УДК 512.72
DOI https://doi.org/10.24144/2616-7700.2024.45(2).65-74

Л. П. Добуляк[1], С. П. Солтис[2], О. Ю. Лисецька[3]

[1] Львівський національний університет ім. Івана Франка,
доцент кафедри математичного моделювання соціально-економічних процесів,
кандидат економічних наук, доцент
lesia.dobuliak@lnu.edu.ua
ORCID: https://orcid.org/0000-0001-8665-8783

[2] Львівський національний університет ім. Івана Франка,
магістр кафедри математичного моделювання соціально-економічних процесів
serhii.soltys@lnu.edu.ua
ORCID: https://orcid.org/0009-0005-6729-2938

[3] Львівський національний університет ім. Івана Франка,
доцент кафедри математичного моделювання соціально-економічних процесів,
доктор філософії з математики
oleksandra.lysetska@lnu.edu.ua
ORCID: https://orcid.org/0009-0000-0730-7222

КРИПТОГРАФІЯ ЕЛІПТИЧНИХ КРИВИХ

У статті детально досліджуються математичні аспекти еліптичних кривих, зокрема операції додавання та подвоєння точок, які є основними операціями для побудови криптографічних алгоритмів. Зосереджено увагу на властивостях еліптичних кривих над скінченними полями, що робить їх придатними для реалізації стійких криптографічних методів, таких як ECDSA (Elliptic Curve Digital Signature Algorithm). Розглянуто математичні основи цих операцій, їх алгоритмічні реалізації та важливість для обчислювальної стійкості. Окрім математичних основ, у статті розглянуто програмну реалізацію операцій додавання та подвоєння точок, що є важливими для ефективності та безпеки криптографічних алгоритмів. Описано алгоритми реалізації мовою програмування Python.

Ключові слова: еліптичні криві, криптографія, ECC, ECSDA, Python.

1. Вступ. Еліптичні криві та алгоритм цифрового підпису на основі еліптичних кривих (Elliptic Curve Digital Signature Algorithm, ECDSA) є ключовими компонентами сучасної криптографії. Вони використовуються для забезпечення безпеки даних у широкому спектрі застосувань, зокрема в блокчейн-технологіях, криптовалютах (наприклад, Bitcoin та Ethereum) та різних системах захисту інформації. Використання еліптичних кривих для криптографічних завдань забезпечує високу стійкість до зламів за допомогою сучасних обчислювальних технологій, водночас маючи порівняно низькі вимоги до ресурсів для обробки та зберігання даних.

Особливістю еліптичних кривих є їх здатність надавати криптографічну стійкість, яка може бути досягнута з меншою кількістю бітів порівняно з класичними методами, такими як RSA. У випадку застосування алгоритмів на еліптичних кривих вважається, що не існує субекспоненціальних алгоритмів щодо розв'язку задачі "дискретного логарифмування в групах їх точок". Це дозволяє створювати більш ефективні системи захисту з меншими обчислювальними витратами. Завдяки цьому алгоритми, засновані на еліптичних кривих, активно впроваджуються в таких технологіях, як інтернет речей (IoT), мобільні комунікації, банківські транзакції та блокчейн-платформи.

Ця стаття зосереджується на теоретичних основах еліптичних кривих та практичному застосуванні ECDSA для захисту даних.

2. Основний результат.

Постановка завдання. Метою даної статті є дослідження математичних основ еліптичних кривих і їхнього використання в алгоритмах цифрового підпису, таких як ECDSA. Основними завданнями дослідження є:

1) Опис математичних властивостей еліптичних кривих і їх застосування в криптографії.
2) Аналіз алгоритму ECDSA і його важливості для криптографічних завдань.
3) Огляд існуючих підходів і літературних джерел щодо використання еліптичних кривих в різних криптографічних системах.
4) Проведення експериментів і аналіз отриманих результатів щодо додавання точок на еліптичній кривій.

