КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет комп'ютерних наук та кібернетики
Кафедра математичної інформатики

Кваліфікаційна робота
на здобуття ступеня бакалавра
за освітньо-професійною програмою "Інформатика"
спеціальності 122 Комп'ютерні науки на тему:

ІМПЛЕМЕНТАЦІЯ КРИПТОСИСТЕМ НА ОСНОВІ ЕЛІПТИЧНИХ КРИВИХ

Виконав студент 4-го курсу
Нікіта СОЛОНКО _____

(підпис)

Науковий керівник:
Член-кореспондент НАН України, професор
Анатолій АНІСІМОВ _____

(підпис)

Засвідчую, що в цій роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

(підпис)

Київ – 2023

РЕФЕРАТ

Обсяг роботи: 31 сторінок, 5 ілюстрацій, 0 таблиць, 28 використаних джерел.

TBD.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

ECC Elliptic curve cryptography;
EC Elliptic curve;
SIMD Single instruction multiple data;
DLP Discrete Logarithm Problem, задача дискретного логарифму;
GNFS General Number Field Sieve;
CDH Computational Diffie–Hellman assumption;
DDH Decisional Diffie–Hellman assumption;

ВСТУП

Оцінка сучасного стану об’єкта дослідження

На даний момент, окрім того, що застосування еліптичних кривих у криптографії активно досліджується в академічних роботах, існує безліч прикладних реалізацій ECC, що використовуються повсюдно. Більш того, криптосистеми з відкритим ключем, що використовують еліптичні криві, потроху стають популярнішими за, найбільш розповсюджений зараз, RSA. Так, наприклад, нові версії ssh рекомендують використовувати ключі Ed25519, що побудовані на еліптичних кривих Едварда, замість RSA. Аналогічні рекомендації зараз дають такі сервіси як: GitLab, GitHub, Amazon Web Services, Google Cloud Platform і багато інших.

Актуальність роботи та підстави для її виконання

Як було зазначено раніше, криптосистеми з відкритим ключем побудовані на основі еліптичних кривих вже активно використовуються, але все ще перебувають у стані активного розвитку. Так, наприклад, стаття [1] в якій була запропонована одна з найбільш популярних зараз еліптичних кривих Ed25519 і алгоритм підпису/верифікації на її основі EdDSA була написана в 2012 році, що для криптографії недавно. Також зараз існує багато програмних реалізацій таких криптосистем, але, у зв’язку зі складністю їх імплементації, не всі вони є цілком безпечними. Варто зазначити, що з кожним роком архітектура обчислювальних систем розвивається: оптимізуються існуючі інструкції та додаються нові, що відкриває можливості як для оптимізації існуючих реалізацій, так і для написання нових - кращих. Наприклад, в архітектурі x64 апаратна підтримка SIMD обчислень з 512 бітними регістрами AVX-512 з’явилася в 2013 році, вже після виходу статті з описом Ed25519.

Мета й завдання роботи

Метою роботи є дослідження застосування еліптичних кривих у сучасній криптографії, і як результат дослідження імплементувати криптографічну бібліотеку для роботи з еліптичними кривими та порівняти її з існуючими реалізаціями.

Можливі сфери застосування

Оскільки програмна реалізація яку я розроблю у рамках цієї роботи не буде сертифікована, то її не варто буде застосовувати в реальних системах на які може бути здійснена атака, але вона може бути застосована у навчальних і академічних цілях. В перспективі така бібліотека може бути застосована всюди де потрібна безпечна комунікація, в умовах коли можливе прослуховування, або втручання в канали зв’язку, між сутностями які ще не узгодили симетричний ключ. Наприклад: TLS, HTTPS і так далі.

РОЗДІЛ 1. ЕЛІПТИЧНІ КРИВІ

1.1. Переваги над $\mathbb{Z}_p^*$

Під $\mathbb{Z}_p^*$ мається увазі мультиплікативна група лишків за модулем p, де p - просте число.

Розглянемо мотивацію використання нового криптографічного примітиву, коли вже побудовано і імплементовано багато криптосистем на основі $\mathbb{Z}_p^*$, таких як: Diffie-Hellman, ElGamal, RSA і багато інших. Проблема використання таких криптосистем полягає у тому, що вони були запропоновані давно: Diffie-Hellman [2] - 1976, RSA [3] - 1978, ElGamal [4] - 1985. З того часу кратно збільшились обчислювальні можливості. Частота ядер виросла з кГц до ГГц, кількість ядер збільшилась з одиниць до сотень, з'явилися зручні інструменти для об'єднання процесорів у кластери. Також, через бажання зламати дані криптосистеми, багато вчених шукали спосіб знайти більш ефективні алгоритми розв'язання проблем на які вони спираються. Таким чином у 1993 Ленстра [5] придумав ефективний алгоритм розкладання великих чисел на множники - GNFS(General Number Field Sieve), який може розкласти число $n > 10^{100}$ за час:

$$\exp\left(\left(\left(\frac{64}{9}\right)^{\frac{1}{3}} + o(1)\right)(\log n)^{\frac{1}{3}}(\log\log n)^{\frac{2}{3}}\right)$$

Це стало основною проблемою криптосистем на основі $\mathbb{Z}_p^*$, тому що він робить вирішення обчислювальних проблем, на які спираються ці криптосистеми субекспоненційним, замість експоненційних. Варто зазначити, що цей алгоритм не є чисто теоретичною загрозою. За його допомогою у 2019 році [6] було знайдено дискретний логарифм 795 бітного числа. Таким чином, в патенті RSA [3] рекомендований розмір ключів був 200біт, в першій NIST сертифікації -

512 біт, зараз же мінімальний рекомендований розмір - 2048 біт. Важливим також є те, що GNFS неможливо узагальнити на будь-які скінчені циклічні групи, тому він не зачіпає групу точок еліптичної кривої над скінченним полям (позначення буде наведено далі), по цій причині найкращий відомий алгоритм для вирішення проблеми дискретного логарифма для групи точок еліптичних кривих (ECDLP) займає час $O(\sqrt{q})$, де q - розмір групи. Через це розміри ключів що мають однакову безпеку для Z_p^* і для групи точок кліптичної кривої сильно відрізняються. Порівняльна характеристика безпеки ключа в залежності від розміру ключа RSA/Diffie-Hellman і ECC:

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 1: NIST Recommended Key Sizes

Рисунок 1: [7]

Як можна побачити з наведеної таблиці використання еліптичних кривих дає вагому перевагу, оскільки зменшує розмір ключа, що в свою чергу зменшує час потрібний на проведення операцій з ключем (що дає виграш у швидкодії), кількість затраченої енергії (що є вагомою перевагою для IoT), навантаження на мережу під час передачі ключа і загалом трафік.

Ще однією перевагою криптографії еліптичних кривих є те, що можна використати багато напрацювань з криптографії з відкритим що використовує $\mathbb{Z}_p^*$. Оскільки ECC як складно обчислювальну проблему використовує

знаходження дискретного логарифма - DLP(у випадку еліптичних кривих вживають термін ECDLP), аналогічно до складно обчислювальної проблеми в Diffie-Hellman і ElGamal. До того еліптичні криві мають додаткову структуру, яку не має $\mathbb{Z}_p^*$, що дозволяє побудувати неможливі для $\mathbb{Z}_p^*$ криптосистеми. Такою додатковую структурою є можливість будувати pairing(спарювання точок еліптичних кривих), lattices(алгебраїчні решітки), isogenies(ізогенії). Алгебраїчні решітки і ізогенії еліптичних кривих зараз активно використовуються для побудови пост-квантових криптосистем з відкритим ключем.

