НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів захисту інформації

«На правах рукопису»
УДК 519.2

«До захисту допущено»
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2022 р.

Магістерська дисертація

на здобуття ступеня магістра
за освітньо-професійною програмою
«Математичні методи криптографічного захисту інформації»

зі спеціальності: 113 Прикладна математика
на тему: **«Удосконалення гріндінг-атаки на протокол консенсусу PoS при неправильній генерації випадкового вектора»**

Виконав:
студент II курсу, групи ФІ-12мп
Тулупов Матвій Миколайович _____

Керівник:
д.т.н., проф., проф. кафедри ММЗІ
Ковальчук Людмила Василівна _____

Рецензент:
К.т.н., доцент кафедри інформаційної безпеки
Ірина Валеріївна Стьопочкіна _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

Київ — 2022

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів захисту інформації

Рівень вищої освіти — другий (магістерський)
Спеціальність — 113 Прикладна математика,
ОПП «Математичні методи криптографічного захисту інформації»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2022 р.

ЗАВДАННЯ
на магістерську дисертацію

Студент: Тулупов Матвій Миколайович

1. Тема роботи: «*Удосконалення гріндінг-атаки на протокол консенсусу PoS при неправильній генерації випадкового вектора*», науковий керівник дисертації: д.т.н., проф., проф. кафедри ММЗІ Ковальчук Людмила Василівна,

затверджені наказом по університету №__ від «__» _____ 2022 р.

2. Термін подання студентом роботи: «__» _____ 2022 р.

3. Об'єкт дослідження: *алгоритм атаки на блокчейн Тезос*

4. Предмет дослідження: *розгляд та порівняння різних стратегій атаки на блокчейн Тезос*

5. Перелік завдань: *виведення формул для оцінки різних стратегій атаки та їх порівняння між собою*

6. Орієнтовний перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): *презентація доповіді*

7. Орієнтовний перелік публікацій: *планується доповідь на всеукраїнській конференції*

8. Дата видачі завдання: 10 вересня 2021 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	01-15 вересня 2021 р.	Виконано
2	Огляд опублікованих джерел за тематикою дослідження	Вересень-листопад 2021 р.	Виконано
3	Виведення формул для подальшого обчислення математичних сподівань і ймовірностей, що потрібні для оцінки атаки	Листопад 2021 р.-серпень 2022 р.	Виконано
4	Написання програм для обчислення оцінок атак по отриманим раніше формулам й порівняння результатів між собою	Серпень-жовтень 2022 р.	Виконано
5	Оформлення усіх отриманих результатів та їх опис	Жовтень-грудень 2022 р.	Виконано

Студент _____ Матвій ТУЛУПОВ

Керівник _____ Людмила КОВАЛЬЧУК

РЕФЕРАТ

Кваліфікаційна робота містить: 62 стор., 17 таблиць, 2 джерела.

Метою цього дослідження є розгляд різних стратегій атаки на блокчейн Тезос та отримання для них числових оцінок. Оцінки різних стратегій атак були згруповані до таблиць для очного порівняння їх між собою.

Об'єктом дослідження є алгоритм атаки на блокчейн Тезос.

Предмет дослідження є розгляд та порівняння різних стратегій атаки на блокчейн Тезос.

Результати порівняння показали, що найефективнішою стратегією проведення атаки на блокчейн Тезос у довгостроковій перспективі є проведення атаки в одну фазу (без додаткового нарощування кількості зобов'язань).

ТЕЗОС, ЗОБОВ'ЯЗАННЯ, СТРАТЕГІЇ АТАКИ, МАТЕМАТИЧНЕ СПОДІВАННЯ КІЛЬКОСТІ БЛОКІВ ПІСЛЯ АТАКИ

ABSTRACT

The qualification work contains: 62 pages, 17 tables, 2 sources.

The purpose of this study is to examine different attack strategies on the Tezos blockchain and obtain numerical evaluations for them. Scores of different attack strategies were grouped into tables for visual comparison.

The object of the study is the attack algorithm on the Tezos blockchain.

The subject of the study is the consideration and comparison of various attack strategies on the Tezos blockchain.

The results of the comparison showed that the most effective strategy for conducting an attack on the Tezos blockchain in the long term is to conduct a single-phase attack (without additionally increasing the number of commitments).

TEZOS, COMMITMENTS, ATTACK STRATEGIES, MATHEMATICAL EXPECTATION OF THE NUMBER OF BLOCKS AFTER ATTACK

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

N - кількість усіх блоків

n - кількість блоків зобов'язаннями

$M_{\zeta}^{S}(X)$ — математичне сподівання кількості блоків після атаки з S фаз на $X \in \{n,N\}$

$P^{*}\{\eta \leqslant T\} = P\{\eta \leqslant T \mid X = n\}$

ВСТУП

Актуальність дослідження полягає у тому, що криптовалюти зараз є дуже популярними й дослідження можливих атак на них допоможе розробникам захистити своїх користувачів від втрати грошей. Поглиблене вивчення можливих атак дозволяє краще виявляти зловмисників серед чесних користувачів, а розуміння того, як зловмисник буде діяти підчас проведення тієї чи іншої атаки, дає змогу побачити, як саме буде діяти зловмисник для максимізації ефективності своєї атаки. Таким чином, це дослідження дозволить краще розуміти, що саме буде робити зловмисник, й, використовуючи цю інофрмацію, блокувати його в мережі.

Метою дослідження є повне дослідження гріндінг-атаки й розгляд можливих варіантів поведінки зловмисника при її проведенні.

Задачами дослідження є:

1) провести аналіз гріндінг-атак й навести отримані в попередній роботі формули;

2) отримати нові формули, для порівняльного аналізу наведених стратегій поведінки зловмисника при проведенні гріндінг-атаки;

3) обчислити оцінки ефективності усіх доліджуваних стратегій;

4) виконати порівняльний аналіз оцінок ефективності різних стратегій.

Об'єктом дослідження є інформаційні процеси при проведенні гріндінг-атаки на протокол консенсусу PoS.

Предметом дослідження є кількість фаз проведення гріндінг-атаки на протокол консенсусу PoS.

Наукова новизна отриманих результатів полягає у тому, що вперше розглянуто, досліджено та порівняно між собою різної кількості фаз проведення гріндінг-атаки зловмисником.

Практичне значення результатів полягає у тому, що розуміння

того, як повинен вести себе зловмисник допоможе краще його виявляти та протидіяти йому.

1 АТАКА НА БЛОКЧЕЙН ТЕЗОС

В цьому розділі буде описана атака на блокчейн Тезос та розглянуті результати попередніх досліджень цієї атаки.

1.1 Опис структури Тезоса

Тезос є блокчейн платформою з особливим протоколом консенсусу - LPOS (Liquid Proof of Stake), який є дещо модифікованим протоколом консенсусу DPOS (Delegated Proof of Stake). Головна відмінність полягає в тому, що у LPOS немає фіксованої кількості делегатів і їх вибір не є обов'язковим. Список делегатів є динамічним, це означає, що туди можна включати нових делегатів та виключати діючих. [4, 5, 6]

Означення 1.1. Випічка блоків (англ. *baking*) в Тезосі - це створення блоків, а ті люди, що створюють блоки звуться пекарями. Створюючи блок пекарь його підписує. [2]

Випадково обраний учасник стає пекарем, а інші 32 випадково обраних учасника стають ендорсерами (від англ. *endorse* - схвалювати), вони повинні підтвердити створений блок. За створення та підтвердження блока видається винагорода, що стимулює до активної участі в алгоритмі консенсусу. [3]

Усі блоки у Тезосі групуються в цикли по 4096 блоків на цикл. Кожен блок створюється приблизно 60 секунд, отже приблизна тривалість циклу - 2 дні 20 годин 16 хвилин.

Ймовірність вибору учасника пекарем залежить від його частки серед усіх токенів Тезосу (XTZ). Частка поділена на порції по 8000 XTZ в кожній, меньше однієї порції не враховується в частці, тому сумарна частка - відношення кількості таких порцій у делегата таких порцій до кількості усіх порцій учасників.

Права пекаря на створення блоку в циклі n отримує випадково обраний учасник з циклу $n - 7$. Для кожного циклу генерується випадкове число, яке далі використовується в генераторі псевдовипадкових чисел, результат якого й допомагає призначити права пекаря. Самей цей генератор псевдовипадкових чисел, та його недоліки використовуються для гріндінг-атаки.

Разом з пекарем, у той же час обираються й ендорсери, що повинні підтвердити створені блоки в циклі n, підтвердження «запікається» в наступному блоці.

Для розподілу прав серед учасників система генерує певне псевдовипадкове число. Алгоритм його генерації є відкритим і кожен може подивитись, як він формується, або перевірити правильність. Генерація цього числа для розподілу прав у циклі під номером n відбувається у циклі під номером $n - 2$.

Означення 1.2. Зобов'язання (англ. *commitment*) - випадкове число, що дістається певним учасникам й гешується. Кожен з учасників може відкрити своє зобов'язання, або не показувати його. Усі відкриті зобов'язання використовуються в алгоритмі формування псевдовипадкового числа, що відповідає за розподіл прав.

Усі зобов'язання для циклу n видаються в циклі $n - 2$, а повинні бути відриті в циклі $n - 1$. Також, повинно бути відкрито випадкове число з зобов'язання до гешування. Одне зобов'язання може бути згенероване кожні 32 блоків, отже кожен цикл їх приблизно 128.

1.2 Концепція гріндінг-атаки

Процедуру розподілу прав у Тезосі можна спростити до популярних розподілів. Якщо доля зловмисника дорівнює p, то внаслідок процедури вибору пекарів, що керується протоколом DPOS, імовірність того, що пекарем буде зловмисник, дорівнює p. Це стандартна вимога до цієї

процедури, і всі алгоритми вибору слотлідерів повинні мати таку властивість. Отже ймовірність стати пекарем одного блоку буде мати розподіл Бернуллі, а отже, ймовірність стати пекарем l з X блоків буде мати біноміальний розподіл.

Без проведення гріндінг-атаки, середня кількість блоків з зобов'язаннями зловмисника з долею p буде дорівнювати $128p$ (це є наслідком того, що середня кількість зобов'язань у циклі дорівнює 128). Як і сказано в протоколі розподілу прав, зловмисник повинен згенерувати випадкові числа по кожному його зобов'язанню й на наступному циклі викласти їх.

