elgamal

INPUT: Khóa bí mật x, thông điệp m.
OUTPUT: Chữ ký số (R, s).

1: Chọn ngẫu nhiên $k \in [1, n-1]$,
2: $R \leftarrow kP$.
3: $s = k^{-1}(m - xf(R))$.
4: Trả về (R, s)

Thuật toán 2.8 Xác thực chữ ký số Elgamal

INPUT: Khóa công khai $Y = xP$, thông điệp nhận được m', chữ ký (R, s).
OUTPUT: Chữ ký hợp lệ hoặc không hợp lệ.

1: Tính $V_1 = f(R)Y + sR$.
2: Tính $V_2 = m'P$.
3: **if** $V_1 = V_2$ **then**
4: return("Chữ ký hợp lệ").
5: **else**
6: return("Chữ ký không hợp lệ").
7: **end if**

Chứng minh tính đúng đắn của thuật toán: khi $m' = m$:

$$V_1 = f(R)Y + sR = f(R)xP + k^{-1}(m - xf(R))R = mP = m'P = V_2$$

Đánh giá bảo mật: Muốn giả mạo chữ ký số, Hacker buộc phải tính được s, để tính được s buộc phải tính được k và khóa bí mật x, để tính được 2 giá trị này Hacker buộc phải giải bài toán Logarithm rời rạc $k = \log_P R$ và $x = \log_P Y$, là 2 bài toán không giải được trong thời gian đa thức.

2.5 Mã hóa - Giải mã

2.5.1 Mã hóa Massey-Omura

Massey-Omura là hai tác giả để xuất lược đồ mã hóa được mô tả trong Patent [32] vào năm 1986. Lược đồ mã hóa này ít được sử dụng trong thực tế nhưng nó có ý nghĩa về mặt lịch sử.

Bên A		Bên B
Biểu diễn thông điệp như $M \in E(\mathbb{F}_{\shortparallel})$		
Chọn $m_A \mid \gcd(m_A, N) = 1$	$\xrightarrow{M_1 = m_A M}$	M_1
M_2	$\xleftarrow{M_2 = m_B M_1}$	Chọn $m_B \mid \gcd(m_B, N) = 1$
Tính $m_A^{-1} \in \mathbb{Z}_{\mathbb{N}}$	$\xrightarrow{M_3 = m_A^{-1} M_2}$	M_3
		$M = m_B^{-1} M_3$

Dễ dàng nhận thấy:

$$m_B^{-1} m_A^{-1} m_B m_A M = M$$

Đánh giá bảo mật: Muốn phá khóa trong lược đồ này, Hacker phải tìm được giá trị m_A, m_B để tìm được các giá trị này Hacker phải lần lượt giải 2 bài toán Logarithm rời rạc $m_A = \log_M M_1$ và $m_B = \log_{M_1} M_2$, và đây là 2 bài toán chưa giải được trong thời gian đa thức.

2.5.2 Mã hóa ElGamal

Trên cơ sở hệ mật ElGamal [30], lược đồ mã hóa được phát biểu như sau:

Bên A		Bên B
Thông điệp $M \in E(\mathbb{F}_{\shortparallel})$		Chọn cặp khóa $(x_B, Y_B) \mid Y_B = x_B P$
Chọn k, tính $M_1 = kP$		
Tính $M_2 = M + kY_B$	$\xrightarrow{M_1, M_2}$	M_1, M_2
		$M = M_2 - x_B M_1$

Chứng minh tính đúng đắn của lược đồ mã hóa:

$$M = M_2 - x_B M_1 = M + kY_B - x_B M_1 = M + k(x_B P) - x_B(kP) = M$$

Đánh giá bảo mật: Để giải mã được văn bản M, Hacker buộc phải tìm được k và x_B, do đó Hacker cần phải giải 2 bài toán Logarithm rời rạc $k = \log_P M_1$ và $x_B = \log_P Y_B$, và đây là 2 bài toán khó.

2.5.3 Mã hóa ECIES (The Elliptic Curve Integrated Encryption System)

ECIES do Bellare và Rogaway đề xuất và là một biến thể của mã hóa dùng hệ mật ElGamal, sau đó thuật toán này đã được đưa vào các chuẩn ANSI X9.63 và ISO/IEC 15946-3, IEEE P1363a và [26].

Tham số D = (q, FR, S, a, b, P, n, h) được chọn tương tự như với ECDSA. Ở đây cần lựa chọn thêm các hàm mã hóa/giải mã đối xứng ký hiệu là $E_k(m)$ và $D_k(c)$. Trong đó m là bản rõ cần mã hóa, c là bản đã được mã. Thuật toán mã hóa đối xứng được chọn ở đây để phục vụ quá trình mã hóa/giải mã được dễ dàng hơn và nhanh hơn so với các thuật toán bất đối xứng. Ngoài ra thay vì sử dụng hàm băm đơn giản, ECIES sẽ sử dụng hai hàm băm sau:

- Message authentication code $MAC_k(c)$:

$$MAC : \{0,1\}^n \times \{0,1\}^* \longrightarrow \{0,1\}^n$$

- Key derivation function $KD(T, l)$:

$$KD : E \times \mathbb{N} \longrightarrow \{0,1\}^*$$

  l là độ dài khóa ($k_1||k_2$). $\{0,1\}$ là chuỗi bit có giá trị $0, 1$ có độ dài n hoặc không xác định (*).