1.2. Загальний огляд еліптичних кривих

1.2.1. $E / \mathbb{R}$

Існує декілька форм задання еліптичних кривих. Почнемо з найбільш класичної:

Еліптичною кривою E називається множина точок, що є розв'язком рівняння: $y^2 = x^3 + a * x + b$.

Наведене рівняння називають рівнянням Веєрштрасса і, відповідно, дану форму задання еліптичної кривої називають формою Веєрштраса. Якщо еліптична крива задана над полем $\mathbb{F}$, будемо позначати це як $E / \mathbb{F}$

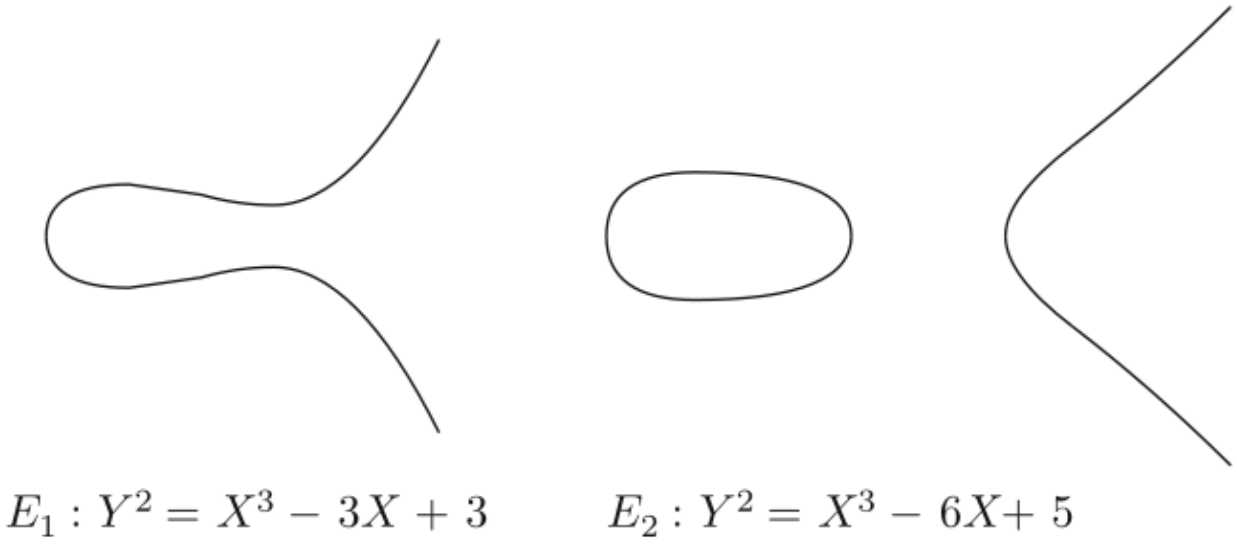

Рисунок 2: Ілюстрація 2 еліптичних кривих з [8]

Еліпітичні криві є не просто об'єктом вивчення аналітичної геометрії, а й цікавим об'єктом розгляду алгебри, бо над точками еліптичної кривої можна проводити обчислення. Перед переходом до еліптичних кривих над скінченими полями, для більш інтуітивного розуміння, розглянемо як «додавати» точки еліптичної кривої над $\mathbb{R}$, бо це можна легко інтерпретувати геометрично. Позначемо операцію додавання точок еліптичної кривої як $\oplus$ і розглянемо її на прикладі з [8].

Для початку визначимо, що якщо $A = (x, y) \in E$, то $-A = (x, -y) \in E$. Нехай E / $\mathbb{R}$ - це еліптична крива задана рівнянням $y^2 = x^3 - 15x + 18$; Точки $P = (7, 16), Q = (1, 2) \in E$, тоді пряма L що їх сполучає задається рівнянням: $y = \frac{7}{3}x - \frac{1}{3}$. Для того щоб знайти в яких точках L перетинає E ми можемо підставити замість y рівність з L в E і розв'язати рівняння відносно x. Для наведеного прикладу маємо:

$$\left(\tfrac{7}{3}x - \tfrac{1}{3}\right)^2 = x^3 - 15x + 18 \rightarrow x^3 - \tfrac{49}{9}x^2 - \tfrac{121}{9}x + \tfrac{161}{9} = 0$$

Отримане рівняння є рівнянням третьої степені, отже воно має 3 корені, 2 з яких нам вже відомі, залишається знайти третій корінь. Найлегшим способом

є поділити отриманий поліном на $(x-7)(x-1)$. Таким чином отримуємо третій корінь $x=-\frac{23}{9}$. Підставляючи x в E отримуємо $y=-\frac{170}{27}$, а отже точку $R=\left(-\frac{23}{9},-\frac{170}{27}\right)$. Далі відображаємо R відносно OX:

$$P \oplus Q = -R = R' = \left(-\tfrac{23}{9}, \tfrac{170}{27}\right)$$

Цей метод називається методом хорди. Наведене вище в графічно:

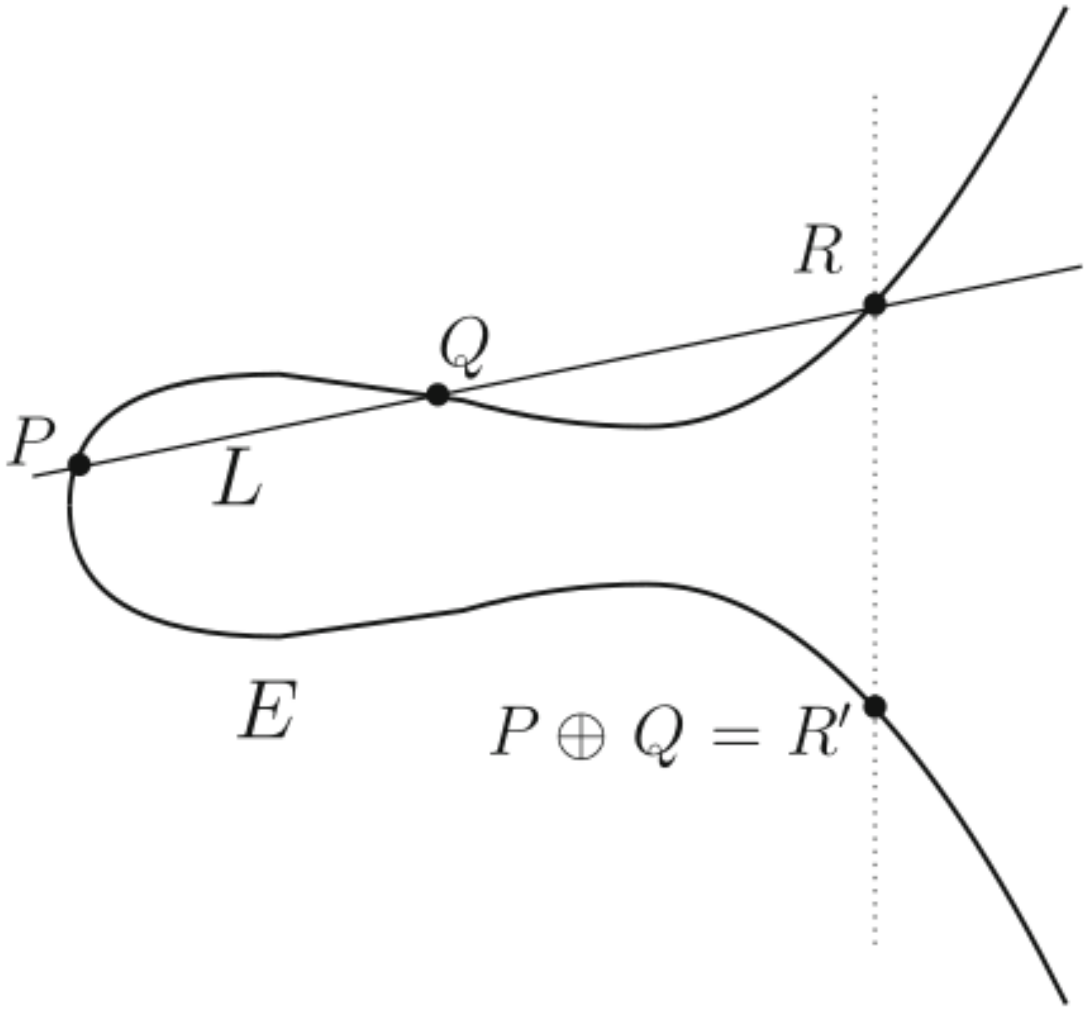

Рисунок 3: $P \oplus Q$ з [8]

Розглянемо випадок $P \oplus P = 2P = R'$. Для цього проведемо дотичну до E в точці P. Для цього потрібно діференціювати E по x. На прикладі, який розглядався раніше:

$$2y\frac{\mathrm{d}y}{\mathrm{d}x} = 3x^2 - 15 \rightarrow \frac{\mathrm{d}y}{\mathrm{d}x} = \frac{3x^2-15}{2y}$$

Підставляючи координати $P = (7, 16)$ отримуємо $L : y = \frac{33}{8}x - \frac{103}{8}$. Аналогічно попередньому прикладу знаходимо $R = \left(\frac{193}{64}, -\frac{223}{512}\right)$, а отже

$$P \oplus P = R' = -R = \left(\frac{193}{64}, \frac{223}{512}\right)$$

Цей метод називається методом дотичної. Наведене вище в графічно:

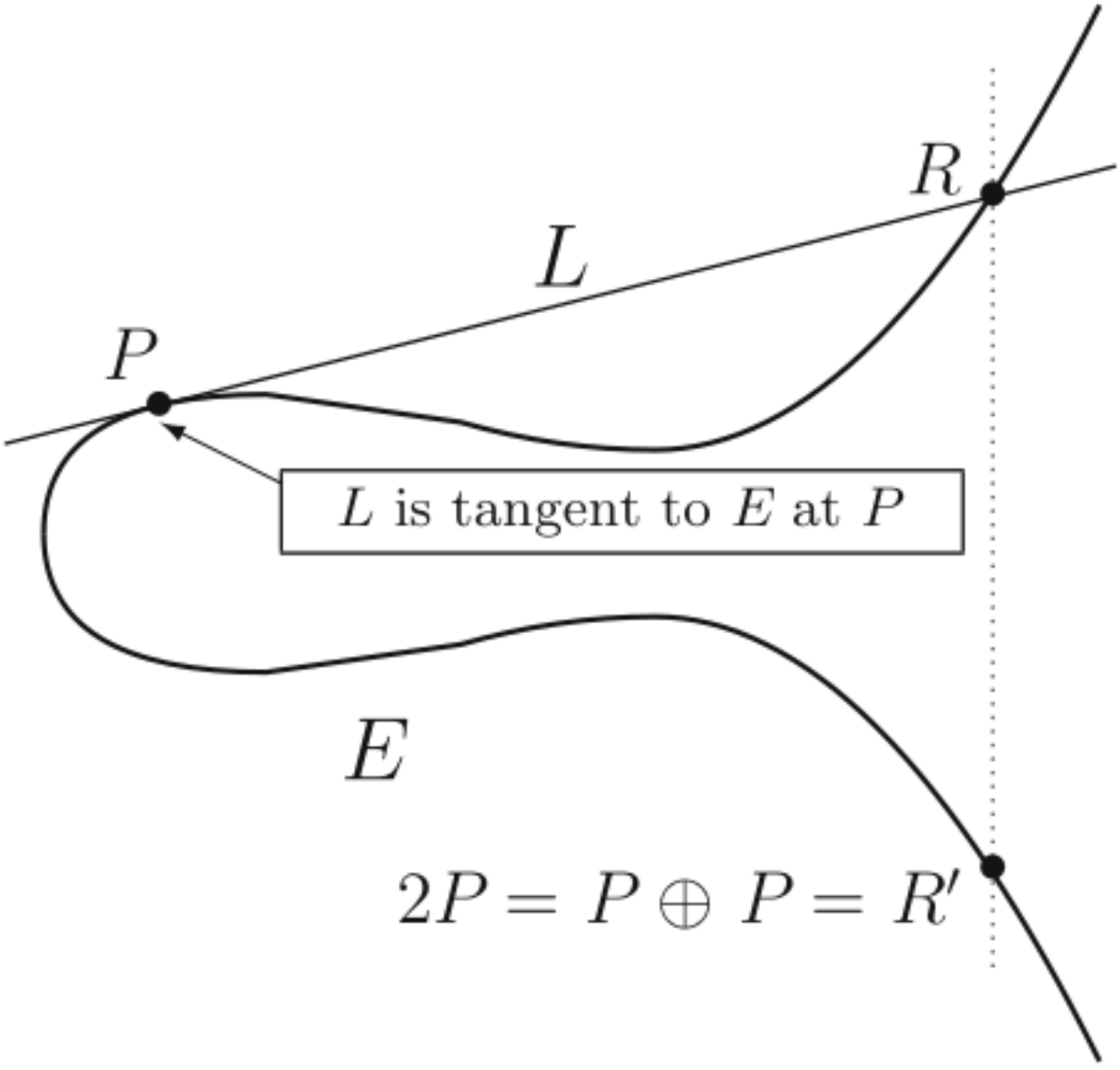

Рисунок 4: $P \oplus P$ з [8]

Залишиось розглянути останній випадок: $P \oplus (-P)$. Для цього введемо спеціальну точку, що називається точкою на нескінченості: $\mathcal{O}$, тоді $P \oplus (-P) = \mathcal{O}$. Графічно це виглядає наступним чином:

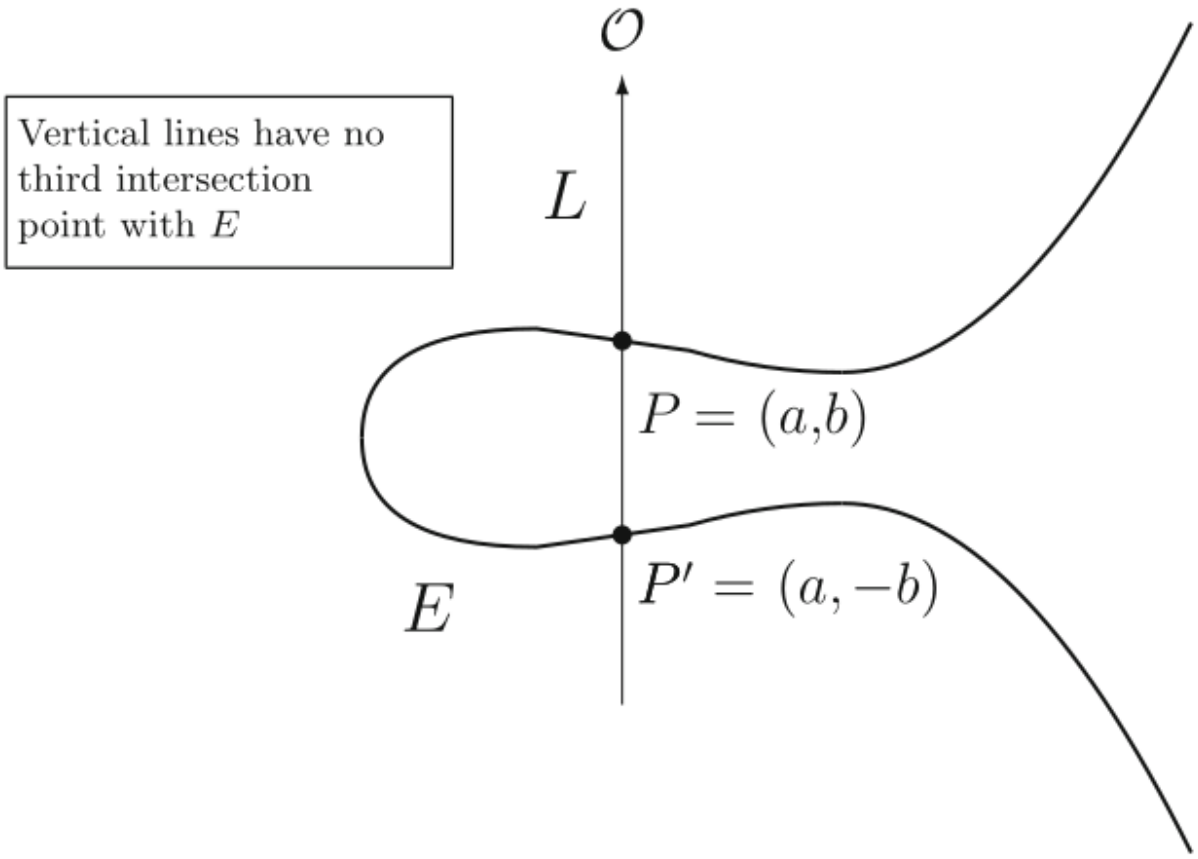

Рисунок 5: $P \oplus (-P) = \mathcal{O}$ з [8]

Формальний кінцевий алгоритм, що покриває всі вищенаведені випадки, буде наведено у наступному підрозділі.

1.2.2. E / $\mathbb{F}_p$

Перейдемо до еліптичних кривих що застосовуються у криптографії - еліптичних кривих над скінченимими полями. Почнемо з визначення:

Нехай $p > 3$ - просте число. Тоді еліптичною кривою E визначеною над $\mathbb{F}_p$ є множина розв'язків рівняння $y^2 = x^3 + a * x + b$, де $a, b \in \mathbb{F}_p$ задовільняють умову $4a^3 + 27b^2 \neq 0$. Ця умова потрібна щоб впевнитись, що рівння $x^3 + ax + b = 0$ має тільки один корінь. У випадку, якщо попередня властивість не виконується, рівняння кривої буде мати 2, або 3 однакових кореня. Нехай у

такому випадку коренем рівняння кривої буде x_0, тоді точка $(x_0, 0)$ називається сингулярною, і відповідна крива теж називається сингулярною.

Множиною точок еліптичною кривої $E \,/\, \mathbb{F}_p : y^2 = x^3 + ax + b; a, b \in \mathbb{F}_p$ є $E\big(F_p\big) = \big\{(x, y) : x, y \in \mathbb{F}_p \wedge y^2 = x^3 + ax + b\big\} \cup \{\mathcal{O}\}$.

Розглянемо приклад. Нехай $E : y^2 = x^3 + 1$ визначена на $\mathbb{F}_{11}$, тоді: $E(\mathbb{F}_{11}) = \{\mathcal{O}, (-1, 0), (0, \pm 1), (2, \pm 3), (5, \pm 4), (7, \pm 5), (9, \pm 2)\}$

Тепер розглянемо $\oplus$ для $E\big(\mathbb{F}_p\big)$. Для цього будемо спиратися на результати розгляду $(E(\mathbb{R}), \oplus)$, бо для $E\big(\mathbb{F}_p\big)$ алгоритм виглядає аналогічно до алгоритму для $E(\mathbb{R})$, але його не вдасться легко інтерпретувати геометрично. Розглядом геометричної інтерпретації обчислень над точками еліптичних кривих над скінченими полями займається алгебраїчна геометрія і це виходить за межі даної роботи. Нехай $P = \big(x_p, y_p\big) \vee \mathcal{O}, Q = \big(x_q, y_q\big) \vee \mathcal{O} \in E\big(\mathbb{F}_p\big)$:

- Якщо $P = \mathcal{O} \to P \oplus Q = Q$
- Інакше, якщо $Q = \mathcal{O} \to P \oplus Q = P$
- Інакше, якщо $Q = -P \to P \oplus Q = \mathcal{O}$ (варто зазначити що це покриває випадок коли $y_p = y_q = 0$)
- Інакше, якщо $P = Q$ використовуємо метод дотичної: $\lambda = \frac{3x_p + a}{2y_p}$
- Інакше використовуємо метод хорди: $\lambda = \frac{y_q - y_p}{x_q - x_p}$
- $x = \lambda^2 - x_p - x_q; y = \lambda\big(x_p - x\big) - y_p \to P \oplus Q = (x, y)$

$\big(E\big(\mathbb{F}_p\big), \oplus\big)$ утворюють циклічну абелеву групу.

Наведений вище алгоритм для еліптичних кривих загального вигляду має 2 основних недоліки:

1. Він не є достатньо швидким, як буде розглянуто далі існують спеціальні види еліптичних кривих, для яких його можна прискорити
2. Імплементація додавання точок еліптичної кривої має працювати за константний час, інакше можлива атака що викриває приватний ключ. Як можна побачити,

даний алгоритм важко реалізувати таким чином, щоб він працював за константиний час через різну кількість обчисленнь за різних умов.

Властивість, коли алгоритм обчислення $\oplus$ має єдину формулу для усіх точок в $E(\mathbb{F}_p)$ при будь яких умовах, тобто може бути імплементований без умовних операторів, називається complete addition law(повний закон додавання).

1.3. Додаткові обчислення в $E(\mathbb{F}_p)$

1.3.1. $|E(\mathbb{F}_p)|$

Однією з найважливіших характеристик еліпитичної над скінченим полем є $q = |E(\mathbb{F}_p)|$ - розмір групи. Стосовно цієї характеристики існує теорема Hesse(Гассе):

Нехай E / $\mathbb{F}_{p^e}$, тоді: $\left|E\left(\mathbb{F}_{p_e}\right)\right| = p^e + 1 - t_{p^e}; t_{p^e} \leq 2\sqrt{p^e}$ - називається слідом Фробеніуса.

Важливим є здатність точно обчислити $\left|E\left(\mathbb{F}_{p_e}\right)\right|$. Вперше ефективний алгоритм запропнував Schoof в [9], потім Elkies і Atkin запропонували поліпшення складності роботи цього алгоритму [10]. Результат називається SEA, він залишається найшвидшим алгоритмом до цього часу і має складність $O(\log^2 p * M(\log^2 p) \,/\, \log\log p)$, де $M(n)$ - складність множення.