Огляд літератури. У 1970 році британський математик та інженер Джеймс Елліс запропонував ідею, засновану на простій концепції. Що, якщо шифрування і дешифрування — зворотні операції на основі двох різних ключів? У традиційній, тобто симетричній криптографії, повідомлення повинно бути надіслано разом з ключем, щоб інша сторона розшифрувала повідомлення. Елліс припустив, що одержувач повідомлення не може бути пасивною стороною, і їм потрібно було мати «замок» і «ключ» для себе. Замок можна було відправити кому завгодно в світі, але ключ повинен залишатися приватним. Криптографія з відкритим ключем була винайдена в 1970-х роках і є математичною основою для комп'ютерної та інформаційної безпеки. З моменту винаходу криптографії з відкритим ключем, було відкрито декілька математичних функцій, таких як піднесення до степеня простого числа і множення еліптичних кривих. Ці математичні функції практично необоротні, це означає, що результат їх виконання легко отримати в одному напрямку і неможливо в зворотному. На підставі цих математичних функцій, криптографія дозволяє створення цифрових шифрів і непідробних цифрових підписів. У біткоїнах використовується множення еліптичних кривих.

Еліптичні криві як основа для криптографії почали активно досліджуватися з кінця XX століття, зокрема завдяки працям таких математиків, як Ніл Кобліц (Neal Koblitz) і Віктор Міллер (Victor Miller), які у 1985 році запропонували використання еліптичних кривих для криптографічних цілей [1, 2]. Їхня робота стимулювала розробку численних досліджень щодо використання еліптичних кривих в криптографічних протоколах.

У [1, 2] описано основні математичні принципи еліптичних кривих, включаючи операції додавання точок та скалярного множення, що використовуються в криптографії. Ці математичні операції лежать в основі алгоритму ECDSA, який забезпечує високий рівень захисту завдяки своїй складності.

У сучасній криптографії важливим є також огляд ефективності ECDSA у порівнянні з іншими криптографічними алгоритмами, такими як RSA або DSA. Згідно з дослідженнями [3], ECDSA пропонує більш високу стійкість при меншій довжині ключа, що робить його особливо привабливим для систем з обмеженими ресурсами, таких як мобільні пристрої або платформи інтернету речей.

Робота [4] аналізує безпеку та стійкість до атак, особливо до таких загроз, як атаки на відкриті ключі та підбор ключів. Автори також зазначають, що алгоритм ECDSA залишається одним із найбільш стійких серед відомих криптографічних методів.

Основні матеріали. Еліптичні криві (Elliptic curves) є важливим об'єктом в алгебрі та математиці, а також в криптографії.

Означення 1. *Еліптична крива над полем K — це множина точок проективної площини над K, що задовольняють рівнянню*

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{1}$$

разом з точкою на нескінченності та не містить особливих точок, де a_i — дійсні константи, які визначають конкретну криву (див. [5]).

Одна з ключових властивостей еліптичних кривих полягає в тому, що вони утворюють абелеву групу з операцією додавання точок. Додавання точок на еліптичних кривих визначається геометрично і має свої особливості. Воно дозволяє обчислювати суму точок і знаходити добуток точки на ціле число. Еліптичні криві також мають нейтральний елемент, який називається «безкінечністю» і позначається символом O.

У криптографії еліптичні криві знаходять широке застосування, зокрема в алгоритмах цифрового підпису. ECC (Elliptic Curve Cryptography) заснована на складності обчислення дискретного логарифма на еліптичних кривих, що робить її ефективною та безпечною для використання в криптографічних протоколах, таких як шифрування, цифровий підпис та обмін ключами.

Основною перевагою криптосистем на еліптичних кривих у порівнянні із звичайними асиметричними алгоритмами є те, шо вони забезпечують еквівалентний захист за меншої довжини ключа (див. табл. 1).

Таблиця 1.

Ступінь захисту RSA та ECC

Ступінь захисту (на кожен біт ключа)	**Мінімальна довжина ключа (в бітах)**	
	RSA/DSA/DH	**ECC**
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

Розглянемо рівняння еліптичної кривої у спрощеному вигляді (рівняння Вейєрштрасса):