Người nhận có cặp khóa công khai/bí mật là (Y, x) trong đó $Y = xP$.

Thuật toán 2.9 Mã hóa ECIES

INPUT: Văn bản cần mã hóa m, khóa công khai Y.
OUTPUT: Văn bản đã được mã hóa (U, c, r).

1: Chọn $k \in [1, q-1]$.
2: $U \leftarrow kP$.
3: $T \leftarrow kY$.
4: $(k_1 \| k_2) \leftarrow KD(T, l)$.
5: Mã hóa văn bản, $c \leftarrow E_{k_1}(m)$.
6: Tính giá trị MAC cho văn bản mã hóa $r = MAC_{k_2}(C)$
7: Trả về return(U, c, r).

Bên giải mã sẽ nhận được tập hợp (U, c, r) gồm các thành phần sau:

- U cần thiết để tính khóa phiên Diffie–Hellman T.
- c là bản đã được mã hóa.
- r được dùng để xác thực mã văn bản..

Thuật toán 2.10 Giải mã ECIES

INPUT: Văn bản mã hóa U, c, r, khóa bí mật x.
OUTPUT: Văn bản đã giải mã m hoặc thông báo "văn bản mã không hợp lệ".

1: $T \leftarrow xU$.
2: $(k_1 \| k_2) \leftarrow KD(T, l)$.
3: Giải mã văn bản, $m \leftarrow D_{k_1}(c)$.
4: **if** $r \neq MAC_{k_2}(C)$ **then**
5: xuất thông báo "văn bản mã không hợp lệ"
6: **end if**
7: Trả về văn bản đã được giải mã m.

Khóa phiên T sau khi được tính trong phần giải mã sẽ có giá trị giống như trong phần mã hóa. Thực vậy:

$$T_{Decryption} = xU = x(kP) = k(xP) = kY = T_{Encryption}$$

Đánh giá bảo mật: Để phá khóa được lược đồ này Hacker cần phải tìm được khóa bí mật x hoặc giá trị k bằng cách giải bài toán $x = \log_P Y$ hoặc $k = \log_P U$, và đây là 2 bài toán khó chưa giải được trong thời gian đa thức.

Một số thuật toán và giao thức khác sử dụng đường cong Elliptic ứng dụng trong mật mã có thể xem thêm trong [10, 21]. Tài liệu "*Guide to Elliptic Curve Cryptography*" [19] với rất nhiều thuật toán chi tiết có thể coi là cẩm nang để triển khai cho những bài toán ứng dụng cụ thể của ECC.

Kết luận

Với 40 trang, chuyên đề này đã được tổng hợp từ gần 4000 trang tài liệu kinh điển về đường cong Elliptic một cách ngắn gọn súc tích nhưng vẫn tương đối đầy đủ. Người viết cũng chứng minh lại một số bổ đề, định lý theo cách đơn giản nhất, hạn chế sử dụng các khái niệm phức tạp, đồng thời cũng chỉ ra một số điểm thiếu chính xác và chưa rõ ràng trong một số tài liệu của nước ngoài [7].

Báo cáo đã đề cập tới các vấn đề cốt lõi của hệ mật mã dựa trên đường cong Elliptic. Chứng minh luật cộng và tính số điểm trên đường cong trong trường hữu hạn. Bên cạnh đó báo cáo chuyên đề cũng trình bầy một số tổ hợp giữa lược đồ ký số tập thể với các mô hình ký số khác như ký ủy nhiệm, ký mù, là 2 dạng lược đồ có nhiều ứng dụng thực tiễn trong công tác quản lý của tổ chức (ký ủy nhiệm) và trong công tác bầu cử điện tử và phát hành tiền điện tử (chữ ký số mù). Báo cáo chuyên đề này có thể được sử dụng làm tài liệu tham khảo cho những người nghiên cứu về đường cong Elliptic trong lĩnh vực toán học cũng như Công nghệ thông tin.

Trong thời gian tới, báo cáo sẽ được mở rộng về hệ mật ID-Based dựa trên các song tuyến tính Weil và Tale, bổ sung thêm cúc ứng dụng của đường cong Elliptic trong việc kiểm tra số nguyên tố và phân tích ra thừa số nguyên tố và đặc biệt sẽ mô tả chi tiết về việc cài đặt các hệ mật trong thực tiễn bằng phần mềm và phần cứng (FPGA) Khi có điều kiện báo cáo sẽ được xuất bản dưới dạng sách chuyên khảo.

