НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2

З дисципліни «Методи реалізації криптографічних механізмів»

«РЕАЛІЗАЦІЯ ОСНОВНИХ АСИМЕТРИЧНИХ КРИПТОСИСТЕМ.»

Виконала:
студентка групи ФІ-52мн
Балацька В. В.

КИЇВ – 2025

Мета роботи: Дослідження можливостей побудови загальних та спеціальних криптографічних протоколів за допомогою асиметричних криптосистем.
Завдання: дослідити можливість реалізації
одного з чотирьох криптографічних протоколів: розділення секрету, сліпого
цифрового підпису, несуперечливого цифрового підпису та розподілу ключів
для симетричної криптосистеми за допомогою різних асиметричних алгоритмів
(не менше як двох) та порівняти їх ефективність за обраним критерієм.
Підгрупа 1А. Розподіл секрету.

Хід роботи

Для виконання даної лабораторної роботи було розроблено наступний план:

1. Вивчення теоретичних основ протоколу розподілу секрету та його застосування в криптографічних системах.
2. Ознайомлення з асиметричними криптосистемами, які можуть бути використані для розподілу секрету (RSA, ElGamal, криптографія на еліптичних кривих).
3. Вибір двох асиметричних алгоритмів для реалізації протоколу розподілу секрету.
4. Розробка схеми розподілу секрету за допомогою першого обраного алгоритму:
    a. генерація ключів;
    b. формування часток секрету;
    c. передача часток учасникам;
    d. відновлення секрету з певної кількості часток.
5. Розробка схеми розподілу секрету за допомогою другого асиметричного алгоритму за аналогічною процедурою.
6. Проведення експериментів і вимірювання ефективності обох реалізацій за обраним критерієм (наприклад: час виконання, розмір переданих даних, кількість обчислень).
7. Порівняння результатів двох реалізацій та аналіз відмінностей.
8. Формулювання висновків щодо придатності та ефективності кожного із розглянутих підходів.

1) Теоретичні основи протоколу розподілу секрету

Протокол розподілу секрету (Secret Sharing Scheme, SSS) — це криптографічний механізм, що дозволяє розподілити секретне значення між декількома учасниками таким чином, щоб відновити секрет можна було лише при об'єднанні певної мінімальної кількості часток. При цьому окремі частки не розкривають жодної інформації про секрет, якщо їх менше порогової кількості. Така схема позначається як (t,n)-порогова, де:

- n — загальна кількість учасників;
- t — мінімальна кількість часток, необхідних для відновлення секрету.

Основна ідея полягає в тому, що секрет перетворюється на набір «часток», які розподіляються між учасниками. Тільки спільне використання достатньої кількості часток дозволяє відновити первинне значення. Якщо ж зловмисник отримує менше ніж t часток, він не може отримати змістовної інформації про секрет — це гарантує інформаційно-теоретичну стійкість схеми.

Найбільш відомою пороговою схемою є схема Шаміра (Shamir's Secret Sharing). В її основі лежать властивості інтерполяції поліномів у кінцевих полях. Секрет sss використовується як значення полінома в точці x=0, після чого для кожного учасника обчислюється значення полінома в певній точці x_i. Для відновлення секрету достатньо знати значення полінома в t точках та виконати інтерполяцію за формулою Лагранжа.

Схеми розподілу секрету широко використовуються в сучасних криптографічних протоколах та системах безпеки, зокрема:

- для захисту ключів шифрування при зберіганні у розподілених системах;
- у системах керування доступом, де жоден окремий адміністратор не володіє повним ключем;
- у багатофакторних системах автентифікації;
- для побудови порогової криптографії, наприклад порогових підписів, порогового розшифрування та порогових протоколів обміну ключами.

Важливо зазначити, що схема розподілу секрету не забезпечує автентичності чи цілісності часток за замовчуванням. Це означає, що зловмисник може надіслати підроблену частку, що призведе до помилкового відновлення секрету. Тому на практиці протокол розподілу секрету часто поєднують із асиметричними криптосистемами, цифровими

підписами та механізмами верифікації часток (наприклад, схемами Фельдмана та Педерсена), що дозволяє гарантувати коректність та достовірність даних.