$$y^2 = x^3 + ax + b. \tag{2}$$

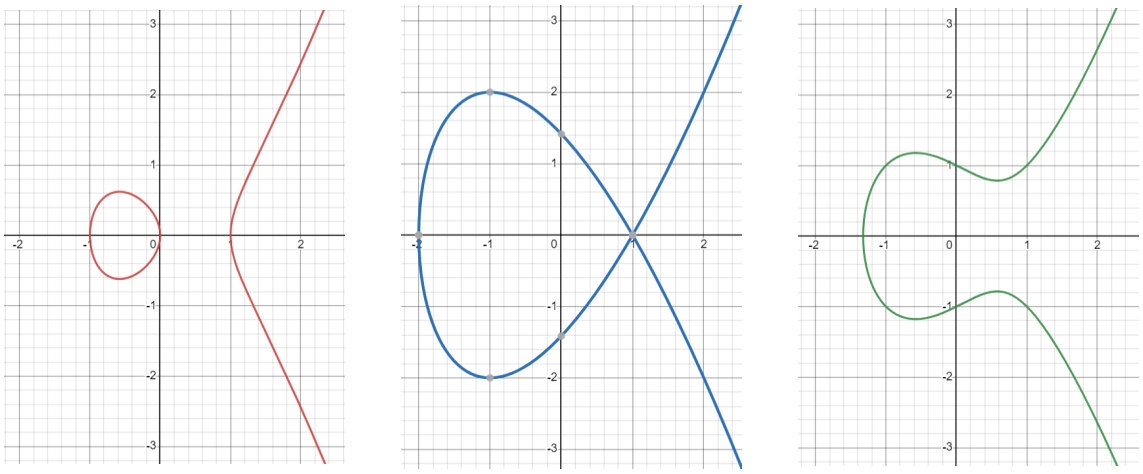
Залежно від значень параметрів a і b еліптичні криві можуть приймати на площині різні форми. Так як $y = \pm\sqrt{x^3 + ax + b}$, то графік кривої симетричний відносно Ox.

Дискримінант рівняння обчислюється так:

$$D = \left(\frac{a}{3}\right)^3 + \left(\frac{b}{2}\right)^2.$$

Розглянемо такі можливі випадки:

- $D < 0$ — три різних дійсних корені (Рис. 4, графік 1);
- $D = 0$ — три дійсних корені, два з яких однакові (Рис. 4, графік 2 — сингулярна крива, такі криві виключають з розгляду);
- $D > 0$ — один дійсний корінь та два комплексних (Рис. 4, графік 3, [6]).



$y^2 = x^3 - x$ $y^2 = x^3 - 3x + 2$ $y^2 = x^3 - x + 1$

Рис. 1. Графіки еліптичних кривих у залежності від значення D.

У реальних криптосистемах використовуються еліптичні криві над скінченним полем P, що описуються рівнянням:

$$y^2 \equiv x^3 + ax + b \ (mod\ p), \tag{3}$$

де $(x,\ y)$ — точки еліптичної кривої, a, b — параметри кривої, p — просте число $(p \neq 2,\ p \neq 3)$. При цьому параметри кривої a та b мають задовольняти умову:

$$4a^3 + 27b^2 \neq 0 \ (mod\ p).$$

Позначимо через $E_p(a,\ b)$ множину точок еліптичної кривої. Зауважимо, що множину точок еліптичної кривої $E_p(a,\ b)$ також включається нескінченно віддалена точка O. Точка належить еліптичній кривій, якщо пара чисел $(x,\ y)$ задовільняє рівняння (3).

Означення 2. *Кількість точок кривої називається порядком кривої ([6]).*

Додавання двох точок кривої.

Означення 3. *Оберненою точкою до точки $P(x,\ y) \in E_p(a,\ b)$ називають точку еліптичної кривої, що симетрична до $P(x,\ y)$ відносно осі Ox та позначають $-P(x,\ -y)$. Варто зауважити, що $-P$ має належати $E_p(a,\ b)$.*

Нехай $P,\ Q \in E_p(a,\ b)$, тоді щоб отримати нову точку R, яка є сумою точок P і Q, застосовуються такі кроки:

1) Якщо P і Q є різними точками, проводемо пряму, яка проходить через ці точки. Ця пряма перетне криву у третій точці R. Проведемо через точку R вертикальну пряму до перетину з кривою у точці $-R = P + Q$. Отже, сумою двох точок P та Q буде точка, обернена до третьої точки перетину еліптичної кривої і прямої, що проходить через задані точки.

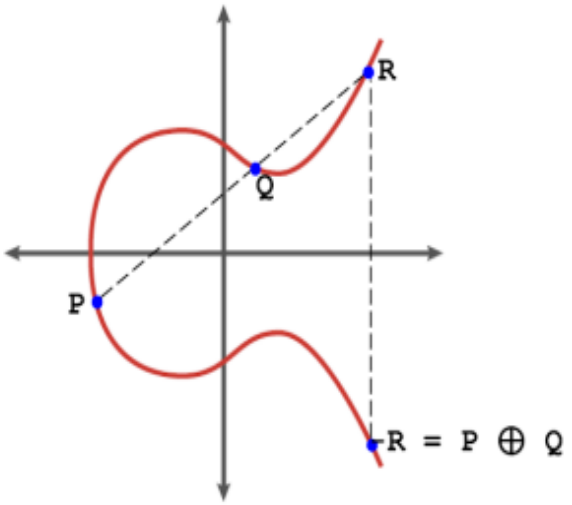


Рис. 2. Додавання точок.