За умови того, що усі інші учасники з зобов'язаннями є чесними й публікують усі свої зобов'язання, то зловмисник може їх побачити й розрахувати випадкове число, від якого залежить розподіл пекарських прав. Зловмисник може не викладати усі свої зобов'язання, а порахувати випадкові числа для усіх можливих комбінацій публікації своїх зобов'язань й побачити кількість блоків, які він зможе згенерувати після відкриття відповідних комбінацій. Таким чином, зловмисник може контролювати випадковий розподіл пекарських прав й збільшувати свій прибуток за рахунок чесних учасників.

1.3 Аналіз імовірностей атаки з вибірковим відкриттям зобов'язань

Для формулювання покращених стратегій проведення гріндінг-атаки, потрібно привести ймовірності, що найкращим чином описують атаку з вибірковим відкриттям зобов'язань, з попереднього дослідження цієї атаки.

Лема 1.1. *Для $X \in \{n,N\}$, $l \in \{0,1,\ldots,X\}$, $K \in N$, імовірність P, що хоча б в одному з K варіантів перебору, кількість пекарських прав*

зловмисника з X, буде не менше ніж l, дорівнює:

$$P = 1 - \left(1 - \sum_{k=l}^{X} \binom{X}{k} p^k q^{X-k}\right)^K \tag{1}$$

Приведене рівняння ймовірності дозволяє порахувати, з якою ймовірністю гріндінг-атака, з метою отримати щонайменше l з X блоків пройде успішно.

Воно підходить лише до класичної гріндінг-атаки, яка проходить в одну фазу й де зловмисник вже знає кількість своїх зобов'язань. Таким чином, кількість його варіантів K буде дорівнювати $2^{\tilde{K}}$, де $\tilde{K}$ - кількість зобов'язань зловмисника.

Теорема 1.1. *Нехай зловмисник намагається збільшити кількість своїх блоків з зобов'язаннями, чи загальну кількість блоків у циклі під номером $i+5$, використовуючи свої зобов'язання з циклу $i-2$. Тоді можна визначити ймовірність події $B(X,l)$, де:*

$$B(X,l) = \{B \text{ циклі під номером } i+5$$

$$\text{зловмисник отримує не менше ніж } l \text{ з } X \text{ блоків}\}$$

$$P(B(X,l)) = \sum_{j=0}^{n} \left(1 - (1 - (\sum_{k=l}^{X} \tbinom{X}{k} p^k q^{X-k}))^{2^j}\right) \cdot \tbinom{n}{j} p^j q^{n-j}, \tag{4}$$

де p - розмір стейку зловмисника, $X \in \{n,N\}$.

Приведена теорема, на відміну від першої леми, вже враховує ймовірності отримати зловмисником певну кількість зобов'язань й дозволяє оцінити ймовірність отримати щонайменше l з X блоків, ще не знаючи, скільки у нього буде зобов'язань. Але, як і попередня ймовірність, ця - підходить також лише для оцінки гріндінг-атаки в одну фазу. [1]

Висновки до розділу 1

У даному розділі була описана гріндінг-атака, а також, були наведені математичні результати з попередніх досліджень цієї атаки.

Крім цього, були наведені деякі потрібні для дослідження теоретичні відомості про блокчейн Тезос.

2 УСПІШНІСТЬ СТРАТЕГІЙ АТАКИ

У даному розділі будуть виведені формули ймовірності і математичного сподівання кількості блоків після атаки, які потім будуть використані для розрахунку оцінок ефективності різних стратегій атаки й порівняння їх між собою.

2.1 Ймовірність атаки

Нехай ξ - кількість блоків, що отримує учасник без проведення атаки. Випадкова величина ξ має біноміальний розподіл (X, p), отже ймовірність того, що вона буде дорівнювати l буде:

$$P\{\xi = l\} = \binom{X}{l} p^l q^{X-l}$$

Також відомо, що ймовірність того, що випадкова величина ξ за біноміальним розподілом (X, p) буде менше, ніж l, буде дорівнювати:

$$P\{\xi < l\} = \sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}$$

Теорема 2.1. *Нехай η - випадкова величина, що дорівнює кількості отриманих блоків в результаті однієї фази атаки з K зобов'язаннями. Тоді ймовірність того, що η буде мати значення l дорівнює:*

$$P\{\eta = l\} = \left(\sum_{k=0}^{l} \binom{X}{k} p^k q^{X-k}\right)^{2^K} - \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K}$$

Доведення.

Нехай ξ_1,...,ξ_{2^K} - незалежні випадкові величини з біноміальним розподілом (X, p). Тоді ймовірність того, що S з них будуть дорівнювати

l, а усі інші - менше ніж l, буде добутком відповідних ймовірностей, написаних вище, помноженим на кількість усіх можливих перестановок S серед 2^K варіантів:

$$\binom{2^K}{S} \cdot \prod_{t=1}^{S} P\{\xi_t = l\} \cdot \prod_{t=S+1}^{2^K-c} P\{\xi_t < l\} =$$

$$= \binom{2^K}{S} \cdot P\{\xi_1 = l\}^S \cdot P\{\xi_1 < l\}^{2^K-S} =$$

$$= \binom{2^K}{S} \cdot \left(\binom{X}{l} p^l q^{X-l}\right)^S \cdot \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K-S}$$

Сума наведених імовірностей по усіх значеннях S знаходиться за допомогою біному Ньютона:

$$\sum_{S=0}^{2^K} \binom{2^K}{S} \cdot \left(\binom{X}{l} p^l q^{X-l}\right)^S \cdot \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K-S} =$$

$$= \left(\binom{X}{l} p^l q^{X-l} + \sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K} =$$

$$= \left(\sum_{k=0}^{l} \binom{X}{k} p^k q^{X-k}\right)^{2^K}$$

Нехай η - випадкова подія того, що серед 2^K незалежних випадкових величин з біноміальним розподілом (X, p) максимальним значенням буде l. Тоді ймовірність η дорівнює:

$$P\{\eta = l\} = \sum_{S=1}^{2^K} \binom{2^K}{S} \cdot \left(\binom{X}{l} p^l q^{X-l}\right)^S \cdot \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K-S} =$$

$$= \left(\sum_{k=0}^{l} \binom{X}{k} p^k q^{X-k}\right)^{2^K} - \left(\binom{X}{l} p^l q^{X-l}\right)^0 \cdot \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K-0} =$$

$$= \left(\sum_{k=0}^{l} \binom{X}{k} p^k q^{X-k}\right)^{2^K} - \left(\sum_{k=0}^{l-1} \binom{X}{k} p^k q^{X-k}\right)^{2^K}$$

□

Оскільки кількість отриманих блоків з X у результаті атаки на блокчейн Тезос може бути представлена як випадкова величина з біноміальним розподілом (X, p), то отримана ймовірність у термінах атаки є ймовірністю отримати рівно l з X блоків (l - максимальне значення серед усіх переборів, отже в результаті атаки зловмисник вибере саме таку конфігурацію) у результаті атаки з K блоками з коммітментами (а отже з 2^K варіантами перебору).

Отримання цієї ймовірності допоможе розрахувати математичне сподівання кількості блоків, отриманих зловмисником в результаті атаки, що дозволить порахувати середню кількість блоків при використанні певних стратегій атаки.

Нехай ζ - кількість блоків з X, отриманих в результаті атаки з використанням K блоків з коммітментами, тоді:

$$P\{\zeta = l\} = P\{\eta \leqslant l\}^{2^K} - P\{\eta < l\}^{2^K}$$

Підчас розрахунку середньої кількості блоків різних стратегій атаки з'явиться проблема розрахунку таких ймовірностей, так як ця різниця - різниця дуже малих чисел у дуже великих степенях.

2.2 Математичне сподівання кількості блоків за атаку

Використовуючи формулу визначення математичного сподівання, можно отримати його значення для кількості блоків з X, отриманих в

результаті атаки з використанням K блоків з коммітментами ζ:

$$M_\zeta(K,X) = \sum_{l=1}^{X} l\left(P\left\{\eta \leqslant l\right\}^{2^K} - P\left\{\eta < l\right\}^{2^K} \right)$$

Отримана формула математичного сподівання залежить від змінної K, але K - це також випадкола величина з біноміальним розподілом (n, p), де n - максимальна кількість блоків з коммітментами в циклі. Оскільки початкова кількість блоків з коммітментами не є результатом атаки, то її ймовірність - це ймовірність біноміально розподіленої випадкової величини:

$$P\left\{\xi = K\right\} = \binom{n}{K} p^K q^{n-K}$$

Таким чином, математичне сподівання кількості блоків з X, отриманих в результаті атаки буде дорівнювати:

$$M_\zeta^1(X) = \sum_{K=0}^{n} \sum_{l=1}^{X} l\left(P\left\{\eta \leqslant l\right\}^{2^K} - P\left\{\eta < l\right\}^{2^K} \right) \cdot \binom{n}{K} p^K q^{n-K} =$$

$$= \sum_{K=0}^{n} \left(\binom{n}{K} p^K q^{n-K} \cdot \sum_{l=1}^{X} l\left(P\left\{\eta \leqslant l\right\}^{2^K} - P\left\{\eta < l\right\}^{2^K} \right) \right) =$$

$$= \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} M_\zeta(K,X)$$

Формулу математичного сподівання кількості блоків з X, отриманих в результаті атаки з використанням K блоків з коммітментами можна дещо спростити, що дозволить легше розраховувати ефективність стратегій:

Теорема 2.2. *Нехай ζ - випалкова величина, що дорівнює кількості отриманих блоків в результаті однієї фази атаки з K зобов'язаннями. Тоді математичне сподівання ζ дорівнює:*

$$M_\zeta(K,X) = X - \sum_{l=0}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K}$$

Доведення.