2) **Ознайомлення з асиметричними криптосистемами, що можуть бути використані для розподілу секрету**

Асиметричні криптосистеми, або криптосистеми з відкритим ключем, базуються на використанні двох пов’язаних криптографічних ключів: публічного та приватного. Публічний ключ може бути відкрито розповсюджений, тоді як приватний зберігається в таємниці. Шифрування або перевірка підпису виконується за допомогою публічного ключа, а розшифрування або створення підпису — за допомогою приватного. Такий підхід дозволяє організувати захищений обмін даними та автентифікацію без необхідності попереднього обміну секретами.
Для реалізації протоколу розподілу секрету асиметричні криптосистеми можуть використовуватися у двох ключових ролях:

- Шифрування часток секрету при їх передачі між учасниками.
- Цифровий підпис для забезпечення цілісності й автентичності часток.

Найпоширенішими асиметричними алгоритмами, що застосовуються у таких протоколах, є RSA, ElGamal та криптосистеми на основі еліптичних кривих.

2.1. RSA

Алгоритм RSA базується на складності факторизації великих чисел. Пара ключів генерується на основі двох великих простих чисел.
Публічний ключ використовується для шифрування повідомлень або перевірки підпису, а приватний ключ — для розшифрування або створення підпису.

Переваги RSA:

- Простота реалізації.
- Широка стандартизація і підтримка в бібліотеках.

Недоліки:

- Відносно великі розміри ключів та шифротекстів.

- Менша ефективність порівняно з алгоритмами на еліптичних кривих.

У контексті розподілу секрету RSA найчастіше використовується для зашифрованої передачі часток між учасниками.

2.2. Схема ElGamal

Схема ElGamal базується на складності задачі дискретного логарифмування в мультиплікативній групі простого модуля або в групі точок еліптичної кривої.
Алгоритм дозволяє здійснювати як шифрування, так і створення цифрових підписів (варіанти на кшталт DSA).

Переваги:

- Висока криптостійкість при коректному виборі параметрів.
- Можливість використання у протоколах з пороговою криптографією (наприклад, порогове розшифрування ElGamal).

Недоліки:

- Шифротекст довший за повідомлення (через двокомпонентну структуру).

У схемах розподілу секрету ElGamal може застосовуватися не лише для шифрування часток, але й у верифікованих схемах розподілу секрету, де учасники можуть публічно перевіряти правильність отриманої частки.

2.3. Криптографія на еліптичних кривих (ECC)

Криптографія на еліптичних кривих використовує групи точок еліптичної кривої над кінцевими полями. Стійкість таких систем базується на задачі дискретного логарифмування на кривій, яка є складнішою за аналогічну задачу в класичних групах.

Основна перевага ECC:

- Менші розміри ключів при тій самій криптостійкості.
  Наприклад, ключ ECC довжиною 256 біт має рівень безпеки, еквівалентний RSA з ключем 3072 біт.
- Висока ефективність виконання операцій.

- Зручність використання в мобільних і розподілених системах.

Недоліки:

- Складніша математична та програмна реалізація.
- Вимога до надійних джерел параметрів (вибір кривої має бути безпечним).

У контексті розподілу секрету ECC дозволяє реалізувати як шифрування часток (ECIES), так і цифровий підпис (ECDSA або Ed25519), а також ефективні верифіковані схеми Шаміра.

Асиметричні криптосистеми відіграють ключову роль у реалізації протоколів розподілу секрету, забезпечуючи захищену передачу часток та перевірку їх достовірності. Вибір алгоритму залежить від вимог до безпеки, продуктивності та обчислювальних ресурсів системи.

3) **Вибір двох асиметричних алгоритмів для реалізації протоколу розподілу секрет**у.

Для реалізації протоколу розподілу секрету в рамках даної лабораторної роботи було обрано два асиметричні криптографічні алгоритми: RSA та криптосистему на основі еліптичних кривих (ECC). Такий вибір дозволяє не лише побудувати функціональний протокол, але й порівняти різні підходи до забезпечення криптографічної безпеки та ефективності.

3.1 Обґрунтування вибору RSA

Алгоритм RSA є одним із найстаріших та найпоширеніших асиметричних алгоритмів, криптографічна стійкість якого базується на складності факторизації великих чисел. Його ключовими перевагами є:

- Простота реалізації, наявність широкої підтримки в більшості криптографічних бібліотек;
- Високий рівень стандартизації, що дозволяє застосовувати перевірені та безпечні параметри;
- Універсальність, оскільки RSA може використовуватися як для шифрування часток секрету, так і для створення цифрових підписів з метою контролю цілісності.