2) У випадку $P = Q$, проведемо пряму, яка дотикається до кривої в точці P. Точка R — це точка перетину цієї дотичної з кривою. При $P = Q$ січна перетворюється на дотичну, тому точка $2P$ є оберненою до точки R.

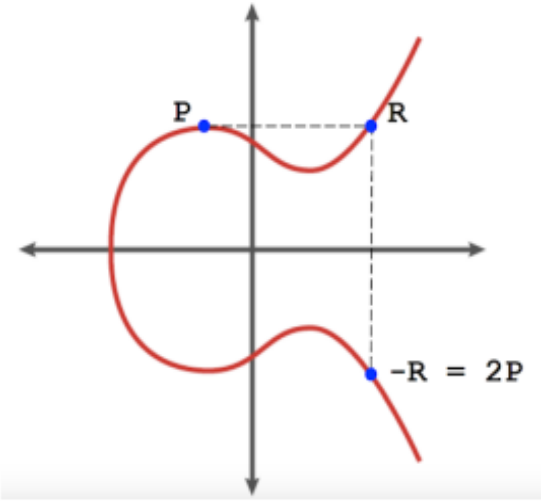


Рис. 3. Подвоєння точки.

Координати $-R(x_3, y_3)$ визначаються за формулами (див. рис. 7), де λ — кутовий коефіцієнт січної, що проведена через точки $P(x_1, y_1)$ та $Q(x_2, y_2)$.

Додавання точок (якщо P ≠ Q)	***Подвоєння точки (якщо P = Q)***
$x_3 = \lambda^2 - x_1 - x_2 (mod\ p);$	$x_3 = \lambda^2 - 2x_1\ (mod\ p);$
$y_3 = \lambda(x_1 - x_3) - y_1 (mod\ p);$	$y_3 = \lambda(x_1 - x_3) - y_1 (mod\ p);$
$\lambda = \frac{y_2 - y_1}{x_2 - x_1} (mod\ p).$	$\lambda = \frac{3x_1^2 + a}{2y_1} (mod\ p).$

Рис. 4. Формули обчислення координат точки.

3) Якщо P і Q мають координати, що не належать кривій, або $P = O$ (нескінченність), тоді R вважається рівним Q (або P, якщо $Q = O$, відповідно).

Отримана точка R є результатом операції додавання точок P і Q на еліптичній кривій. Важливо враховувати, що координати точок можуть бути елементами поля (раціональні числа або скінченні поля) і операція додавання узгоджується із операціями на полі.

Операція додавання точок на еліптичних кривих комутативна (тобто $P + Q = Q + P$) та асоціативна (тобто $(P + Q) + R = P + (Q + R)$).

Приклад 1. *Множина точок* $E_5(2, 1)$ *еліптичної кривої*

$$y^2 \equiv x^3 + 2x + 1(mod\ 5),$$

складається з 6 *точок. Порядок кривої* — 7.

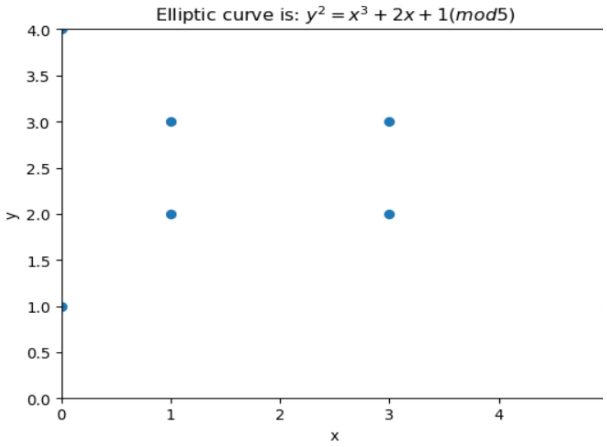


Рис. 5. Множина точок $E_5(2, 1)$.