$$M_\zeta(K,X) = \sum_{l=1}^{X} l\Big(P\left\{\eta \leqslant l\right\}^{2^K} - P\left\{\eta < l\right\}^{2^K} \Big) =$$

$$= \sum_{l=1}^{X} l\Big(P\left\{\eta \leqslant l\right\}^{2^K} - P\left\{\eta \leqslant l-1\right\}^{2^K} \Big) =$$

$$= \sum_{l=1}^{X} l \cdot P\left\{\eta \leqslant l\right\}^{2^K} - \sum_{l=1}^{X} l \cdot P\left\{\eta \leqslant l-1\right\}^{2^K} =$$

$$= \sum_{l=1}^{X} l \cdot P\left\{\eta \leqslant l\right\}^{2^K} - \sum_{l=0}^{X-1} (l+1) \cdot P\left\{\eta \leqslant l\right\}^{2^K} =$$

$$= X \cdot P\left\{\eta \leqslant X\right\}^{2^K} - P\left\{\eta = 0\right\}^{2^K} + \sum_{l=1}^{X-1} (-1) \cdot P\left\{\eta \leqslant l\right\}^{2^K} =$$

$$= X \cdot 1 - q^{X \cdot 2^K} - \sum_{l=1}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K} =$$

$$= X - \sum_{l=0}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K}$$

$$\Rightarrow M_\zeta(K,X) = X - \sum_{l=0}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K}$$

□

Наслідок 2.1.

$$M_\zeta^1(X) = \sum_{K=0}^{n} \left(\binom{n}{K} p^K q^{n-K} \cdot \left(X - \sum_{l=0}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K} \right) \right)$$

$$M_\zeta^1(X) = X - \sum_{K=0}^{n} \left(\binom{n}{K} p^K q^{n-K} \cdot \sum_{l=0}^{X-1} P\left\{\eta \leqslant l\right\}^{2^K} \right)$$

Наслідок 2.2.

$$M_\zeta^1(X) = X - \sum_{l=0}^{X-1} M_K(P\{\eta \leqslant l\}^{2^K})$$

$$K \sim Bin(X,p)$$

Доведення.

$$M_\zeta^1(X) = X - \sum_{K=0}^{n} \left(\binom{n}{K} p^K q^{n-K} \cdot \sum_{l=0}^{X-1} P\{\eta \leqslant l\}^{2^K} \right)$$

$$\Rightarrow M_\zeta^1(X) = X - \sum_{l=0}^{X-1} \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} P\{\eta \leqslant l\}^{2^K}$$

По формулі математичного сподівання випадкової величини вираз можна згорнути:

$$\Rightarrow M_\zeta^1(X) = X - \sum_{l=0}^{X-1} M_K(P\{\eta \leqslant l\}^{2^K})$$

$$K \sim Bin(X,p)$$

□

Аналогічно можна отримати формулу математичного сподівання кількості блоків з X, отриманих в результаті атаки з нарощуванням кількості блоків з коммітментами протягом $F-1$ циклів.

Теорема 2.3. *Нехай ζ - випадкова величина, що дорівнює кількості отриманих блоків в результаті K фаз атаки. Тоді ймовірність того, що ζ буде мати значення l дорівнює:*

$$P_F\{\zeta = l\}(X) = \underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_F=0}^{n}}_{F} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot\prod_{t=1}^{F-1}\left(P^*\{\eta\leqslant K_{t+1}\}^{2^{K_t}}-P^*\{\eta\leqslant K_{t+1}-1\}^{2^{K_t}}\right)$$

$$\cdot\left(\left(P\{\eta\leqslant l\}(X)\right)^{2^{K_{F-1}}}-\left(P\{\eta\leqslant l-1\}(X)\right)^{2^{K_{F-1}}}\right)$$

Доведення.

Ймовірність отримати l блоків після F фаз атаки залежить від кількості зобов'язань, отриманих на попередньому кроці, а кількість зобов'язань на попередньому - від їх кількості на кроці до нього.

Таким чином, можна отримати ймовірність отримати l блоків після F фаз атаки, закріпивши певну кількість зобов'язань на кожному кроці. Отже ймовірність за усі фази буде ймовірністю за умови виконання усіх умові, а саме - отримання відповідної кількості блоків на кожному кроці. Ймовірність кожного наступного кроку залежить тільки від ймовірності на попередньому кроці, отже можна розписати ймовірність у ланцюг з добутку ймовірностей:

$$P_F\left\{\zeta=l\mid K_1=\tilde{K_1},...,K_F=\tilde{K_F}\right\}(X)=$$

$$=P\left\{\zeta=l\mid K_F=\tilde{K_F}\right\}(X)\cdot P\left\{K_F=\tilde{K_F}\mid K_{F-1}=\tilde{K_{F-1}}\right\}(X)\cdot$$

$$\cdot P\left\{K_{F-1}=\tilde{K_{F-1}}\mid K_{F-2}=\tilde{K_{F-2}}\right\}(X)\cdot\ldots\cdot$$

$$\cdot P\left\{K_2=\tilde{K_2}\mid K_1=\tilde{K_1}\right\}(X)\cdot P\left\{K_1=\tilde{K_1}\right\}(X)=$$

$$=\binom{n}{\tilde{K_1}}p^{\tilde{K_1}}q^{n-\tilde{K_1}}\cdot$$

$$\cdot\prod_{t=1}^{F-1}\left(P^*\left\{\eta\leqslant\tilde{K}_{t+1}\right\}^{2^{\tilde{K}_t}}-P^*\left\{\eta\leqslant\tilde{K}_{t+1}-1\right\}^{2^{\tilde{K}_t}}\right)$$

$$\cdot\left(\left(P\left\{\eta\leqslant l\right\}(X)\right)^{2^{\tilde{K}_{F-1}}}-\left(P\left\{\eta\leqslant l-1\right\}(X)\right)^{2^{\tilde{K}_{F-1}}}\right)$$

Отримана ймовірність - ймовірність отримати ймовірність отримати l блоків після F фаз атаки за умови того, що кількість зобов'язань на кожному кроці буде мати конкретне значення. Отже, щоб отримати загаульну ймовірність отримати ймовірність отримати l блоків після F фаз атаки - потрібно просумувати цю ймовірність по усім можливим комбінаціям кількості зобов'язань на кожному кроці, що й доводить теорему.

□

Наслідок 2.3. *З отриманої в теоремі 2.3 ймовірності можна легко отримати математичне сподівання кількості блоків після F фаз атаки, просумувавши по усім можливим кількостям блоків на відповідну ймовірність:*

$$M_\zeta^F(X)=\underbrace{\sum_{K_1=0}^{n}\ldots\sum_{K_F=0}^{n}}_{F}\binom{n}{K_1}p^{K_1}q^{n-K_1}\cdot M_\zeta(X,K_F)\cdot$$

$$\cdot\prod_{t=1}^{F-1}\left(P^*\{\eta\leqslant K_{t+1}\}^{2^{K_t}}-P^*\{\eta\leqslant K_{t+1}-1\}^{2^{K_t}}\right)$$

Таким чином, для малої кількості фаз атаки формула математичного сподівання буде виглядати так:

$$M_\zeta^1(X)=\sum_{K_1=0}^{n}\binom{n}{K_1}p^{K_1}q^{n-K_1}\cdot M_\zeta(X,K_1)$$

$$M_\zeta^2(X) = \sum_{K_1=0}^{n} \sum_{K_2=0}^{n} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \left(P^*\{\eta \leqslant K_2\}^{2^{K_1}} - P^*\{\eta \leqslant K_2 - 1\}^{2^{K_1}} \right) \cdot M_\zeta(X, K_2)$$

$$\cdots$$

Виведення формули математичного сподівання кількості блоків з X за F циклів допоможе розрахувати середню кількість блоків після різних стратегій атак і порівняти їх одна з одною.

2.3 Оптимізація атаки

Нехай підчас перебору усіх можливих комбінацій відкриття своїх коммітментів, яких K, перебравши c варіантів з 2^K, знайшов таку комбінацію, яка дасть в результаті атаки на збільшення кількості коммітментів $\tilde{K}$ блоків з n.

Нехай θ_1,...,θ_{2^K-c} - незалежні випадкові величини з біноміальним розподілом (n, p). Тоді ймовірність того, що S з них будуть більше за $\tilde{K}$, а усі інші - менше ніж $\tilde{K}$, буде добутком відповідних ймовірностей, помноженим на кількість усіх можливих перестановок S серед $2^K - c$ варіантів:

$$\binom{2^K - c}{S} \cdot \prod_{t=1}^{S} P\left\{\theta_t > \tilde{K}\right\} \cdot \prod_{t=S+1}^{2^K-c} P\left\{\theta_t \leqslant \tilde{K}\right\} =$$

$$= \binom{2^K - c}{S} \cdot P\left\{\theta_1 > \tilde{K}\right\}^S \cdot P\left\{\theta_1 \leqslant \tilde{K}\right\}^{2^K-c-S} =$$

$$= \binom{2^K - c}{S} \cdot \left(\sum_{k=\tilde{K}+1}^{n} \binom{n}{k} p^k q^{n-k} \right)^S \cdot \left(\sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \right)^{2^K-c-S}$$

Сума наведених імовірностей по усіх значеннях S знаходиться за допомогою біному Ньютона:

$$\sum_{S=0}^{2^K-c} \binom{2^K-c}{S} \cdot \Big(\sum_{k=\tilde{K}+1}^{n} \binom{n}{k} p^k q^{n-k} \Big)^S \cdot \Big(\sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \Big)^{2^K-c-S} =$$

$$= \Big(\sum_{k=\tilde{K}+1}^{n} \binom{n}{k} p^k q^{n-k} + \sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \Big)^{2^K-c} = 1$$

Нехай γ - випадкова подія того, що серед $2^K - c$ незалежних випадкових величин з біноміальним розподілом (n, p) буде щонайменше одна зі значенням більше ніж $\tilde{K}$. Тоді ймовірність γ дорівнює:

$$P\{\gamma\} = \sum_{S=1}^{2^K-c} \binom{2^K-c}{S} \cdot \Big(\sum_{k=\tilde{K}+1}^{n} \binom{n}{k} p^k q^{n-k} \Big)^S \cdot \Big(\sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \Big)^{2^K-c-S} =$$

$$= 1 - \Big(\sum_{k=\tilde{K}+1}^{n} \binom{n}{k} p^k q^{n-k} \Big)^0 \cdot \Big(\sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \Big)^{2^K-c-0} =$$

$$= 1 - \Big(\sum_{k=0}^{\tilde{K}} \binom{n}{k} p^k q^{n-k} \Big)^{2^K-c}$$

Таким чином, знайшовши підчас перебору усіх можливих комбінацій коммітментів варіант, який дає достатньо велику відносно стейку зловмисника кількість блоків, можна порахувати ймовірність знайти хоча б один варіант з більшою кількістю блоків серед решти комбінацій. Якщо ймовірність занадто мала для зловмисника (менша за певну межу), він може зупинити перебір решти варіантів, що дозволить заощаджувати розрахункові потужності, та зменшити розміри переборів на кожному кроці.