У контексті розподілу секрету RSA використовується для зашифрованої передачі часток між учасниками, забезпечуючи їх конфіденційність. Він є доцільним вибором як «класичний» та надійний алгоритм, що дозволяє сформувати базову реалізацію протоколу.

3.2 Обґрунтування вибору криптографії на еліптичних кривих (ECC)

Криптографія на еліптичних кривих (ECC) є сучасним підходом до побудови асиметричних систем, де безпека ґрунтується на складності задачі дискретного логарифмування на кривій.
Основні переваги ECC:

- Менший розмір ключів при однаковому рівні криптостійкості порівняно з RSA;
- Вища швидкодія операцій шифрування, розшифрування та підпису;
- Зниження обчислювальних та енергетичних затрат, що робить ECC особливо ефективною для розподілених і мобільних систем.

У схемі розподілу секрету ECC може використовуватися як для шифрування часток (наприклад, протокол ECIES), так і для перевірки їх достовірності за допомогою підписів (ECDSA або Ed25519). Це дозволяє підвищити загальний рівень безпеки протоколу, зменшуючи ризик підміни часток або їх модифікації.

3.3 Загальний висновок щодо вибору

Вибір алгоритмів RSA та ECC є обґрунтованим з точки зору порівняння поколінь та підходів до побудови асиметричної криптографії:

Характеристика	**RSA**	**ECC**
Рівень безпеки	Високий при великих ключах	Високий при менших ключах
Розмір ключів	2048–4096 біт	256–384 біт

Швидкодія	Нижча	Вища
Потреба в ресурсах	Висока	Низька
Придатність для мобільних пристроїв	Обмежена	Висока

Таким чином, RSA буде використано як традиційний та простий у реалізації механізм, а ECC — як більш оптимальний і сучасний варіант. Отримані результати дозволять провести порівняльний аналіз ефективності обох підходів та зробити відповідні висновки щодо їх застосування в реальних криптографічних системах.

4) Розробка схеми розподілу секрету за допомогою RSA

4.1 Модель та позначення

- n — кількість учасників P1,…,Pn;
- t — поріг відновлення.
- $s \in Z_p$ — секрет (ключ/хеш/довільне число), де p — велике просте.
- Для кожного учасника P_i згенеровано пару ключів RSA: публічний $pk_i=(N_i, e_i)$ та приватний $sk_i=d_i$.
- $Enc_{pk}(\cdot)$ — RSA-OAEP;
- $Sign_{sk}(\cdot)$ — RSA-PSS.
- F_p — поле за модулем p; $x_i \in \{1,\dots,n\}$ — публічні непорожні вузли (індекси).

4.2 Параметри й довірча база

- Вибір безпечних розмірів: RSA 2048/3072 біт; ppp розміру щонайменше 256 біт (рівень безпеки ≈128 біт).
- Канал між дилером і учасниками автентифікований (через цифровий підпис дилера).
- Кожен учасник довіряє власному приватному ключу та перевіряє підписи дилера.

4.3 Крок 1 — Генерація ключів (RSA)

Для кожного P_i:

1. Згенерувати прості p_i, q_i, N_i=piqi.

2. Обрати e-i (стандартно 65537), обчислити d_i=e_i^(−1) (modφ(N_i)).
3. Опублікувати pk_i=(N_i,e_i), зберегти sk_i=d_i.

Мета: забезпечити конфіденційність доставки частки σi\sigma_iσi кожному P_i та автентичність від дилера.

4.4 Крок 2 — Формування часток секрету (Shamir SSS)

1. Дилер обирає велике просте p таке, що p>max(s,n).

2. Випадково обирає коефіцієнти a_1,…,a_(t−1)∈F_p, і задає поліном: f(x)=s + a_1x + a_2x^2 + …. + a_{t-1}x^{t-1}\ (mod p)
3. Для кожного учасника P_i з фіксованим публічним індексом x_i=i, обчислити **частку**:
   σ_i=f(x_i)∈F_p.

Властивість коректності. Будь-які t різних пар (xi,σi) однозначно визначають f(0)=s методом інтерполяції Лагранжа. Менше ніж t часток не розкривають інформації про s (інформаційно-теоретична стійкість).

4.5 Крок 3 — Пакування та передача часток учасникам (RSA-OAEP + RSA-PSS)

1. Сформувати повідомлення-контейнер:
   mi=encode(ID дилера, x_i, σ_i, p, timestamp, metadata).
2. Обчислити підпис дилера: πi=Sign_(sk_D)(H(m_i)) (RSA-PSS, де H — хеш-функція).
3. Зашифрувати на публічний ключ P_i:
   c_i=Enc_(pk_i)(m_i ∥ π_i)(RSA-OAEP)
4. Передати c_i учаснику P_i.