1.3.2. Обчислення nP

Більшість ECC криптосистем використовують обчислення $nP, P \in E(\mathbb{F}_p)$. Для того щоб ці системи було можливо використовувати у реальному світі це обчислення має бути швидким. Спочатку визначимо формально, що

$$P \in E(\mathbb{F}_p), n \geq 1 \in \mathbb{N} : nP = (n-1)P \oplus P$$

Очевидний приклад: $3P = P + 2P = P + P + P$. Також зрозуміло що обчислювати nP наївно, за $n - 1$ додавань - неоптимально, тому використовують оптимізації:

Double-and-Add(подвоюй-і-додавай). З її допомогою можна обчислити nP за не більше ніж $2\log_2 n$ додавань. Існують 2 версії цього алгоритму: яка використовує інверсію($P \to -P$) і яка не використовує. Варто відзначити, що знаходження точки еліптичної кривої оберненою до даної - тривільне і швидке, і перша версія для випадковго n потребує меншу кількість операцій у середньому. Приклад реалізації другого варіанту з використанням псевдокоду:

```
Double-and-Add(P, n):
  Q := P
  R := 𝒪
  while n > 0:
    if n ≡ 1 (mod 2):
      R := R + Q
    Q := 2 * Q
    n := n / 2
  return R
```

На практиці, якщо потрібно обчислити, наприклад, $947P$ першим методом буде обчислено:

$$947P = P \oplus 2P \oplus (-2^4P) \oplus (-2^6P) \oplus 2^{10}P$$

А другим:

$$947P = P \oplus 2P \oplus 2^4P \oplus 2^5P \oplus 2^7P \oplus 2^8P \oplus 2^9P$$

Недоліком цього алгоритму є те, що час його роботи залежить від значення n. Це робить його вразливим до timing attack.

Montgomery ladder(Сходи Монтгомері). Цей алгоритм має аналогічну часову складіність до алгоритму Double-and-Add, але його перевагою є те, що час його роботи залежить від кількості бітів в n, а не від самого значення, що робить його невразливим до timing-attack при умові якщо $\oplus$ також не вразлива до неї. Сам алгоритм [11], нехай $n = (n_{t-1}, ..., n_0)_2$, кількість бітів в $n = t$ позначимо $n[i] = n_i$:

```
MontgomeryLadder(P, n):
  R := 𝒪
  Q := P
  for i = t - 1 downto 0:
    if n[i] = 0:
      Q := Q + R
      R := R + R
    else:
      R := R + Q
      Q := Q + Q
  return R
```

Нехай $x(P), P \in E(\mathbb{F}_p)$ - x координата точки P. Також, модифікація наведеної імплементації цього алгоритму дозволяє обчислити $x(nP)$ знаючи тільки $x(P)$, використовуючи наступні формули:

$$\text{Якщо } (2\alpha)P \neq \mathcal{O} : x_{2\alpha} = \frac{1}{4}\Big(\left(x_\alpha^2 - a\right)^2 - 8bx_\alpha\Big) \,/\, \left(x_\alpha^3 + ax_\alpha + b\right)$$

$$\text{Якщо } (2\alpha+1)P \neq \mathcal{O} : x_{2\alpha+1} = \left(\left(a - x_\alpha x_{\alpha+1}\right) - 4b(x_\alpha + x_{\alpha+1})\right) \,/\, \left(x_1(x_\alpha - x_{\alpha+1}^2)\right)$$

Як висновок, можна резюмувати, що швидкість обчислення nP є важливою характеристикою еліптичної кривої. Існують криві, для яких можна оптимізувати як одиничну операцію $P + Q$, так і обчислення nP в цілому.

1.3.3. Дискретний логарифм в $E(\mathbb{F}_p)$

Нехай E - еліптична крива, $\mathbb{F}_p$ - скінчене поле, $P \in E(\mathbb{F}_p), Q \in \langle P \rangle$, тоді дискретним логарифмом Q за основою P є таке n, що $Q = nP$ і позначається $n = \log_P Q$.

Нехай q - порядок P, тоді очевидно що, якщо $Q = n_0 P$, то $Q = (n_0 + iq)P, \forall i \in \mathbb{N}$. Для визначеності будемо вважати, що дискретний логарифм - це найменше таке n, що $Q = nP$. У такому випадку зрозуміло, що $\log_P(Q) \in \mathbb{Z}_q$.

Важливо відмітити, що для дискретного логарифму точок еліптичної кривої виконується властивість звичайного логарифму:

$$\log_P(Q_1 + Q_2) = \log_P(Q_1) + \log_P(Q_2), \forall Q_1, Q_2 \in \langle P \rangle$$

З цього випливає, що $\log_P$ визначає груповий гомоморфізм:

$\log_P : \langle P \rangle \to \mathbb{Z}_q$

Зазвичай на практиці P вибирається з генераторів $E(\mathbb{F}_p)$.

ECDLP - проблема знаходження дискретного логарифму для точок еліптичної кривої.

1.4. Pairing(Спарювання точок еліптичної кривої)

$E / \mathbb{F}_p$ має додаткову структуру - pairing. Вона може бути використана для побудови криптосистем неможливих для циклічних груп у загальному випадку, а також для $\mathbb{Z}_p^*$ і дозволяє розширити множину криптосистем ECC. Зазвичай у визначенні використовують мультиплікативні групи, замість адитивних, але оскільки ця робота сконцентрована саме на еліптичних кривих, то:

Нехай $\mathbb{G}_0, \mathbb{G}_1, \mathbb{G}_T$ - циклічні групи порядку q, де q - просте.

$g_0 \in \mathbb{G}_0, g_1 \in \mathbb{G}_1$ - генератори. Pairing - це ефективно обчислювальна функція $e : \mathbb{G}_0 \times \mathbb{G}_1 \to \mathbb{G}_T$, яка задовільняє наступні вимоги:

1. Білінеарність. $\forall u, u' \in \mathbb{G}_0, \forall v, v' \in \mathbb{G}_1$:

$e(u + u', v) = e(u, v)e(u', v) \land e(u, v + v') = e(u, v)e(u, v')$

2. Недегенаривність: $g_T = e(g_0, g_1)$ - генератор в $\mathbb{G}_T$

Коли $\mathbb{G}_0 = \mathbb{G}_1$ назвемо це спарювання симетричним, інакше - асиметричним. $\mathbb{G}_0, \mathbb{G}_1$ називаються pairing groups, а $\mathbb{G}_T$ - target group(цільовою групою).

З білінеарності випливає основна властивість спарювання, яка використовується в побудові криптосистем:

$\forall \alpha, \beta \in \mathbb{Z}_q : e\left(g_0^\alpha, g_1^\beta\right) = e(g_0, g_1)^{\alpha\beta} = e\left(g_0^\beta, g_1^\alpha\right)$

Вимога недегенеративності потрібна щоб e не повертала завжди $1 \in \mathbb{G}_T$ для всіх аргументів.