2.4 Дослідження функції ймовірності

Для того, щоб мати уявлення про розподіл кількості блоків з n, отриманих в результаті атаки, її потрібно дослідити.

$$P\{\eta = l\} = \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} \cdot \left(\left(\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k} \right)^{2^K} - \left(\sum_{k=0}^{l-1} \binom{n}{k} p^k q^{n-k} \right)^{2^K} \right)$$

$$\frac{dF}{dl} = \frac{F(l+\Delta l) - F(l)}{\Delta l}$$

Оскільки l - кількість блоків з n, отриманих в результаті атаки, то вона може приймати тільки цілі значеняя, отже:

$$\Delta l = 1$$

$$\frac{dF}{dl} = F(l+1) - F(l)$$

$$\frac{dP\{\eta = l\}}{dl} = \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} \cdot \left(\left(\sum_{k=0}^{l+1} \binom{n}{k} p^k q^{n-k} \right)^{2^K} - \left(\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k} \right)^{2^K} - \left(\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k} \right)^{2^K} + \left(\sum_{k=0}^{l-1} \binom{n}{k} p^k q^{n-k} \right)^{2^K} \right) =$$

$$= \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} \cdot \left(\left(\sum_{k=0}^{l+1} \binom{n}{k} p^k q^{n-k} \right)^{2^K} - 2 \left(\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k} \right)^{2^K} + \right.$$

$$\left. + \left(\sum_{k=0}^{l-1} \binom{n}{k} p^k q^{n-k} \right)^{2^K} \right)$$

Позначимо суму до l за a:

$$\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k} = a$$

Тоді:

$$\frac{dP\{\eta = l\}}{dl} = \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} \cdot \left(\left(a + \binom{n}{l+1} p^{l+1} q^{n-l-1} \right)^{2^K} - 2a^{2^K} \right.$$

$$\left. + \left(a - \binom{n}{l} p^l q^{n-l} \right)^{2^K} \right)$$

Нехай:

$$b = \binom{n}{l+1} p^{l+1} q^{n-l-1} = \frac{n!}{(n-l-1)!(l+1)!} \cdot p^{l+1} q^{n-l-1}$$

$$\tilde{b} = \binom{n}{l} p^l q^{n-l} = \frac{n!}{(n-l)!l!} \cdot p^l q^{n-l} =$$

$$= \frac{n!(l+1)}{(n-l-1)!(l+1)!(n-l)} p^{l+1} q^{n-l-1} \frac{q}{p} = \frac{(l+1)q}{(n-l)p} \cdot b$$

$$\Rightarrow \frac{dP\{\eta = l\}}{dl} = \sum_{K=0}^{n} \binom{n}{K} p^K q^{n-K} \cdot \left((a+b)^{2^K} + (a - \frac{(l+1)q}{(n-l)p} \cdot b)^{2^K} - 2a^{2^K} \right)$$

$$\frac{dP\left\{\eta = l\right\}}{dl} < 0$$

Перевірка усієї суми на від'ємність є надскладною задачею, але можливо перевірити достатню умову від'ємності похідної - перевірити від'ємність кожного елементу суми.

$$\forall K \in \mathbb{Z}_{[1;n]} : (a+b)^{2^K} + (a - \frac{(l+1)q}{(n-l)p} \cdot b)^{2^K} - 2a^{2^K} < 0$$

$$\sum_{t=0}^{2^K} \left(\binom{2^K}{t} a^{2^K - t} b^t + \binom{2^K}{t} a^{2^K - t} b^t \left(-\frac{(l+1)q}{(n-l)p} \right)^t \right) - 2a^{2^K} < 0$$

$$\sum_{t=1}^{2^K} \left(\binom{2^K}{t} a^{2^K - t} b^t + \binom{2^K}{t} a^{2^K - t} b^t \left(-\frac{(l+1)q}{(n-l)p} \right)^t \right) < 0$$

$$\sum_{t=1}^{2^K} \binom{2^K}{t} a^{2^K - t} b^t \left(1 + \left(-\frac{(l+1)q}{(n-l)p} \right)^t \right) < 0$$

Перевірка усієї суми на від'ємність є надскладною задачею, але можливо перевірити достатню умову від'ємності похідної - перевірити від'ємність кожного елементу суми.

$$\forall t \in \mathbb{Z}_{[1;2^K]} : 1 + \left(-\frac{(l+1)q}{(n-l)p} \right)^t < 0$$

Оскільки ця умова не виконується для усіх парних t - сума завжди буде додатною, то залишається перевірити цю умову для усіх не парних t.

$$\forall t \in \mathbb{Z}_{[0;2^{K-1}-1]} : 1 + \left(-\frac{(l+1)q}{(n-l)p} \right)^{2t+1} < 0$$

$$-\left(\frac{(l+1)q}{(n-l)p} \right)^{2t+1} < -1$$

$$\left(\frac{(l+1)q}{(n-l)p}\right)^{2t+1} > 1$$

$$\frac{(l+1)q}{(n-l)p} > 1$$

$$(l+1)(1-p) > (n-l)p$$

$$l - lp + lp > np - 1 + p$$

$$l > p(n+1) - 1$$

$$\Rightarrow l \geqslant \lfloor p(n+1) \rfloor$$

Таким чином, була отримана нижня границя значення l, не раніше якого починається спад ймовірності.

2.5 Теоретичні розрахунки математичного сподівання

Теоретично розрахувати значення математичного сподівання кількості блоків після атаки можна за допомогою виведеної раніше формули:

$$M_\zeta^F(X) = \underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_F=0}^{n}}_{\text{F}} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot M_\zeta(X, K_F) \cdot$$

$$\cdot \prod_{t=1}^{F-1} \left(P^* \left\{ \eta \leqslant K_{t+1} \right\}^{2^{K_t}} - P^* \left\{ \eta \leqslant K_{t+1} - 1 \right\}^{2^{K_t}} \right)$$

Розрахунки за допомогою цієї формули є складними через те, що

у формулі присутні дуже великі степені й числа, близькі до одиниці, які сильно змінюють свої значення при піднесенні до великого степеня.

По рівнянню видно, що одні й ті самі ймовірності зустрічаються багато разів. Таким чином, можна розрахувати усі ймовірності завчасно, а потім у розрахунках використовувати вже пораховані значення з таблиці. Ця операція сильно полегшить знаходження теоретичної оцінки математичного сподівання кількості блоків після атаки.

До першої таблиці були занесені:

$$P_1(l,K) = \left(\sum_{k=0}^{l} \binom{N}{k} p^k q^{N-k}\right)^{2^K}$$

Для усіх можливих значень l від 0 до 4096, а K від 0 до 256 при $N = 4096$. Ці ймовірності використовуються на останньому кроці розрахунку математичної ймовірності, де шукаються ймовірності безпосередньої атаки на усі блоки циклу з K зобов'язаннями.

До другої таблиці були занесені:

$$P_2(l,K) = \left(\sum_{k=0}^{l} \binom{n}{k} p^k q^{n-k}\right)^{2^K}$$

Для усіх можливих значень l від 0 до 256, а K від 0 до 256 при $n = 256$. Ці ймовірності використовуються на кожному кроці розрахунку математичної ймовірності, де шукаються ймовірності атаки на блоки з зобов'язаннями при K зобов'язань з попереднього кроку.

Навіть після полегшення обчислень за допомогою завчасно порахованих ймовірностей, розрахунки є занадто складними. Це пов'язано з тим, що потрібна точність, щонайменше 5000 знаків після коми. Таким чином, теоретично, за допомогою формули математичного сподівання, можна порахувати оцінку математичного сподівання кількості блоків після атаки лише до двох етапів. Результати цих обчислень були занесені до таблиць 2.1 та 2.2.

Таблиця 2.1 – Теоретичні оцінки математичного сподівання кількості блоків після атаки (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
0	40.96	81.92	204.8	409.6	614.4
1	45.1559	92.7390	238.7721	482.8285	724.9263
2	47.3918	99.4054	259.3915	524.0429	783.6501

Таблиця 2.2 – Теоретичні оцінки математичного сподівання кількості блоків після атаки (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
0	819.2	1024	1228.8	1638.4	2048
1	964.5753	1201.5541	1435.6793	1894.6285	2339.7314
2	1037.7749	1286.2385	1528.9155	1996.4375	2439.2188

2.6 Експериментальні розрахунки математичного сподівання

Оскільки теоретична оцінка математичного сподівання не дозволила розрахувати математичне сподівання кількості блоків після атаки з більш, ніж двома фазами, варто спробувати отримати експериментальні оцінки кількості блоків атаки з більшою кількістю фаз.

Для розрахунку експериментальної оцінки, потрібно генерувати дуже багато випадкових величин з біноміальним розподілом, й обирати серед них максимальне значення K, після чого 2^K стане кількістю випадкових величин з біноміальним розподілом на наступній фазі, після чого знову обирається максимум серед них.

Дана процедура генерації копіює процедуру перебору усіх можливих комбінацій зобов'язань зловмисником з метою знаходження максимальної кількості блоків на наступній фазі.

Отже ця процедура має подібну реальній атаці складність й не

дозволяє робити розрахунки на недостатньо пристосованих до цього обчислювальних приладах при великій долі зловмисника, так як це призводить до дуже стрімкого зростання значення K скрізь фази, що дуже швидко призводить до неспроможності генерації такої великої кількості випадкових величин з біноміальним розподілом.

Таким чином, експериментальна оцінка математичного сподівання кількості блоків після атаки дозволяє збільшити кількість фаз, відносно теоретичної оцінки, але для дуже малих розмірів долі зловмисника.

Отримані в таблиці 2.3 оцінки математичного сподівання кількості блоків після атаки дають змогу оцінити лише значення для малих часток зловмисника (приблизно 1.5 %), але вже до семи фаз атаки. Завдяки цим результатам, можна оцінити, як змінюється приріст кількості блоків зі збільшенням кількості фаз атаки.