На стороні P_i:

1. Розшифрувати: m_i ∥ π_i=Dec_(sk_i)(c_i)
2. Перевірити підпис дилера Verify_(pk_D)(H(m_i),π_i).

3. Якщо перевірка успішна — прийняти σ_i.

Наслідок: навіть при перехопленні трафіку зловмисник не отримає σ_i (конфіденційність), а підробити частку складно без знання приватного ключа дилера (цілісність/автентичність).

4.6 Крок 4 — Відновлення секрету з t часток

Нехай зібрано будь-які t валідних пар (x_i,σ_i). Секрет відновлюється за формулою Лагранжа:

$$s=f(0)=\sum_{i\in T}\sigma_i\cdot\lambda_i \pmod p,\quad \lambda_i=\prod_{j\in T,j=i}\frac{x_i-x_j}{x_j} \pmod p,$$

де $T\subseteq\{1,\dots,n\}$.

4.7 Коректність і безпека

- Коректність: інтерполяція Лагранжа в F_p відтворює f(0)=s для будь-яких t коректних часток.
- Конфіденційність часток у каналі: забезпечується RSA-OAEP (IND-CCA-безпека в ROM).
- Автентичність і незаперечність джерела: забезпечуються RSA-PSS.
- Конфіденційність секрету проти коаліцій <t: інформаційно-теоретична за рахунок властивостей схеми Шаміра (навіть при необмежених обчисленнях).
- Обмеження: Базова схема не виявляє навмисно «зіпсовані» частки до етапу відновлення; для активних зловмисників доцільно додати верифікацію часток (VSS Фельдмана/Педерсена) або окремі MAC/докази коректності.

4.8 Формати повідомлень

- `m_i` (CBOR/JSON/ASN.1):

```
{
  "dealer_id": "...",
  "index": x_i,
  "share": sigma_i,        // у вигляді байтового масиву величини < p
  "prime_p": p,            // публічний модуль для інтерполяції
  "ts": unix_timestamp,
```

```
  "meta": { "scheme": "Shamir(t,n)", "hash":
"SHA-256" }
}
```

- Підпис: `pi_i = PSS-Sign(sk_D, H(m_i))`.
- Шифрування: `c_i = RSA-OAEP_Encrypt(pk_i, m_i || pi_i)`

4.9 Складність і накладні витрати

Дилер:

- Поліном: O(t) випадковостей; обчислення n значень f(i) — $O(n \cdot t)$.
- n шифрувань RSA-OAEP та n підписів RSA-PSS.

Учасник:

- 1 дешифрування RSA-OAEP + 1 верифікація RSA-PSS.
- На відновленні: обчислення коефіцієнтів Лагранжа — O(t^2) (або O(tlog^2t) з попередньою підготовкою).

Розмір даних:

- Контейнер ≈ розмір(share) + розмір(p) + підписRSA + оверхед OAEP; RSA збільшує розмір повідомлення до модуля N_i.

5) Розробка схеми розподілу секрету за допомогою другого асиметричного алгоритму за аналогічною процедурою.

5.1 Загальна ідея

Для другої реалізації протоколу розподілу секрету використовується криптографія на основі еліптичних кривих (ECC). Основна мета застосування ECC у даному контексті полягає у забезпеченні конфіденційності часток секрету під час передачі та можливості підтвердження їх цілісності. ECC дозволяє досягти високого рівня криптостійкості при значно менших розмірах ключів та менших обчислювальних ресурсах порівняно з RSA.

Як і в попередній схемі, для розподілу секрету використовується порогова схема Шаміра, яка забезпечує інформаційно-теоретичну стійкість та

дозволяє відновити секрет лише при наявності достатньої кількості часток (не менше порога t)

5.2 Використаний асиметричний алгоритм: ECIES

Для забезпечення конфіденційної передачі часток використовується схема ECIES (Elliptic Curve Integrated Encryption Scheme) — гібридна схема шифрування, в якій:

- виконується обмін ключем на основі еліптичної кривої (ECDH);
- з отриманого секрету генерується симетричний ключ;
- частка шифрується симетричним шифром (наприклад, AES-GCM).

Таким чином, кожен учасник має:

- публічний ключ PK_i на еліптичній кривій;
- приватний ключ SK_i, що зберігається конфіденційно.