Найбільш зручним і ефективним спарюванням для $E / \mathbb{F}_p$ - є асиметричне спарювання. Спарювання сконструйоване з $E / \mathbb{F}_p$ має групи $\mathbb{G}_0, \mathbb{G}_1, \mathbb{G}_T$ з наступними властивостями:

- $\mathbb{G}_0$ - це підгрупа $E(\mathbb{F}_p)$ порядку q, де q - просте.
- $\mathbb{G}_1$ - це підгрупа $E(\mathbb{F}_{p^d})$ порядку q, для деякого $d \in \mathbb{N} > 0$, де $\mathbb{G}_0 \cap \mathbb{G}_1 = \{\mathcal{O}\}$
- $\mathbb{G}_T$ - це мультиплікативна група

Число $d \in \mathbb{N}$ називається embedding degree(ступіню вкладенння) кривої. Зрозуміло, для того що спарювання було ефективним d має убти малим, тому що, якщо d буде великим ми не зможемо записати елемменти в $\mathbb{G}_1$ та $\mathbb{G}_T$. Еліптитчні криві, де спарювання мале, наприклад $d \leq 16$, називають pairing friendly elliptic curves(еліптичними кривими дружніми для спарювання).

Оригінально почали використовувати Weil pairing(Спарювання Вейля). Нехай $m \in \mathbb{N}, P \in E : mP = \mathcal{O}$. Точка P є точкою скінченого порядку на еліптичній кривій, або torsion points(торсіонною точкою). Більш точно кажуть, що P є точкою порядку m. Позначимо множину точок порядку m як:

$$E(\mathbb{F}_p)[m] = \{P \in E(\mathbb{F}_p) : mP = \mathcal{O}\}$$

Тоді, нехай f_P, f_Q - раціональні функції над E(не наводжу тут визначення, ознайомитися з цією концепцією можна в [8], [12]), такі що: $\text{div}(f_P) = m[P] - m[\mathcal{O}], \text{div}(f_Q) = m[Q] - m[\mathcal{O}]$, тоді Wei pairing це:

$$e_m(P, Q) = \frac{f_P(Q+S)}{f_P(S)} \,/\, \frac{f_P(P-S)}{f_P(-S)},, \text{ де } S \in E, S \notin \{\mathcal{O}, P, -Q, P-Q\}$$

Для побудови Weil pairing існує ефективний Miller's algorithm [13] (Алогоритм Міллера). Але на практиці зараз використовують Tate pairing та Ate pairing, оскільки алгоритм Міллера для них ефективніший.

1.5. Спеціальні види еліптичних кривих

1.5.1. Montgomery Curve(Крива Монтгомері)

Еліптична крива $E / \mathbb{F}_p$ у формі Монгомері у зміних u, v має вигляд:

$$Bv^2 = u^3 + Au^2 + u; A, B \in \mathbb{F}_p, B(A^2 - 4) \neq 0$$

Це рівняння кривої може бути легко приведено до рівняння Веєрштрасса заміною зміних: $u = Bx - \frac{A}{3}; n = By$. Цікавою особливістю кривої Монгомері є те, що $\left|E\left(\mathbb{F}_p\right)\right|$ завжди ділиться на 4. З цього також випливає, що не кожну криву у формі Веєрштраса можна привести до кривої у формі Монтгомері заміною зміних.

Криві Монгомері дозволяють пришвидшити операцію додавання, шляхом оптимізації Montgomery ladder. Нехай $X_1 = x(P), Z_1 = 1$, і:

$$X_{2\alpha} = \left(X_\alpha^2 - Z_\alpha^2\right)^2$$

$$Z_{2\alpha} = 4X_\alpha Z_\alpha (X_\alpha^2 + AX_\alpha Z_\alpha + Z_\alpha^2)$$

$$X_{2\alpha+1} = 4Z_1 \left(X_\alpha X_{\alpha+1} - Z_\alpha Z_{\alpha+1}\right)^2$$

$$Z_{2\alpha+1} = 4X_1 \left(X_\alpha Z_{\alpha+1} - Z_\alpha X_{\alpha+1}\right)^2$$

Тоді $x(\alpha P) = X_\alpha \,/\, Z_\alpha$, якщо $\alpha P \neq \mathcal{O}$

1.5.2. Edwards Curve(Крива Едвардса)

Еліптична крива $E \,/\, \mathbb{F}_p$ формі Едвардса має вигляд:

$$x^2 + y^2 = 1 + dx^2y^2, d \neq 0, 1 \in \mathbb{F}_p$$

Аналогічно кривій Монтгомері, крива Едвардса може бути приведена до форми Вейерштраса заміною зміних, але не навпаки. $\left|E\left(\mathbb{F}_p\right)\right|$ завжди ділиться на 4. Перевагою кривих Едвардса є те, що $\oplus$ дуже просто імплементувати:

$$\forall P, Q \in E\left(\mathbb{F}_p\right) : P \oplus Q = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - x_1x_2}{1 - dx_1x_2y_1y_2}\right)$$

Для цього визначемо $\mathcal{O} = (0, 1)$. Як можна побачити криві Едвардса мають complete addition law.

1.5.3. Koblitz Curve(Крива Кобліца)

Оригінально крива була запропонована Кобліцом над $\mathbb{F}_2$ і мала вигляд:

$$E : y^2 + xy = x^3 + ax^2 + 1, a \in \{0, 1\}$$

Далі, почалося дослідження цієї кривої над $\mathbb{F}_{2^m}$, вона має вигляд:

$$y^2 + xy = x^3 + ax^2 + b, b \neq 0$$

Але, оскільки на практиці частіше застосовують криві цього виду над $\mathbb{F}_p$, і дана робота зосересереджена на $E \,/\, \mathbb{F}_p$, сконцентруємося саме на кривих Кобліца над $\mathbb{F}_p$. Крива Кобліца E над $\mathbb{F}_p$ задається рівнянням [14] :

$E : y^2 = x^3 + b, p \equiv 1 (\mathrm{mod}\, 3)$

В більш широкому розуміні кривими Кобліца називають еліптичні криві для яких існує ефективно обчислювальний нетривіальний ендоморфізм $\phi : \forall P, Q : \phi(P + Q) = \phi(P) + \phi(Q)$, який дозволяє оптимізувати операцію множення точки на число. Для $E(\mathbb{F}_{2^m})$ використовують ендоморфізм Фробеніуса: $\phi : \mathbb{F}_{p^k} \to \mathbb{F}_{p^k}, \alpha \to \alpha^p$. Застосовуючи його для $E \,/\, \mathbb{F}_{2^m}$ отримуємо $P = (x, y) \in E(\mathbb{F}_{2^m}), \phi(x, y) = (x^2, y^2)$. Для $E \,/\, \mathbb{F}_p$ використовують енфоморфізм $\phi(x, y) = (\omega x, y), 1 \neq \omega \in \mathbb{F}_p, \omega^3 = 1$. Зрозуміло що таке ω знайдеться, оскільки $p \equiv 1 (\mathrm{mod}\, 3)$, що означає, що 1 має 3 кубічних кореня: $1, u, u^2 \big(u \in \mathbb{F}_p\big)$.

Легко перевірити, що якщо $(x, y) \in E\big(\mathbb{F}_p\big)$, то і $\phi(x, y) = (\omega x, y) \in E(\mathbb{F} - p)$, оскільки $y^2 = x^3 + b = (\omega x)^3 + b = \omega^3 x^3 + b = x^3 + b$. Також додатково визначемо, що $\phi(\mathcal{O}) = \mathcal{O}$. Тобто, доведно, що $\phi : E\big(\mathbb{F}_p\big) \to E\big(\mathbb{F}_p\big)$.