З таблиці видно, що не є ефективним проведення атаки в занадто багато фаз, так як кількість витрачених обчислювальних потужностей не буде відповідати кількості отриманих додаткових блоків.

Відсутність у таблиці великих часток зловмисника робить обчислення більш реалістичними й застосовними на практиці, бо ймовірність того, що атаку вирішить провести учасник з великою часткою є дуже малою.

2.7 Комбіновані розрахунки математичного сподівання

Як один з варіантів оцінки значень математичного сподівання кількості блоків після атаки, можна скомбінувати алгоритм теоретичної й експериментальної оцінки для отримання кращого комбінованого результату.

Замість генерації дуже великої кількості біноміально розподілених випадкових величин й вибору максимального з них, можна генерувати одразу значення максимуму.

Для генерації цієї специфічної випадкової величини, потрібно знати

Таблиця 2.3 – Експериментальні оцінки математичного сподівання кількості блоків після атаки для малих часток зловмисника

Ітерацій\ Доля	0.01	0.0125	0.015	0.175	0.02
0	40.96	51.2	61.44	71.68	81.92
1	45.0529	56.9043	68.7387	80.8694	92.8615
2	47.3417	60.1831	73.0818	86.2195	99.3433
3	48.6059	62.0720	75.7340	89.4059	103.3301
4	49.2056	63.0136	77.1718	91.2794	105.5421
5	49.6245	63.7204	77.9257	92.3956	106.6442
6	49.9314	64.1119	78.5127	92.9456	107.4118
7	50.0260	64.2742	78.6918	93.2691	107.7632

її розподіл. Отримати розподіл цієї випадкової величини допоможуть ймовірності, пораховані при теоретичній оцінці математичного сподівання.

Таким чином, можна представити цю нову випадкову величину як кульку, що випадковим чином кидається на відрізок $[0; 1]$, на якому розміщені проміжки, рівні ймовірності отримати відповідне значення випадкової величини. Випадкова величина буде приймати те значення, до якого відрізка потрапила кулька.

Таке спрощення моделі генерації дозволяє замінити генерацію дуже великої кількості біноміально розподілених випадкових величин й вибору максимуму з них на генерацію всього однієї випадкової величини з рівномірним розподілом.

У таблицях 2.6 і 2.7 були експериментально оцінені значення математичного сподівання кількості блоків після атаки, використовуючи нову, комбіновану формулу генерації випадкових величин із запропонованою оптимізацією. Ця оптимізація дозволила оцінити математичне сподівання для будь-якої долі зловмисника й великої кількості фаз атаки.

Таблиця 2.4 – Комбіновані оцінки математичного сподівання кількості блоків після атаки (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
0	40.96	81.92	204.8	409.6	614.4
1	45.2019	92.7407	238.7423	482.9551	724.8268
2	47.3368	99.405	259.4576	523.9545	783.671
3	48.6449	103.4066	270.4536	544.1322	810.3914
4	49.303	105.5269	275.9929	553.4369	821.7479
5	49.7125	106.7373	278.6058	557.5532	826.4515
6	49.8216	107.3988	279.9188	559.3609	828.3701
7	49.9531	107.7206	280.6026	560.2229	829.1542
8	49.9947	107.9584	280.9069	560.54	829.4544
9	50.0732	108.0492	281.0566	560.7203	829.5585
10	50.1153	108.0801	281.1245	560.7491	829.6473

Таблиця 2.5 – Комбіновані оцінки математичного сподівання кількості блоків після атаки (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
0	819.2	1024	1228.8	1638.4	2048
1	964.4901	1201.593	1435.5385	1894.6117	2339.6497
2	1037.8448	1286.2564	1528.9858	1996.4169	2439.1916
3	1068.8134	1319.5494	1562.7343	2026.6892	2461.0438
4	1081.0757	1331.6302	1573.8382	2034.4656	2464.2959
5	1085.7387	1335.8548	1577.3657	2036.301	2464.7472
6	1087.4517	1337.2867	1578.4267	2036.7371	2464.8731
7	1088.1218	1337.7832	1578.7921	2036.7902	2464.9289
8	1088.3571	1337.9716	1578.8763	2036.7766	2464.9458
9	1088.4656	1338.0081	1578.9252	2036.8464	2464.9367
10	1088.4751	1338.0583	1578.9298	2036.8799	2464.8968

2.8 Оцінка приростів математичного сподівання за різну кількість фаз атаки

Для формулювання стратегії й оцінки ефективності атаки, потрібно розглянути, як змінюється приріст кількості блоків зі збільшення кількості фаз атаки. Результати приросту відносно середньої кількості блоків без застосування атаки представлені у таблицях 2.6 і 2.7.

Таблиця 2.6 – Оцінка приростів математичного сподівання відносно кількості без атаки (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
1	0.1036	0.1321	0.1657	0.1791	0.1797
2	0.1557	0.2134	0.2669	0.2792	0.2755
3	0.1876	0.2623	0.3206	0.3284	0.319
4	0.2037	0.2882	0.3476	0.3512	0.3375
5	0.2137	0.3029	0.3604	0.3612	0.3451
6	0.2163	0.311	0.3668	0.3656	0.3483
7	0.2196	0.3149	0.3701	0.3677	0.3495
8	0.2206	0.3179	0.3716	0.3685	0.35
9	0.2225	0.319	0.3723	0.3689	0.3502
10	0.2235	0.3193	0.3727	0.369	0.3503

Як видно з приведених таблиць, максимально можливе значення зростання спочатку зростає, для малих значень долі зловмисника, а потім знову починає спадати. Максимальний приріст видно для $p = 0.05$, приблизно за 7 фаз з такою долею, зловмисник досягає близько 37% зростання відносно кількості блоків без атаки.

Також, можна побачити, як сильно падає зростання середньої кількості блоків після атаки, це каже про те, що у довгостроковій перспективі, використовування атаки з дуже великою кількістю фаз не є вигідною.

Таблиця 2.7 – Оцінка приростів математичного сподівання відносно кількості без атаки (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
1	0.1774	0.1734	0.1682	0.1564	0.1424
2	0.2669	0.2561	0.2443	0.2185	0.191
3	0.3047	0.2886	0.2718	0.237	0.2017
4	0.3197	0.3004	0.2808	0.2417	0.2033
5	0.3254	0.3045	0.2837	0.2429	0.2035
6	0.3275	0.3059	0.2845	0.2431	0.2036
7	0.3283	0.3064	0.2848	0.2432	0.2036
8	0.3286	0.3066	0.2849	0.2431	0.2036
9	0.3287	0.3066	0.2849	0.2432	0.2036
10	0.3287	0.3067	0.2849	0.2432	0.2036

Для явного порівняння швидкості зростання зі зіблышенням кількості фаз, потрібно розглянути таблиці зростань 2.8 і 2.9.

Таблиця 2.8 – Оцінка приростів математичного сподівання відносно меншої на один кількості фаз (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
1	0.0938	0.1166	0.1421	0.1518	0.1523
2	0.0472	0.0719	0.0868	0.0849	0.0812
3	0.0276	0.0403	0.0424	0.0385	0.0341
4	0.0135	0.0205	0.0205	0.0171	0.014
5	0.0083	0.0115	0.0095	0.0074	0.0057
6	0.0022	0.0062	0.0047	0.0032	0.0023
7	0.0026	0.003	0.0024	0.0015	0.0009

На відміну від таблиць 2.6 і 2.7, прирости зі збільшенням фази (для великої кількості фаз) майже стабільно спадають при збільшенні долі зловмисника, тут вже не видно екстремуму, що знаходиться не в кінці відрізка. Але видно, що максимум не в кінці для менших фаз: для першої

Таблиця 2.9 – Оцінка приростів математичного сподівання відносно меншої на один кількості фаз (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
1	0.1506	0.1478	0.144	0.1352	0.1247
2	0.0761	0.0705	0.0651	0.0537	0.0425
3	0.0298	0.0259	0.0221	0.0152	0.009
4	0.0115	0.0092	0.0071	0.0038	0.0013
5	0.0043	0.0032	0.0022	0.0009	0.0002
6	0.0016	0.0011	0.0007	0.0002	0.0001
7	0.0006	0.0004	0.0002	0.0	0.0

фази він приналежний долі 0.15 і зміщується по таблиці з кожною фазою до мінімального значення долі. З цього можна зробити висновок, що обирати кількість фаз варто, відштовхуючись від долі зловмисника, бо чим більше доля, тим менше приріст кожної нової фази. Наприклад, зловмисник збільшить кількість своїх блоків майже в однакову кількість разів за дві фази зі стейком 0.15, три фази 0.02 чи три фази 0.3. Але все ж таки, більше варто оцінювати саме кількість блоків, що є більш важливою, ніж приріст.

Ці дані також дають зрозуміти, що якщо б зловмисник мав долю 0.1 у п'яти різних Тезос-подібних блокчейнах (якщо б вони існували), то він отримав би значно більше блоків, ніж якщо б мав долю 0.5 у одному такому блокчейні.

Кількість фаз у останніх таблицях була зменшена до семи, так як подальші прирости є вкрай малими, й через наближеність даних, варіюються в певному околі.

Для подальшого дослідження приростів зі збільшенням кількості фаз, варто розглягути відношення різниці приростів, що покаже швидкість падіння кількості додаткових блоків за кожну додаткову фазу у таблицях 2.10 і 2.11.

Як видно з приведених таблиць, за сім фаз атаки, приріст з кожного

Таблиця 2.10 – Оцінка відношення приростів математичного сподівання відносно меншої на один кількості фаз (до 0.2)

Зміна ітерації\ Доля	0.01	0.02	0.05	0.1	0.15
1\2	0.5035	0.6163	0.6106	0.5592	0.5331
2\3	0.5851	0.5602	0.4884	0.4536	0.42
3\4	0.4896	0.5094	0.4833	0.444	0.411
4\5	0.614	0.5594	0.4623	0.435	0.4084
5\6	0.2641	0.5403	0.4978	0.4359	0.4056
6\7	1.2034	0.4835	0.5183	0.4753	0.4078

Таблиця 2.11 – Оцінка відношення приростів математичного сподівання відносно меншої на один кількості фаз (від 0.2)

Зміна ітерації\ Доля	0.2	0.25	0.3	0.4	0.5
1\2	0.505	0.4767	0.4521	0.3974	0.3412
2\3	0.3923	0.3674	0.3391	0.2822	0.2106
3\4	0.3845	0.3537	0.3219	0.253	0.1475
4\5	0.376	0.3465	0.3154	0.2351	0.1386
5\6	0.3658	0.3379	0.3001	0.2374	0.2791
6\7	0.3905	0.3463	0.3442	0.1217	0.4424

наступного кроку приблизно в два рази меньший, за попередній приріст, а після перетину долею зловмисника 0.25, приріст навіть не перетинає грань у 50% від попереднього приросту.