Як можна побачити, на рисунку 8 зображено усі точки, що задовольняють умови кривої, це точки $(0, 1), (0, 4), (1, 2), (1, 3), (3, 2), (3, 3)$. *Перевірити цей факт можна простим підставленням:*

$$1^2 \equiv 0^3 + 2 * 0 + 1;$$
$$3^2 \equiv 3^3 + 2 * 3 + 1 \Rightarrow 9 \equiv 27 + 6 + 1(mod\ 5) \Rightarrow = 4 \equiv 4.$$

Розглянемо приклад з додаванням детальніше. Спочатку проведемо аналітичний пошук розв'язків, взявши невеликі значення, а далі обрахуємо це програмно і перевіримо чи відповіді співпадають.

Аналітичне розв'язання. Розглянемо таке рівняння еліптичної кривої:

$$y^2 \equiv x^3 + x + 1(mod\ 23), \tag{4}$$

і перевіримо чи точки $P(3, 10)$ та $Q(9, 7)$ належать кривій і, якщо належать, знайдемо їхню суму.

Підставимо значення у рівняння еліптичної кривої і переконаємося, що точки належать кривій 4.

$$10^2 \equiv 3^3 + 3 + 1(mod\ 23) \Leftrightarrow 100 mod\ 23 \equiv 31(mod\ 23) \Leftrightarrow 8 \equiv 8(mod\ 23);$$

$$7^2 \equiv 9^3 + 9 + 1(mod\ 23) \Leftrightarrow 49 mod\ 23 \equiv 739(mod\ 23) \Leftrightarrow 3 \equiv 3(mod\ 23).$$

Перевіривши правильність виконання операції, здійснимо додавання і знайдемо кутовий коефіцієнт січної.

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}(mod\ p) = \frac{7-10}{9-3}(mod\ 23) = -3/6(mod\ 23) =$$
$$= -1/2(mod\ 23) = 22/2(mod\ 23) = 11.$$

Далі знайдемо точки x та y:

$$x_3 = \lambda^2 - x_1 - x_2(mod\ p) = 121 - 3 - 9(mod\ 23) = 109(mod\ 23) = 17;$$
$$y_3 = \lambda(x_1 - x_3) - y_1(mod\ p) = 11(3 - 17) - 10(mod\ 23) = -164(mod\ 23) = 20.$$

Отже $P + Q = (3,\ 10) + (9,\ 7) = (17,\ 20)$.

Програмне розв'язання. Далі розглянемо програмне розв'язання, покажемо вивід (логіка програми описана нижче).

$$p1 + p2 = (17, 20)$$
$$p1 == p2 : False$$
$$p1! = p2 : True$$

У результаті можемо побачити як це виглядатиме на рисунку 9.

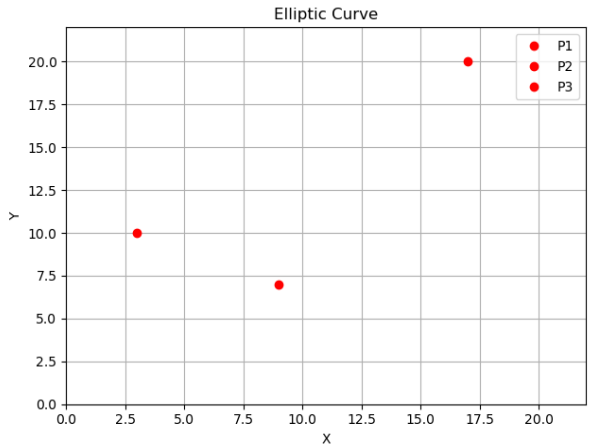


Рис. 6. Розв'язки отримані внаслідок програмної реалізації.

Код працює за такою логікою:

1) Клас EllipticCurvePoint визначає точку на еліптичній кривій. Конструктор `__init__` ініціалізує координати точки (x і y), параметри еліптичної кривої (a і b), які виступають коефіцієнтом біля x та вільним членом та поле простого числа (p).
2) Метод `__eq__` перевіряє, чи дві точки належать одній еліптичній кривій, порівнюючи координати x, y, a, b і p точок.
3) Метод `__ne__` перевіряє чи точки не належать кривій, використовуючи метод `__eq__`.

4) Метод `__add__` виконує додавання двох точок на еліптичній кривій. Він перевіряє, чи точки належать одній еліптичній кривій, обчислює кутовий коефіцієнт прямої або дотичної, залежно від того, чи точки рівні або різні, і обчислює координати третьої точки додавання.
5) Метод `__str__` повертає рядок, що представляє точку у форматі (x, y).
6) Задається поле простого числа p (у випадку даного прикладу використовується значення 23).
7) Створюються дві точки $P1$ і $P2$ з відповідними координатами і параметрами еліптичної кривої.
8) Виконується операція додавання двох точок: $P3 = P1 + P2$.
9) Малюємо графік з точками $P1, P2, P3$.