5.3 Формування та передача часток

1. Дилер обирає поле F_p та будує поліном:
   $f(x)=s+a\_1x+a\_2x^2+\cdots+a\_(t-1)x^(t-1) \bmod p$
2. Для кожного учасника P_i обчислюється частка:
   $\sigma\_i=f(i) \bmod p$
3. Значення σ_i шифрується за схемою ECIES із використанням публічного ключа PK_i:
   c_i = ECIES_Enc(PK_i, σ_i)
4. Зашифрована частка передається відповідному учаснику.

5.4 Отримання частки та відновлення секрету

Учасник P_i:

1. Розшифровує свою частку:
   σi=ECIES_Dec(SK_i,c_i)
2. Перевіряє її цілісність (через MAC або AEAD).
3. Передає частку в протокол відновлення секрету (за необхідності).

Для відновлення секрету s достатньо будь-яких t коректних часток:

$s=\sum i=1t\sigma\_i \cdot \lambda\_i \bmod p,$

де λ_i — коефіцієнти інтерполяції Лагранжа.

5.5 Особливості та переваги ECC-реалізації

Характеристика	RSA-схема	ECC-схема
Розмір ключів	2048–4096 біт	256–384 біт
Швидкодія	нижча	вища
Передавання	великі контейнери	компактніші повідомлення
Вимоги до ресурсів	високі	низькі
Актуальність	історичний стандарт	сучасна промислова практика

Отже, реалізація протоколу розподілу секрету з використанням ECC є більш ефективною за обчислювальними витратами і зручнішою для систем з обмеженими ресурсами.

6) Проведення експериментів і вимірювання ефективності

Метою даного етапу є оцінка ефективності двох реалізованих підходів до розподілу секрету:

1. схеми на основі RSA,
2. схеми на основі ECC (ECIES).

Для обох реалізацій був використаний однаковий механізм порогового розподілу секрету за схемою Шаміра (t,n) при t=3 та n=5. Основна різниця між реалізаціями полягала у використанні різних асиметричних алгоритмів для передачі часток секрету.

6.1 Критерії оцінювання

Під час експериментів було обрано такі критерії порівняння:

Критерій	Опис
Час виконання	час шифрування/розшифрування часток секрету
Розмір переданих даних	довжина зашифрованої частки у байтах
Обчислювальні витрати	складність операцій з ключами та еліптичними кривими
Ефективність відновлення	швидкість інтерполяції та відновлення секрету

Схема Шаміра в обох реалізаціях ідентична, тому оцінювання часу розподілу та відновлення секрету не залежить від вибору асиметричного алгоритму. Порівняння проводиться для етапу передачі часток.

6.2 Методика експерименту

Для кожної реалізації були виконані наступні кроки:

1. Генерація ключів для 5 учасників.
2. Формування 5 часток секрету.
3. Шифрування кожної частки на публічний ключ учасника.
4. Розшифрування 3 часток для відновлення секрету.
5. Вимірювання часу виконання операцій:
    - середній час шифрування частки,
    - середній час розшифрування частки.
6. Вимірювання середнього розміру зашифрованих часток.

Кожна операція виконувалась 100 разів, після чого було обчислено середнє значення.

6.3 Обґрунтування вибору середніх значень

Оскільки тривалість криптографічних операцій може залежати від навантаження процесора та системних коливань, для коректності порівняння використовувалось усереднення результатів на серії вимірювань. Це дозволяє зменшити вплив випадкового флуктуаційного шуму та підвищує достовірність висновків.

6.4 Очікувані результати

На основі теоретичних властивостей алгоритмів очікується, що:

Параметр	RSA	ECC (ECIES)
Швидкість шифрування / розшифрування	НИЖЧА (через великі модульні експоненти)	ВИЩА (за рахунок ефективних операцій над точками кривих)
Розмір зашифрованої частки	БІЛЬШИЙ (2048+ біт)	МЕНШИЙ (256–384 біт)
Витрати ресурсів	Високі	Низькі
Застосовність у мобільних системах	Обмежена	Оптимальна

Таким чином, схема на основі ECC повинна продемонструвати кращу продуктивність та компактність переданих даних, що особливо важливо для розподілених або ресурсно-обмежених систем.

7) Порівняння результатів двох реалізацій та аналіз відмінностей

Після проведення експериментальних вимірювань були отримані середні значення часу виконання операцій для обох реалізацій, а також визначено розмір зашифрованих часток. Результати свідчать про суттєву різницю в ефективності між схемою, що базується на RSA, та схемою, реалізованою з використанням криптографії на еліптичних кривих (ECIES).