2.9 Оцінка сумарної середньої кількості блоків за атаку

Для фіксації кількості фаз з найбільшою ефективністю й отримання найбільшого можливого середнього прибутку (по кількості блоків), варто розрахувати середню кількість блоків для відповідної числа фаз, усереднену по їх кількості.

Оскільки при фазі атаки на збільшення кількості зобов'язань,

зловмисник фокусується саме на збільшенні кількості зобов'язань і не зважає на звичайні блоки, а отже його прибуток підчас цієї фази буде сумою кількості отриманих зобов'язань і кількістю отриманих без атаки звичайних блоків серед усіх інших.

Теорема 2.4. *Математичне сподівання кількості блоків зловмисника δ підчас багатофазової атаки з поступовим нарощуванням кількості зобов'язань на фазі F (при тому, що F - не остання фаза) дорівнює:*

$$M_\delta^F(N) = \sum_{l=1}^{N} l \cdot \sum_{r=0}^{l} \Bigg(\Bigg(\underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_{F-1}=0}^{n}}_{F\text{-}1} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-1} \left(P^*\{\eta \leqslant K_{t+1}\}^{2^{K_t}} - P^*\{\eta \leqslant K_{t+1}-1\}^{2^{K_t}} \right) \cdot$$

$$\cdot \left(P^*\{\eta \leqslant r\}^{2^{K_{F-1}}} - P^*\{\eta \leqslant r-1\}^{2^{K_{F-1}}} \right) \Bigg) \cdot$$

$$\cdot \left(\binom{N-n}{l-r} p^{l-r} q^{N-n-l+r} \right) \Bigg)$$

Доведення.

Оскільки на фазі атаки на блоки з зобов'язаннями зловмисника не цікавлять ніякі блоки, окрім блоків з зобов'язаннями, то розподіл його блоків серед усіх інших не залежить від атаки й має біноміальний розподіл, як звичайний (без проведення атаки) розподіл пекарських прав серед усіх учасників по їх долям. Але у даному випадку, максимумом для біноміального розподілу буде вже $N - n$, так я серед інших n блоків

зловмисник обирає блоки з зобов'язаннями. Таким чином, вибір блокіцв на фазі був розділений на два випадкові числа: керований вибір блоків з зобов'язаннями серед n блоків і не керований вибір зивчайних блоків з усіх інших $N - n$ блоків циклу.

Оскільки ці випадкові величини є незалежними, то ймовірність отримати рівно l блоків буде дорівнювати ймовірності усіх можливих комбінацій l_1 і l_2, де $l_1 + l_2 = l$, l_1 - кількксість блоків з зобов'язаннями, а l_2 звичайних блоків.

Отримана у теоремі 2.3 ймовірність при $X = n$ буде ймовірністю отримати l_1 блоків з зобов'язаннями за F фаз атаки:

$$P_F\left\{\zeta = l_1\right\}(n) = \underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_{F-1}=0}^{n}}_{\text{F-1}} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-1} \left(P^*\left\{\eta \leqslant K_{t+1}\right\}^{2^{K_t}} - P^*\left\{\eta \leqslant K_{t+1} - 1\right\}^{2^{K_t}} \right)$$

$$\cdot \left(P^*\left\{\eta \leqslant l_1\right\}^{2^{K_{F-1}}} - P^*\left\{\eta \leqslant l_1 - 1\right\}^{2^{K_{F-1}}} \right)$$

Ймовірністю отримати l_2 серед $N - n$ блоків буде ймовірністю біноміального розподілу:

$$P\left\{\zeta = l_2\right\} = \binom{N-n}{l_2} p^{l_2} q^{N-n-l_2}$$

Отже ймовірність того, що ці випадкові величини дадуть в сумі l буде дорівнювати сумі:

$$P_F\left\{\delta = l\right\} = \sum_{r=0}^{l} \left(\left(\underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_{F-1}=0}^{n}}_{\text{F-1}} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot \right. \right.$$

$$\cdot \prod_{t=1}^{F-1} \left(P^* \left\{ \eta \leqslant K_{t+1} \right\}^{2^{K_t}} - P^* \left\{ \eta \leqslant K_{t+1} - 1 \right\}^{2^{K_t}} \right) \cdot$$

$$\cdot \left(P^* \left\{ \eta \leqslant r \right\}^{2^{K_{F-1}}} - P^* \left\{ \eta \leqslant r - 1 \right\}^{2^{K_{F-1}}} \right) \Bigg) \cdot$$

$$\cdot \left(\binom{N-n}{l-r} p^{l-r} q^{N-n-l+r} \right) \Bigg)$$

Таким чином, порахувавши математичне сподівання для випадкової величин з такими ймовірностями теорема буде доведена.

□

Наслідок 2.4. *Математичне сподівання кількості блоків зловмисника δ підчас багатофазової атаки з поступовим нарощуванням кількості зобов'язань на фазі F (при тому, що F - не остання фаза) дорівнює:*

$$M_\delta^F(N) = M_\zeta^F(n) + (N-n)p$$

Доведення. Це є наслідком того, що обидві випадкові величини є нелалежними, а як відомо з властивостей математичного сподівання, математичне сподівання суми незалежних випадкових величин дорівнює сумі їх математичних сподівань.

□

Теорема 2.5. *Нехай δ - випадкова величина, що дорівнює сумарній кількості отриманих блоків в результаті атаки з F фаз. Тоді її математичне сподівання дорівнює:*

$$M_\delta^F(N) = (N-n)(F-1)p + \sum_{S=1}^{F-1} M_\zeta^S(n) + M_\zeta^F(N)$$

Доведення.

Аналогічно з доведенням теореми 2.3 отримуємо ймовірність:

$$P_F\left\{\zeta = l \mid K_1 = \tilde{K}_1,...,K_{F-1} = \tilde{K}_{F-1}\right\}(X) = \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-1} \left(P^*\left\{\eta \leqslant K_{t+1}\right\}^{2^{K_t}} - P^*\left\{\eta \leqslant K_{t+1} - 1\right\}^{2^{K_t}} \right)$$

$$\cdot \left(\left(P\left\{\eta \leqslant l\right\}(N)\right)^{2^{K_{F-1}}} - \left(P\left\{\eta \leqslant l-1\right\}(N)\right)^{2^{K_{F-1}}} \right)$$

Але в даному випадку для математичного сподівання враховується кількість отриманих блоків на кожному етапі атаки:

$$K_2 + K_3 + \ldots + K_{F-1} + l$$

Отже, математичне сподівання сумарної кількості блоків, отриманих виключно атакою (β), дорівнює:

$$M_\beta^F(N) = \underbrace{\sum_{K_1=0}^{n} \cdots \sum_{K_{F-1}=0}^{n}}_{\text{F-1}} \sum_{l=0}^{N} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-2} \left(P^*\left\{\eta \leqslant K_{t+1}\right\}^{2^{K_t}} - P^*\left\{\eta \leqslant K_{t+1} - 1\right\}^{2^{K_t}} \right)$$

$$\cdot \left(\left(P\left\{\eta \leqslant l\right\}(N)\right)^{2^{K_{F-1}}} - \left(P\left\{\eta \leqslant l-1\right\}(N)\right)^{2^{K_{F-1}}} \right) \cdot$$

$$\cdot \left(K_2 + K_3 + \ldots + K_{F-1} + l \right) =$$

$$= M_\zeta^F(N) + \underbrace{\sum_{K_1=0}^{n} \dots \sum_{K_{F-1}=0}^{n}}_{\text{F-1}} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-2} \left(P^* \left\{ \eta \leqslant K_{t+1} \right\}^{2^{K_t}} - P^* \left\{ \eta \leqslant K_{t+1} - 1 \right\}^{2^{K_t}} \right)$$

$$\cdot \Big(K_2 + K_3 + \ldots + K_{F-1} \Big) \cdot$$

$$\cdot \sum_{l=0}^{N} \left(\left(P \left\{ \eta \leqslant l \right\} (N) \right)^{2^{K_{F-1}}} - \left(P \left\{ \eta \leqslant l-1 \right\} (N) \right)^{2^{K_{F-1}}} \right) =$$

Сума ймовірностей по всім можливим значенням дорівнює 1

$$= M_\zeta^F(N) + M_\zeta^{F-1}(n) + \underbrace{\sum_{K_1=0}^{n} \dots \sum_{K_{F-2}=0}^{n}}_{\text{F-2}} \binom{n}{K_1} p^{K_1} q^{n-K_1} \cdot$$

$$\cdot \prod_{t=1}^{F-3} \left(P^* \left\{ \eta \leqslant K_{t+1} \right\}^{2^{K_t}} - P^* \left\{ \eta \leqslant K_{t+1} - 1 \right\}^{2^{K_t}} \right)$$

$$\cdot \Big(K_2 + K_3 + \ldots + K_{F-2} \Big) \cdot$$

$$\cdot \sum_{K_{F-1}=0}^{n} \left(\left(P^* \left\{ \eta \leqslant K_{F-1} \right\} (X) \right)^{2^{K_{F-2}}} - \left(P^* \left\{ \eta \leqslant K_{F-1} - 1 \right\} (X) \right)^{2^{K_{F-2}}} \right)$$

Таким чином, виносячи на кожному кроці з суми $K_2 + K_3 + \ldots + K_{F-1} + l$ крайній елемент, сума з винесеним членом згорнеться по формулі натематичного сподівання кількості блоків за S

фаз атаки, у той час, як для усіх інших елементів суми зникне сума по тому доданку, що був винесений, і через це, сума його ймовірностей перетворюється на 1 для членів суми, що залишилися. Повторивши таку процедуру F разів, буде отримана формула математичного сподівання випадкової величини β:

$$M_{\beta}^{F}(N) = \sum_{S=1}^{F-1} M_{\zeta}^{S}(n) + M_{\zeta}^{F}(N)$$

Використовуючи наслідок 2.4, а саме: додавши математичні сподівання незалежних біноміально розподілених випадкових величин кількості отриманих блоків в результаті не кінцевої фази атаки з усіх блоків цикла без зобов'язань за кожну не кінцеву фазу атаки, яких $F-1$, випливає, що математичне сподівання сумарної кількості усіх блоків за усі фази атаки, це:

$$M_{\delta}^{F}(N) = (N-n)(F-1)p + M_{\beta}^{F}(N)$$

□

Методом генерації великої кількості атак, експериментально були виміряні математичні сподівання кількості усіх блоків після не кінцевої фази атаки у таблицях Б.1 і Б.2.