Нехай $q = \big|E\big(\mathbb{F}_p\big)\big|$. Оскільки, ϕ є ендоморфізмом і $E\big(\mathbb{F}_p\big)$ - циклічна група, має існувати таке $\lambda \in \mathbb{Z}_q$, що: $\forall P \in E\big(\mathbb{F}_p\big) : \phi(P) = \lambda P$. Можливо визначити λ наступним чином. Очевидно що $P, \phi(P), \phi^2(P)$, мають однакову y координату, а отже знаходяться на одній прямій, до того ж горизонтальній. З геометричної інтерпретації $\oplus$ відомо, що це означає, що їх сума $\mathcal{O}$, отже:

$$\forall P \in E\big(\mathbb{F}_p\big) : \mathcal{O} = P \oplus \phi(P) \oplus \phi^2(P) = P \oplus \lambda P \oplus \lambda^2 P = \big(1 + \lambda + \lambda^2\big) P$$

Звідки можемо зробити висновок, що

$$1 + \lambda + \lambda^2 = 0 \to (\lambda - 1)\big(1^2 + 1 * \lambda + \lambda^2\big) = 0 * (1 - \lambda) \to \lambda^3 - 1 = 0 \to \lambda^3 = 1$$

Тобто λ аналогічно до ω це один з двох нетривіальних кубічних коренів 1.

Тепер розглянемо як ϕ може допомогти в прискорені обчислення nP. Нехай $n \in \mathbb{Z}_q$ і ми хочемо обчислити nP для деякого $P \in E(\mathbb{F}_p)$. Для більшості таких n можливо знайти такі $\tau_0, \tau_1, \tau_2 \in \mathbb{Z}$, що:

$$n = \tau_0 + \tau_1\lambda + \tau_2\lambda^2 \wedge |\tau_i| \leq 2q^{\frac{1}{3}} \text{ для } i = 0, 1, 2$$

Тобто, $nP = \tau_0 + \tau_1\phi(P) + \tau_2\phi^2(P)$

Така оптимізація дозволяє прискорити операцію множення у 2-3 рази [6] в порівнянні з Double-and-Add.

1.6. Припущення на яких будується ECC

1.6.1. ECDLP

Основне припущення ECC полягає у тому, що ECDLP для точок еліптичної кривої над скінченим полем є складнообчислювальною проблемою, тобто не існує ефективного алгоритму обчислення дискретного логарифму.

На ECDLP існує багато атак - ефективних алгоритмів розв'язання. Ним присвячена ціла глава в [12]. Нехай $P, Q \in E(\mathbb{F}_p), |E(\mathbb{F}_p)| = q$. Перша група атак працює для будь якої циклічної групи:

- Baby Step, Giant Step. Має часову складність $O(\sqrt{q})$ і потребує $O(\sqrt{q})$ місця. Цей алготирм є детерміністичним
- Алгоритми ρ і λ Полларда. ρ є узагальненням λ алгоритму. Вони також мають часову складність $O(\sqrt{q})$, але є потребують тільки $O(1)$ додаткового місця.
- The Pohlig-Hellman method [15](Метод Поліга-Геллмана). Ефективний якщо q не є простим і в його факторизації нема великих прочтих чисел. Нехай $q = \prod_i p_i^{e_i}$, тоді складність часова складність цього алгоритму $O\left(\sum_i e_i(\log q + \sqrt{p_i})\right)$

Такж існує група атак, що використовують pairing, вони в деяких є більш ефективними ніж атаки на циклічні групи. Такими атаками є:

- MOV(Menezes, Okamoto and Vanstone) [16]. Ця атака використовує Weil pairing щоб звести ECDLP в $E(\mathbb{F}_p)$ до DLP в $\mathbb{F}_{p^m}$

- The Frey-Rück Attack [17] (Атака Фрея-Рюка). Аналогічно MOV, тільки використовується більш ефективне Tate pairing.

Оскільки ECC грунтується на припущені про обчислювальну складність ECDLP і ECC активно використовується в реальних криптосистемах можна припустити, що для будь якої кривої $E \,/\, \mathbb{F}_p$, де p - достатньо велике просте число ECLP складнообчислювальне, але це є хибним твердженням, у тому числі через наведені до цього атаки, наприклад:

- Якщо $\left|E\left(\mathbb{F}_p\right)\right|$ не є простим і всі прості множники у його розкладі менші деякого $q_{\max}$, то тоді ми можемо застувати атаку з викристанням метода Поліга-Геллмана. Зокрема, якщо $q_{\max} < 2^{80}$, то ECDLP не є складнообчислювальною проблемою на практиці. Тому в реальних системах використовують криві $E \,/\, \mathbb{F}_p$ для яких $\left|E\left(\mathbb{F}_p\right)\right|$ рівне $q, 4q,$ або $8q$ для деякого простого q.
- Якщо $\left|E\left(\mathbb{F}_p\right)\right| = p$, то ECDLP для $E\left(\mathbb{F}_p\right)$ розв'язний за поліноміальний час. Такі криві називаються аномальнимим і не використовуються на практиці
- Нехай існує таке мале $\tau > 0 \in \mathbb{N}$, що $\left|E\left(\mathbb{F}_p\right)\right|$ ділить $p^\tau - 1$. Тоді ECDLP в $E\left(\mathbb{F}_p\right)$ можна звести до DLP в $\mathbb{F}_{p^\tau}$ використовуючи MOV, або атаку Фрея-Рюка, що в свою чергу дозволить використати GNFS. На практиці, якщо p - 256 бітне число і $\tau = 2$, то розв'язок ECDLP займе приблизно пару годин. Для захисту від цієї атаки, треба переконатися що p^τ достатньо велике для того щоб GNFS в $\mathbb{F}_{p^\tau}$ був нездійсненний.

Але, якщо параметри кривої підібрано правильно, то найоптимальніший алгоритм вирішення ECDLP має часову складність $O\left(\sqrt{q}\right)$, де $q = \left|E\left(\mathbb{F}_p\right)\right|$.

1.6.2. CDH

Нехай $\mathbb{G} = \langle g \rangle$ - циклічна група порядку q, $a, b \in \mathbb{Z}_q$. Припущення полягає у тому, що якщо зловмисник перехопив g, g^a, g^b він не зможе обчислити g^{ab}. Зараз єдиним відомим спрособом роз'язати CDH є розв'язати DLP, але не

доведено що DLP і CDH - це рівнозначні проблеми. CDH є сильнішим припущенням ніж DLP. Також, цікавою оболивістю CDH є те, що на відміну від RSA, не існує ефективного способу перевірити по заданому x, чи дійсно він є правильною відповіддю на CDH.

1.6.3. DDH

Нехай $\mathbb{G} - \langle g \rangle$ - циклічна група порядку q, $a, b, \in \mathbb{Z}_q$ обраін випадково. Припущеня полягає у тому, що зловмисник не може знаючи g, q відрізнити $\left(g^a, g^b, g^c\right)$ від $\left(g^a, g^b, g^{ab}\right)$. Це припущення є сильнішим ніж CDH.

1.7. Twist curves(скручені криві)

Кожна еліптична крива $E \,/\, \mathbb{F}_p$ має пов'язану з нею криву $\tilde{E} \,/\, \mathbb{F}_p$, яка називається twist of E(скрученою кривою E). Нехай $c \in \mathbb{F}_p$ не є квадратичним лишком в $\mathbb{F}_p$. Якщо E задається рівнянням $y^2 = x^3 + ax + b$, тоді $\tilde{E}$ задається рівнянням $cy^2 = x^3 + ax + b$. Нескладно показати, що $\left|E\left(\mathbb{F}_p\right)\right| + \left|\tilde{E}\left(\mathbb{F}_p\right)\right| = 2p + 2$, з цього, з використанням теореми Гассе, випливає що $\left|\tilde{E}\left(\mathbb{F}_p\right)\right| = p + 1 - t$.