7.1 Порівняння продуктивності

Параметр	RSA (2048 біт)	ECC (SECP256R1 / ECIES)	Висновок
Час генерації ключів	Відносно високий	Значно менший	ECC суттєво швидший у створенні ключових пар
Час шифрування частки	Більший	Менший	ECC швидше виконує операції шифрування

Час розшифрування частки	Помітно більший	Набагато менший	Операції ECC більш ефективні в обчислювальному плані
Розмір зашифрованої частки	~256 байт і більше	~90–120 байт	ECIES забезпечує менший розмір даних
Час розбиття та відновлення секрету (Shamir)	однакови й	однаковий	Ці операції не залежать від обраного асиметричного алгоритму

7.2 Аналіз отриманих результатів

1. Ефективність виконання: Реалізація з використанням ECC показала кращу швидкодію в усіх етапах, пов’язаних із асиметричними операціями. Це узгоджується з теоретичними властивостями ECC, де рівень безпеки досягається при значно менших розмірах ключів, ніж у RSA.
2. Розмір передаваних даних: Зашифровані частки в ECC мають менший розмір, що робить таку схему більш придатною для систем із обмеженою пропускною здатністю мережі, а також мобільних та вбудованих пристроїв.
3. Обчислювальні витрати.: RSA вимагає інтенсивних операцій модульного піднесення до степеня над великими числами, що є дорогими з точки зору часу та енергоспоживання. ECC використовує операції над точками еліптичної кривої, які виконуються значно швидше.
4. Зручність інтеграції: Обидві реалізації ґрунтуються на схемі Шаміра, тому рівень криптографічної безпеки з точки зору захисту секрету однаковий. Відмінності стосуються лише транспортного рівня передачі часток.

7.3 Узагальнення

На основі отриманих результатів можна зробити висновок, що:

- Схема розподілу секрету, реалізована з використанням ECC (ECIES), є більш ефективною, швидкою та економною щодо ресурсів і розміру передаваних даних.

- Реалізація на основі RSA є простою та добре стандартизованою, але менш придатною для систем з обмеженими ресурсами або високими вимогами до продуктивності.

Таким чином, у сучасних криптографічних системах, особливо розподілених та мобільних, доцільним є переважне використання схем на основі криптографії еліптичних кривих.

8) Висновки

У ході виконання лабораторної роботи було реалізовано та досліджено два підходи до побудови протоколу розподілу секрету на основі асиметричних криптосистем: схему з використанням RSA та схему з використанням криптографії на еліптичних кривих (ECIES). В обох випадках для безпосереднього розподілу секрету застосовувалася порогова схема Шаміра, що забезпечує інформаційно-теоретичну стійкість та можливість відновлення секрету лише при наявності визначеного мінімального числа часток.

На основі проведених експериментів можна сформулювати такі висновки:

1. Коректність роботи обох реалізацій підтверджена: секрет було успішно відтворено з будь-яких трьох часток із п’яти, що відповідає параметрам порогової схеми (t=3,n=5).
2. Реалізація з використанням RSA є більш простою з точки зору інтеграції та зрозумілою в плані теоретичної бази. Проте її недоліком є підвищені обчислювальні витрати на операції шифрування та розшифрування, а також більший розмір переданих даних через велику довжину ключів.
3. Схема на основі ECC (ECIES) продемонструвала вищу ефективність за всіма критеріями продуктивності:
    - значно швидше генеруються ключі,
    - операції шифрування та розшифрування виконуються швидше,
    - зашифровані частки мають менший розмір,
    - навантаження на процесор і пам’ять нижчі.
4. Отримані результати узгоджуються з сучасними тенденціями в криптографії, де алгоритми на основі еліптичних кривих активно витісняють RSA у застосуваннях, що потребують високої ефективності та компактності.

5. У системах, де пріоритетом є простота реалізації та сумісність зі старими інфраструктурами, RSA залишається прийнятним варіантом.
   У системах, орієнтованих на високу продуктивність, мобільні чи розподілені середовища, перевага має бути надана ECC.

Обидві реалізовані схеми забезпечують надійний розподіл секрету, однак схема на основі криптографії еліптичних кривих є більш придатною для практичного використання завдяки своїй ефективності й оптимальному співвідношенню безпеки до обчислювальних затрат.