Як видно з приведених даних, кількість блоків з зобов'язаннями на проміжних фазах атаки поводить себе аналогічно фінальній кількості блоків за атаку - швидкість зростання дуже стрімко падає з кожною новою фазою атаки.

Просумувавши усі значення кількості усіх блоків на проміжних й на останній фазах, були отримані таблиці 2.12 і 2.13 з сумарною середньою кількістю блоків за усю багатофазову атаку.

Але, приведені вище дані складно аналізувати й порівнювати між собою. Для явного порівняння їх між собою, у таблицях Б.3 і Б.4 ці ж самі дані були усереднені по кількості фаз атаки.

Таблиця 2.12 – Сумарна середня кількість блоків за усі фази (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
0	40.96	81.92	204.8	409.6	614.4
1	45.2019	92.7407	238.7423	482.9551	724.8268
2	89.0735	183.3524	470.8296	947.8619	1419.2252
3	132.4567	272.6293	697.5577	1400.5973	2093.8931
4	175.562	361.0531	921.5244	1847.0117	2758.8707
5	218.4566	448.9957	1143.8665	2290.3751	3419.655
6	261.2566	536.6033	1365.5082	2732.3878	4078.6381
7	304.0462	624.1276	1586.7316	3173.7982	4736.891
8	346.8141	711.5481	1807.8022	3614.8828	5394.841
9	389.5404	798.8849	2028.821	4055.9127	6052.6381
10	432.3163	886.3196	2249.704	4496.8725	6710.475

Таблиця 2.13 – Сумарна середня кількість блоків за усі фази (від 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
0	819.2	1024	1228.8	1638.4	2048
1	964.4901	1201.593	1435.5385	1894.6117	2339.6497
2	1884.5927	2343.4092	2795.7984	3680.0106	4535.8393
3	2777.1114	3450.2372	4113.4863	5410.0844	6666.8499
4	3657.1757	4542.1646	5414.3922	7121.9879	8781.2248
5	4532.0829	5628.4552	6709.6427	8829.1112	10893.1491
6	5405.0493	6712.8295	8003.1112	10534.9868	13004.7444
7	6277.272	7796.4884	9295.9967	12240.621	15116.2994
8	7149.1533	8879.9149	10588.7341	13946.1918	17227.7896
9	8020.9927	9963.2064	11881.4057	15651.7249	19339.3144
10	8892.7928	11046.5814	13174.0438	17357.22	21450.8732

Висновки до розділу 2

У даному розділі були отримані формули ймовірностей і математичного сподівання кількості блоків після атаки, які були

використані для отримання числових оцінок різних стратегій атаки.

Ці оцінки були згруповані до таблиць для очного порівняння їх між собою й оцінки ефективності тієї чи іншої атаки.

Дані в таблицях дозволили також побачити швидкість падіння ефективності атаки з ростом кількості фаз атаки.

ВИСНОВКИ

Під час роботи проведено дослідження та аналіз атаки на блокчейн Тезос в цілому й розглянуті отримані в попередній роботі формули, що дозволяють краще описати гріндінг-атаку. Ця робота є продовженням дослідження, зробленого у бакалаврській роботі.

У дослідженні отримані формули ймовірностей і математичного сподівання кількості блоків після проведення гріндінг-атаки. Отримані формули дозволили провести розрахунки оцінок ефективності різних, наведених у роботі, стратегій гріндінг-атаки й порівняти їх між собою.

Результати обчислень показали, що найбільш ефективною стратегією атаки на проміжку часу є атака в одну фазу, яка не використовує збільшення кількості зобов'язанн на попередніх фазах.

Але, якщо ціллю зловмисника є «максимізувати кількість отриманих блоків за один цикл», то зловмисник скоріше за все буде обирати атаку з максимальною кількістю фаз. З кожною новою фазою приріст кількості блоків відносно приросту на попередній фазі буде дуже стрімко падати (приблизно в два рази), але задача максимізації прибутку за один цикл буде виконаною.

Стратегія максимізації кількості блоків за один цикл є малоімовірною, так як зловмисник, міг би заробити значно більше, проводячи однофазову атаку, а настільки велике відхилення кількості отриманих блоків від його математичного сподівання з великою ймовірністю приверне до зловмисника багато непотрібної уваги, у той час, як зростання кількості блоків на 10-20% може залишитись непоміченим.

ПЕРЕЛІК ПОСИЛАНЬ

1. Тулупов, М. М. Побудова та аналіз нової атаки з вибірковим відкриттям коммітментів на блокчейн Тезос : дипломна робота бакалавра : 113 Прикладна математика / Тулупов Матвій Миколайович. Київ, 2021. – 54 с.

2. Tezos — a self-amending crypto-ledger, White paper, L.M Goodman, September 2, 2014

3. Proof-of-stake in Tezos, gitlab

4. Ariel Gabizon Iddo Bentov and Alex Mizrahi. Cryptocurrencies without proof of work. 2014.

5. V. Buterin, “Proof of Stake: How I Learned to Love Weak Subjectivity,” Nov 2014.

6. PROOF-OF-STAKE (POS) | Ethereum

ДОДАТОК А ТЕКСТИ ПРОГРАМ

Алгоритми для обчислення числових даних дослідження були написані мовою Python.

А.1 Підключення бібліотек

```
from scipy.stats import binom
import numpy as np

from decimal import *
#from decimal import Decimal
getcontext().prec = 5000

from tqdm import tqdm

from scipy.stats import uniform
np.set_printoptions(suppress=True)

import random
```

А.2 Клас для обрахунків ймовірностей атаки

```
class Commitment_Attack():

    n = 128 + 1
    N = 4096 + 1

    def __init__(self, p = 0):
        self.p = Decimal(str(p))
```

```
    self.q = Decimal(str(1 - p))



def get_comb_vector(self, max_blocks = 128):
    vector = np.array([0] * (max_blocks + 1), dtype = Decimal)
    for l in range(max_blocks + 1):
        vector[l] = Decimal(str(comb(l, max_blocks)))
    self.comb_vector = np.array(vector)



def get_table(self):
    Table = np.array([[0] * Commitment_Attack.n] *
    Commitment_Attack.n, dtype = Decimal)
    for l in tqdm(range(Commitment_Attack.n)):

        pow_ans = Decimal('0')

        for i in range(l + 1):
            pow_ans += self.comb_vector[i]*



        Table[l][0] = pow_ans
        for K in range(1, Commitment_Attack.n):
            pow_ans = pow_ans ** 2
            Table[l][K] = pow_ans
    self.prob_table = Table



def get_binom_vector(self, b_max):
    Vector = np.array([0] * (b_max + 1), dtype = Decimal)
```

```
        self.get_comb_vector(b_max)
        for i in tqdm(range(b_max + 1)):
            Vector[i] = self.comb_vector[i]*
            (self.p**i)*(self.q**(b_max - i))
        self.binom_vector = Vector


    def get_4096_table(self):

        #self._4096_vector = np.array([0]
        * 4097, dtype = Decimal)

        Table = np.array([[0] * (128 + 1)]
        * 4097, dtype = Decimal)

        ans = Decimal('0')
        for i in tqdm(range(4096 + 1)):

            ans += comb(i,4096)*
            (self.p**i)*(self.q**(4096 - i))

            #self._4096_vector[i] = pow_ans
            pow_ans = ans
            Table[i][0] = pow_ans
            for K in range(1, Commitment_Attack.n):
                pow_ans = pow_ans ** 2
                Table[i][K] = pow_ans

        self._4096_table = Table
```

```
def iteration_real(self, K, iterations):
    ans = Decimal('0')
    if not iterations:
        for i in range(self.N - 1):
            ans += self._4096_table[i][K]
        return self.N - 1 - ans
    else:
        ans += self.prob_table[0][K] *
        self.iteration_real(0, iterations - 1)
        for i in range(1,self.n):
            ans += (self.prob_table[i][K] -
                    self.prob_table[i - 1][K]) *
                    self.iteration_real(i, iterations - 1)
        return ans


def M(self, iterations):
    ans = Decimal('0')

    for i in tqdm(range(self.n)):
        ans += self.comb_vector[i] *
        (self.p ** i) * (self.q ** (self.n - 1 - i)) *
        self.iteration_real(i, iterations - 1)
    return float(ans)


def get_easy_table(self, eps = 10):
    Table = np.array([[0] * Commitment_Attack.n] *
    Commitment_Attack.n, dtype = Decimal)
    for K in tqdm(range(Commitment_Attack.n)):
```

```
            Table[0][K] = self.prob_table[0][K]
            for l in range(1, Commitment_Attack.n):
                number = Decimal(str(self.prob_table[l][K] -
                self.prob_table[l - 1][K]))
                #print(l,K)
                #print(number)
                if number > 10 ** (-eps):
                    Table[l][K] = number
        self.prob_easy_table = np.array(Table, dtype = float)

        print(sum(np.sum(self.prob_easy_table,
        axis = 0)) / Commitment_Attack.n)




def get_easy_4096_vector(self):
    self.easy_4096_vector = np.array(4096 -
    np.sum(self._4096_table[:-1], axis = 0), dtype = float)




def iteration_real_mod(self, K, iterations):
    ans = 0
    if iterations:
        for i in range(self.n):
            if self.prob_easy_table[i][K]:
```

```
                ans += self.prob_easy_table[i][K] *
                self.iteration_real_mod(i, iterations - 1)
        return ans



    else:
        return self.easy_4096_vector[K]


def M_mod(self, iterations, eps = 10):
    if not iterations:
        return 0
    ans = 0
    p_f = float(self.p)
    q_f = float(self.q)

    for i in tqdm(range(self.n - 1, -1, -1)):
        comb_number = float(self.comb_vector[i]) *
        (p_f ** i) * (q_f ** (self.n - 1 - i))
        if comb_number > 10 ** (-eps):
            ans += comb_number *
            self.iteration_real_mod(i, iterations - 1)
    return float(ans)




def iteration_distr(self, K, l_end, iterations, eps):
```

```
    ans = 0
    if iterations:
        for i in range(self.n):
            if self.prob_easy_table[i][K]:
                ans += self.prob_easy_table[i][K] *
                self.iteration_distr(i,
                l_end, iterations - 1, eps)
        return ans

    if not l_end:
        return float(self._4096_table[0][K])
    return float(self._4096_table[l_end][K] -
    self._4096_table[l_end - 1][K])
```

```
def get_distribution(self, iterations, eps = 45):
    self.distr_table = []
    for k_start in tqdm(range(128)):
        Table = np.array([0] * self.N, dtype = Decimal)

        if not iterations:
            return 0
        ans = 0

        if self._4096_table[0][k_start] > 10 ** (-eps):
            Table[i] = self.iteration_distr(k_start,
            i, iterations - 1)
```

```
            for i in range(1,self.N):
                if (self._4096_table[i][k_start] -
                self._4096_table[i - 1][k_start]) > 10 ** (-eps):
                    Table[i] = self.iteration_distr(k_start,
                    i, iterations - 1, eps)
                Table[i] += Table[i - 1]

            self.distr_table.append(Table)
        self.distr_table = np.array(self.distr_table)
```