Визначемо, що крива $E \,/\, \mathbb{F}_p$ twist secure(безпечно скручена), якщо ECDLP складно розв'язний як в $E \,/\, \mathbb{F}_p$, так і в $\tilde{E} \,/\, \mathbb{F}_p$.

Розглянемо для чого потрібна twist security. Нехай Боб має секретний ключ $\alpha \in \mathbb{Z}_q$, на надіслане $P \in E\left(\mathbb{F}_p\right)$ Боб відправляє αP. Для того щоб система була безпечною Боб має перевіряти чи $P \in E\left(\mathbb{F}_p\right)$, але нехай замість того щоб відсилати $P = (x, y)$, відсилається тільки x координата, тоді також можна як і отримати y, так і перевірити чи $\in E\left(\mathbb{F}_p\right)$. Тоді, перевірка $P \in E\left(\mathbb{F}_p\right)$ зводиться до перевірки чи є $y^2 = x^3 + ax + b$ квадратичним лишком в $\mathbb{F}_p$, що є відносно дорогою по часу операцією. Нехай, у такому випадку при імплементації крок перевірки був пропущений. Тоді, зловмисник може надіслати Бобу $x_1 = x\left(\tilde{P}\right) \in \mathbb{F}_q, \tilde{P} \in \tilde{E} \,/\, \mathbb{F}_q$. Якщо ECDLP легкорозв'язний в $\tilde{E} \,/\, \mathbb{F}_q$, тобто $E \,/\, \mathbb{F}_p$ не twist secure, то відповідь Боба розкриє його секретний ключ. У випадку, якщо

$E / \mathbb{F}_p$ twist secure то Бобу не загрожує така атака і він може використовувати оптимізацію при якій відправляється тільки x координата і не витрачати час на її перевірку.

1.8. Застосування еліптичних кривих поза межами ECC

Еліптичні криві можна застосувати у криптографії поза межами ECC:

1. Для факторизації, наприклад алгоритм факторизації Ленстра [18]. Він має часову складність $O\Big(\exp\Big(\big(\sqrt{2}+o(1)\big)\sqrt{\log p \log\log p}\Big)(\log N^2)\Big)$ [19]
2. Для тестування на простоту. Існують, наприклад, тести на простоту Atkin–Morain elliptic curve primality test зі складністю $O(\log^{6+\varepsilon} N)$, для деякого $\varepsilon > 0$ і Goldwasser–Kilian algorithm зі складністю $O(\log^{10+c_2} N)$, де c_2 деяке число більше 0.

Загалом, 20 найбільших чисел простоту яких було доведено, було доведено за допомгою еілптичних кривих [20].

1.9. Висновки

У цьому розліді було розглянуто і дослідження теоретичний фундамент ECC, а саме: розглянуто еліптитчні криві над $\mathbb{R}$, еліптичні криві над полями скінченої характеристики p, обчислення над точками еліптичних кривих і їх геометрична інтерпретація для еліптичних кривих над $\mathbb{R}$, обчислення кількості точок еліптичної кривої, досліджено способи обчислення і оптимізації множенння точки еліптичної кривої на ціле число, розглянуто спарювання точок еліптичної кривої, спеціальні випадки еліптичних кривих (Монгомері, Едвардса, Кобліца), досліджено проблему знаходження дискретного логарифму для точок еліптичної кривої і можливі атаки на нього та припущення на яких будується криптографія еліптичних кривих, розглянуто скручування еліптичних кривих і застосування еліптичних кривих поза межами ECC. Як висновок, з цього розділу має бути очевидно, що розробка своєї еліптичної кривої - це складна справа з великою кількістю нюансів, до того ж, неправильно побулована еліптична крива може призвести до того, що криптосистеми з її

використанням не будуть безпечними. Тому на практиці використовують набір затверджених, або просто перевірених кривих. Деякі з них будуть розглянуті у наступному.

РОЗДІЛ 2. ЗАСТОСУВАННЯ У КРИПТОГРАФІЇ

2.1. Практична перевага над $\mathbb{Z}_p^*$

2.2. Кодування інформації за допомогою еліптичних кривих

2.3. Приклади еліптичних кривих у сучасній криптографії

2.3.1. secp256r1 (P256)

2.3.2. secp256k1 (Bitcoin curve)

2.3.3. Curve25519

2.3.4. bn256

2.4. Імплементація Curve25519

2.5. Протоколи узгодження ключів

2.5.1. ECDH

2.5.2. X25519

2.6. Протоколи підпису і верифікації

2.6.1. ECDSA

2.6.2. EdDSA

2.6.3. BLS

2.6.4. Ed25519

2.7. Порівняння з існуючими імплементаціями

ВИСНОВКИ

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

[1] Daniel J. Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang, “High-speed high-security signatures,” 2011. [Online]. Available: https://ed25519.cr.yp.to/ed25519-20110926.pdf

[2] W. Diffie, and M. E Hellman, “New directions in cryptography,” 1976. [Online]. Available: https://ee.stanford.edu/~hellman/publications/24.pdf

[3] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” 1985. [Online]. Available: https://people.csail.mit.edu/rivest/Rsapaper.pdf

[4] TAHER ELGAMAL, “Public key cryptosystem and a signature scheme based on discrete logarithms.” [Online]. Available: https://caislab.kaist.ac.kr/lecture/2010/spring/cs548/basic/B02.pdf

[5] Arjen K. Lenstra, and H. W. Lenstra, “The development of the number field sieve.”

[6] Dan Boneh, and Victor Shoup, *A Graduate Course in Applied Cryptography*, 2023. [Online]. Available: http://toc.cryptobook.us/book.pdf

[7] “What is ECC and why would I want to use it?.” [Online]. Available: https://www.globalsign.com/en/blog/elliptic-curve-cryptography

[8] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman, *An Introduction to Mathimatical Cryptography*, 2014.

[9] R. Schoof., “Elliptic curves over finite fields and the computation of square roots mod p,” 1985.

[10] R. Schoof., “Counting points on elliptic curves over finite fields,” 1995.

[11] Daniel J. Bernstein, and Tanja Lange, “Montgomery curves and the montgomery ladder.” [Online]. Available: https://eprint.iacr.org/2017/293.pdf

[12] Lawrence C. Washington, *Elliptic Curves: Number Theory and Cryptography*, 2008.

[13] V.S. Miller, “The weil pairing, and its efficient calculation,” 2004.

[14] Han Wu, and Guangwu Xu, “A note on koblitz curves over prime fields,” 2020. [Online]. Available: https://eprint.iacr.org/2020/1136.pdf

[15] S. Pohlig, and M. Hellman, “An improved algorithm for computing logarithms over Gf(p) and its cryptographic significance,” 1978. [Online]. Available: https://ee.stanford.edu/~hellman/publications/28.pdf

[16] A. J. Menezes, T. Okamoto, and S. A. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field,” 1993.

[17] G. Frey, M. Müller, and H.-G. Rück, “The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems,” 1999.

[18] H. W. Lenstra, “Factoring integers with elliptic curves,” 1987.

[19] Leo Lai, “Lenstra’s elliptic curve factorization method,” 2016. [Online]. Available: https://web.ma.utexas.edu/users/sl55444/CompsciTalk.pdf

[20] [Online]. Available: https://t5k.org/top20/page.php?id=27