Файл з відповідним кодом програми завантажено на GitHub (див. [7]).

Алгоритм ECDSA. ECDSA (Elliptic Curve Digital Signature Algorithm) — це криптографічно безпечна схема цифрового підпису, заснована на криптографії еліптичних кривих (ECC). ECDSA спирається на математику циклічних груп еліптичних кривих над скінченними полями та на складність задачі ECDLP (задача дискретного логарифмування еліптичних кривих). Алгоритм підпису/перевірки ECDSA ґрунтується на точковому множенні еліптичних кривих. Ключі і підписи ECDSA коротші, ніж в RSA для того ж рівня безпеки [8]. Крім того, завдяки високій швидкості генерації підписів, цей алгоритм активно застосовується в сучасних протоколах безпеки, таких як TLS та SSH, що забезпечує захищену передачу даних в Інтернеті.

ECDSA використовує криптографічні еліптичні криві над скінченними полями в класичній формі Вейєрштрасса (див. рівняння 2).

У Python є спеціальна бібліотека, випущена за ліцензією MIT [9]. За допомогою цієї бібліотеки можна швидко створювати пари ключів (ключ підпису та ключ перевірки), підписувати повідомлення та перевіряти підписи. Також можна не лише створити, а й порівняти скільки часу потрібно кожній кривій для генерації пар ключів (keygen), підписання даних (sign), перевірки цих підписів (verify), отримання спільного секрету (ecdh) і перевірки підписів без попереднього обчислення ключа (no PC verify). Ось декілька популярних кривих: NIST, SECP, BRAINPOOLP. Він включає 256-бітну криву — secp256k1, яку використовує біткойн.

ECDSA стандартизований на міжнародному рівні організаціями, такими як NIST та SECG, що робить його сумісним із багатьма сучасними криптографічними системами. Він також використовується для забезпечення електронних підписів, що підтверджують автентичність документів, а також для автентифікації користувачів у веб-додатках. Завдяки цій універсальності, ECDSA став важливим елементом цифрової безпеки, забезпечуючи як цілісність даних, так і їх конфіденційність.

3. Висновки та перспективи подальших досліджень. Еліптичні криві є важливим інструментом сучасної криптографії, забезпечуючи високу стійкість до атак при оптимальних обчислювальних витратах. Алгоритм ECDSA, заснований на цих кривих, продемонстрував свою ефективність у багатьох галузях, зокрема у блокчейн-технологіях і системах захисту даних. Еліптичні криві дозволяють зменшити розмір ключів і підвищити швидкість обчислень без втрати рівня безпеки, що робить їх перспективними для подальшого розвитку

криптографії.

Еліптичні криві демонструють високу стійкість до класичних методів криптоаналізу, але існує необхідність подальшого дослідження стійкості цих алгоритмів в умовах появи квантових обчислювальних систем, які можуть загрожувати традиційним методам шифрування. (Зокрема, можливі атаки на основі квантового обчислення можуть значно скоротити час, необхідний для розв'язання задачі дискретного логарифмування на еліптичних кривих). Однак на сьогоднішній день ECDSA залишається одним із найбільш безпечних і ефективних криптографічних алгоритмів.

Важливо також відзначити гнучкість використання еліптичних кривих у різних протоколах безпеки. Окрім цифрових підписів, еліптичні криві застосовуються в шифруванні та генерації спільних ключів, що дозволяє будувати комплексні системи захисту з різнорівневим шифруванням.

Отже, на основі проведених досліджень і експериментів можна зробити висновок, що еліптичні криві та алгоритм ECDSA є важливими складовими сучасної криптографії, забезпечуючи високий рівень безпеки, ефективність та стійкість до більшості відомих атак. Це робить їх незамінними в сучасних інформаційних системах і криптографічних рішеннях. Проте для подальшого вдосконалення технологій захисту необхідно враховувати перспективи розвитку квантових обчислювальних систем і продовжувати дослідження в галузі стійкості криптографії на основі еліптичних кривих до нових загроз.