Tài liệu tham khảo

[1] J. W. Bos, J. A. Halderman, N. Heninger, J. Moore, M. Naehrig, and E. Wustrow, "Elliptic Curve Cryptography in Practice," *Financial Cryptography and Data Security*, vol. 8437, pp. 157–175, 2014.

[2] J. H. Silverman and J. T. Tate, *Rational Points on Elliptic Curves - Second Edition.* Springer, 2015.

[3] L. C. Washington, *Elliptic Curves Number Theory and Cryptography, Second Edition.* CRC Press, 2008.

[4] J. W. S. Cassels, *Lectures on Elliptic Curves.* University of Cambridge, 1991.

[5] S. Lang, *Elliptic Curves Diophantine Analysis.* Springer, 1978.

[6] C. Kenig, A. Ranicki, and M. Rockner, *Elliptic Curves A Computational Approach.* Walter de Gruyter GmbH & Co., 2003.

[7] H. Cohen and G. Frey, *Handbook of Elliptic and Hyperelliptic Curve Cryptography.* Chapman Hall/CRC, 2006.

[8] J. H. Silverman, *The Arithmetic of Elliptic Curves.* Springer, 2009.

[9] L. Berger, G. Bockle, L. D. M. Dimitrov, T. Dokchitser, and J. Voight, *Elliptic curves, Hilbert modular forms and Galois deformations.* Birkhauser, 2013.

[10] I. F. Blake, G. Seroussi, and N. P. Smart, *Advances in Elliptic Curve Cryptography.* Cambridge University Press, 2005.

[11] I. Connell, *Elliptic Curve Handbook.* McGill University, 1999.

[12] T. H. Otway, *Elliptic Hyperbolic Partial Differential Equations.* Springer, 2015.

[13] Dang Minh Tuan, "Che tao thiet bi VPN IPSec bang phan cung dau tien o Vietnam," *Tap chi CNTT & TT*, no. 2, pp. 41–45, 2014.

[14] H. Lenstra., "Factoring Integers with Elliptic Curves," *The Annals of Matematics*, vol. 126, no. 3, pp. 649–673, 1987.

[15] V. S. Miller, "Use of elliptic curves in cryptography," *CRYPTO '85*, pp. 417–428, 1985.

[16] N. Koblitz, "Elliptic curve cryptosystem," *Math.Comp*, vol. 48, no. 16, pp. 203–209, 1987.

[17] A. Enge, *Elliptic Curves and Their Applications to Cryptography.* Kluwer Academic Publishers, 2001.

[18] D. Hankerson, J. L. Hernandez, and A. Menezes, "Software Implementation of Elliptic Curve Cryptography over Binary Fields," *CHES2000*, vol. 1965, pp. 243–267, 2000.

[19] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography.* Springer-Verlag, 2004.

[20] R. Schoof, "Elliptic Curves Over Finite Fields and the Computation of Square Roots," *Matematics of Computation*, pp. 483–495, 1985.

[21] I. Blake, G. Seroussi, and N. Smart, *Elliptic Curves in Cryptography.* Cambridge University Press, 1999.

[22] H. Cohen, *A Course in Computational Algebraic Number Theory.* Springer-Verlag, 1993.

[23] J. M. Pollard, "Monte Carlo Methods for Index Computations (mod p)," *Mathematics of Computation*, vol. 32, no. 143, pp. 918–924, 1978.

[24] S. C. Pohlig and M. E. Hellman, "An Improved Algorithm for Computing Logarithms over GF(p) and its Cryptographic Significance," *IEEE Transactions on Information Theory*, vol. 24, pp. 106–110, 1978.

[25] A. J. Menezes, T. Okamoto, and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *IEEE Trans. Inform. Theory*, vol. 39, no. 5, pp. 1639–1646, 1993.

[26] C. Research, *Standards For Efficient Cryptography, SEC 1: Elliptic Curve Cryptography.* Certicom Corp, 2000.

[27] L. Gao, S. Shrivastava, and G. E. Sobelman, "Elliptic Curve Scalar Multiplier Design Using FPGAs," *CHES'99*, vol. 1717, pp. 257–268, 1999.

[28] L. Laurie, M. Alfred, Q. Minghua, S. Jerry, and V. Scott, "An Efficient Protocol for Authenticated Key Agreement," *Designs Codes and Cryptography*, vol. 28, no. 2, 1998.

[29] D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," 2001.

[30] T. E. Gamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *CRYPTO '84*, vol. 196, pp. 10–18, 1985.

[31] NIST, *Digital Signature Standard (DSS) FIPS 186-4.* National Institute of Standards and Technology, 2013.

[32] J. Massey and J. Omura, "Method and apparatus for maintaining the privacy of digital messages conveyed by public transmission," Jan. 28 1986, US Patent 4,567,600. [Online]. Available: https://www.google.com/patents/US4567600