A.3 Функція для гібридного розрахунку математичного очікування сумарної кількості блоків після атаки

```
def full_N(num_of_cycles, size = 5000):
    iter_vector = [i for i in binom.rvs(128,
    float(a.p), size=size)]
    rand_vector = [Decimal(str(i))/1000000000000
    for i in uniform.rvs(0,1000000000000,size)]

    ans_vector = np.zeros(size)

    for cycles in range(num_of_cycles - 1):
        iter_vector = [len(a.prob_table[:,
        commitments][a.prob_table[:,commitments] < rand_number])
                                for commitments, rand_number
                                in zip(iter_vector, rand_vector)]

        print(cycles, np.mean(iter_vector))
        ans_vector = ans_vector + iter_vector
```

```
        rand_vector = [Decimal(str(i))/1000000000000
        for i in uniform.rvs(0,1000000000000,size)]



    N_vector = [len(a._4096_table[:,
    commitments][a._4096_table[:,commitments] < rand_number])
                      for commitments, rand_number
                      in zip(iter_vector, rand_vector)]

    ans_vector = ans_vector + N_vector



    print((np.mean(ans_vector) + (4096 - 128) *
    float(a.p) * (num_of_cycles - 1)) / num_of_cycles)
    print(np.mean(N_vector))
```

A.4 Функція для гібридного розрахунку математичного очікування кількості блоків після атаки

```
def for_N(size = 30000):
    ans_arr = []
    for cycles in range(1,11):
        iter_vector = [i for i in binom.rvs(128,
        float(a.p), size=size)]

        rand_vector = [Decimal(str(i))/1000000000000
        for i in uniform.rvs(0,1000000000000,size)]

        if cycles - 1 > 0:
```

```
            for c in range(cycles - 1):

                iter_vector = [len(a.prob_table[:,
                commitments][a.prob_table[:,
                commitments] < rand_number])
                                for commitments,
                                rand_number
                                in zip(iter_vector, rand_vector)]


                rand_vector = [Decimal(str(i))/1000000000000
                for i in uniform.rvs(0,1000000000000,size)]

        ans_vector = [len(a._4096_table[:,
        commitments][a._4096_table[:,
        commitments] < rand_number])
                        for commitments, rand_number
                        in zip(iter_vector, rand_vector)]



        print(cycles, "cycles", np.mean(ans_vector))
        ans_arr.append(np.mean(ans_vector))
    return ans_arr
```

A.5 Функція для гібридного розрахунку математичного очікування кількості блоків після не кінцевої фази атаки

```
def for_n(size = 100000):
    ans_arr = []
```

```
for cycles in range(0,11):
    iter_vector = [i for i in binom.rvs(128,
    float(a.p), size=size)]
    rand_vector = [Decimal(str(i))/1000000000000
    for i in uniform.rvs(0,1000000000000,size)]
    for c in range(cycles):
        iter_vector = [len(a.prob_table[:,
        commitments][a.prob_table[:,
        commitments] < rand_number])
                        for commitments, rand_number
                        in zip(iter_vector, rand_vector)]
        rand_vector = [Decimal(str(i))/1000000000000
        for i in uniform.rvs(0,1000000000000,size)]

    ans_vector = [len(a.prob_table[:,
    commitments][a.prob_table[:,
    commitments] < rand_number])
                        for commitments, rand_number
                        in zip(iter_vector, rand_vector)]
    probs = np.unique(ans_vector,return_counts=1)
    prob_dict = {i:j for i,j
    in zip(probs[0],probs[1]/100000)}
    ans = Decimal(str(0))
    for l in range(4096 - 128):
        l_ans = Decimal(str(0))
        for r in range(l):
            if r in prob_dict:
                l_ans += a.binom_vector[l - r] *
                Decimal(str(prob_dict[r]))
        ans += l * l_ans
    print(cycles, np.round(float(ans),4))
```

```
        ans_arr.append(float(ans))
    return ans_arr
```

A.6 Функція для виведення таблиць у форматі Latex

```
for num, i in enumerate(array):
    print('\hline')
    print(num + 2, end = ' ')
    for j in i:
        print('&',np.round(j,4), end = ' ')

    print('\\\\', end = '\n')
```

A.7 Функції емпіричного розрахунку математичного очікування кількості блоків після атаки для малих долей зловмисника

```
def first_iteration(iterations, size = 1000):
    for i in 2**np.array(binom.rvs(128,
    p, size=size), dtype=object):
        iteration_emp(i, iterations - 1)

def iteration_emp(prev, iter_number):
    if iter_number > 0
    iteration_emp(2**np.max(np.array(binom.rvs(128,
    p, size=prev), dtype=object)), iter_number - 1)
    else:
        number = np.max(binom.rvs(4096, p, size=prev))
        if number in dict:
            dict[number] += 1
```

```
        else:
            dict[number] = 1
```

A.8 Виведення і порівняння емпіричного математичного очікування після атаки й без атаки

```
p = 0.02
dict = {}
first_iteration(number_of_iterations,10000)

ans = 0
for i in dict:
    ans += i * dict[i] / (10000)
print(ans)
print(p*4096)
```

ДОДАТОК Б ВЕЛИКІ РИСУНКИ ТА ТАБЛИЦІ

Таблиця Б.1 – Кількість блоків за фазу атаки на зобов'язання (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
1	41.7096	83.9504	211.3643	423.7822	635.6302
2	42.16	85.3484	215.831	432.6607	647.8961
3	42.3989	86.2076	218.3678	437.1386	653.6111
4	42.5386	86.7097	219.6654	439.2375	656.049
5	42.6244	86.994	220.3102	440.1795	657.0553
6	42.6631	87.144	220.6185	440.5965	657.4738
7	42.6853	87.2268	220.7707	440.7768	657.6422
8	42.7029	87.2686	220.8384	440.8609	657.7076
9	42.705	87.2949	220.8766	440.8893	657.7377

Таблиця Б.2 – Кількість блоків за фазу атаки на зобов'язання (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
1	846.7878	1057.197	1266.8309	1683.569	2096.6132
2	861.5219	1073.4982	1283.8727	1699.8065	2109.2086
3	867.7938	1079.8369	1289.8369	1704.1757	2111.1398
4	870.262	1082.1013	1291.7604	1705.261	2111.4265
5	871.2079	1082.8862	1292.3654	1705.4807	2111.4947
6	871.556	1083.162	1292.5597	1705.528	2111.5111
7	871.6882	1083.2526	1292.6168	1705.5363	2111.5156
8	871.7364	1083.2813	1292.6394	1705.5391	2111.5156
9	871.7579	1083.3016	1292.6426	1705.5386	2111.5152

Таблиця Б.3 – Усереднене по кількості фаз математичне очікування кількості блоків за усі фази (до 0.2)

Ітерацій\ Доля	0.01	0.02	0.05	0.1	0.15
0	40.96	81.92	204.8	409.6	614.4
1	45.2019	92.7407	238.7423	482.9551	724.8268
2	44.5367	91.6762	235.4148	473.9309	709.6126
3	44.1522	90.8764	232.5192	466.8658	697.9644
4	43.8905	90.2633	230.3811	461.7529	689.7177
5	43.6913	89.7991	228.7733	458.075	683.931
6	43.5428	89.4339	227.5847	455.398	679.773
7	43.4352	89.1611	226.6759	453.3997	676.6987
8	43.3518	88.9435	225.9753	451.8604	674.3551
9	43.2823	88.765	225.4246	450.657	672.5153
10	43.2316	88.632	224.9704	449.6872	671.0475

Таблиця Б.4 – Усереднене по кількості фаз математичне очікування кількості блоків за усі фази (від 0.2)

Ітерацій\ Доля	0.2	0.25	0.3	0.4	0.5
0	819.2	1024	1228.8	1638.4	2048
1	964.4901	1201.593	1435.5385	1894.6117	2339.6497
2	942.2963	1171.7046	1397.8992	1840.0053	2267.9196
3	925.7038	1150.0791	1371.1621	1803.3615	2222.2833
4	914.2939	1135.5412	1353.598	1780.497	2195.3062
5	906.4166	1125.691	1341.9285	1765.8222	2178.6298
6	900.8416	1118.8049	1333.8519	1755.8311	2167.4574
7	896.7531	1113.7841	1327.9995	1748.6602	2159.4713
8	893.6442	1109.9894	1323.5918	1743.274	2153.4737
9	891.2214	1107.0229	1320.1562	1739.0805	2148.8127
10	889.2793	1104.6581	1317.4044	1735.722	2145.0873