Список використаної літератури

1. Koblitz N. Elliptic curve cryptosystems. *Mathematics of Computation.* 1987. Vol. 48, No. 177. P. 203–209. DOI: https://doi.org/10.2307/2007884
2. Miller V. S. Use of Elliptic Curves in Cryptography. Advances in Cryptology — CRYPTO '85 Proceedings. Springer. Berlin : Heidelberg, 1985. P. 417–426. DOI: https://doi.org/10.1007/3-540-39799-X_31
3. Johnson D. B., Menezes A. J., Vanstone S. A. The elliptic curve digital signature algorithm (ECDSA). *International Journal of Information Security.* 2001. Vol. 1. P. 36–63. DOI: https://doi.org/10.1007/s102070100002
4. Brown D. R. L., Gallant R. P. The Static Diffie-Hellman Problem. *Cryptology ePrint Archive. Paper 2004/306.* 2004. URL: https://ia.cr/2004/306 (date of access: 21.07.2024).
5. Silverman J. H. The Arithmetic of Elliptic Curves. Berlin: Springer. 1986. P. 41–73. DOI: https://doi.org/10.1007/978-1-4757-1920-8
6. Криптографічні перетворення в групах точок еліптичних кривих. Метод. розроб. URL: https://learn.ztu.edu.ua/mod/resource/view.php?id=212444 (дата звернення: 12.08.2024).
7. GitHub : Elliptic curve and Finite Field. Python Source Code. URL: https://github.com/Sirko30/EllipticCurve/blob/main/ellipticcurves.py (date of access: 27.09.2024).
8. ECDSA : Elliptic Curve Signatures. Practical Cryptography for Developers. URL: https://cryptobook.nakov.com/digital-signatures/ecdsa-sign-verify-messages (date of access: 11.08.2024).
9. GitHub : Pure-python ECDSA signature/verification and ECDH key agreement. URL: https://github.com/tlsfuzzer/python-ecdsa (date of access: 01.09.2024).

Dobuliak L. P., Soltys S. P., Lysetska O. Y. Elliptic Curve Cryptography.

The article investigates in detail the mathematical aspects of elliptic curves, in particular the operations of adding and doubling points, which are the main operations for constructing cryptographic algorithms. Attention is focused on the properties of elliptic curves over finite fields, which makes them suitable for implementing strong cryptographic methods such as ECDSA (Elliptic Curve Digital Signature Algorithm). The mathematical

foundations of these operations, their algorithmic implementations, and their importance for computational security are discussed. In addition to the mathematical foundations, the article discusses the software implementation of point addition and doubling operations, which are important for the efficiency and security of cryptographic algorithms. The algorithms of implementation in the Python programming language are described.

Keywords: elliptic curve, cryptography, ECC, ECSDA, Python.

References

1. Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation,* 48(177), 203–209. https://doi.org/10.2307/2007884
2. Miller, V. S. (1985). *Use of Elliptic Curves in Cryptography.* Advances in Cryptology — CRYPTO '85 Proceedings. Springer, Berlin: Heidelberg, 417–426. https://doi.org/10.1007/3-540-39799-X_31
3. Johnson, D. B., Menezes, A. J., & Vanstone, S. A. (2001). The elliptic curve digital signature algorithm (ECDSA). *International Journal of Information Security,* 1, 36–63. https://doi.org/10.1007/s102070100002
4. Brown, D. R. L., & Gallant, R. P. (2004). The Static Diffie-Hellman Problem. *Cryptology ePrint Archive. Paper 2004/306.* Retrieved from https://ia.cr/2004/306
5. Silverman, J. H. (1986). *The Arithmetic of Elliptic Curves.* Berlin: Springer, 41–73. https://doi.org/10.1007/978-1-4757-1920-8
6. Cryptographic transformations in groups of points of elliptic curves. Method. Dev. Retrieved from https://learn.ztu.edu.ua/mod/resource/view.php?id=212444 [in Ukrainian].
7. GitHub : Elliptic curve and Finite Field. Python Source Code. Retrieved from https://github.com/Sirko30/EllipticCurve/blob/main/ellipticcurves.py
8. ECDSA : Elliptic Curve Signatures. Practical Cryptography for Developers. Retrieved from https://cryptobook.nakov.com/digital-signatures/ecdsa-sign-verify-messages
9. GitHub : Pure-python ECDSA signature/verification and ECDH key agreement. Retrieved from https://github.com/tlsfuzzer/python-ecdsa

Одержано 14.10.2024