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

«На правах рукопису»
УДК 004.056.55

«До захисту допущено»
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2023 р.

Дипломна робота

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

зі спеціальності: 113 Прикладна математика
на тему: **«Порівняльний аналіз схем доказів з нульовими знаннями та блокчейнів як ядер побудови криптографічних протоколів довільної складності»**

Виконав:
студент IV курсу, групи ФІ-94
Куценко Андрій Ігорович _____

Керівник:
професор кафедри, доктор технічних наук, старший науковий співробітник
Кудін Антон Михайлович _____

Рецензент:
К.т.н., доцент, заступник начальника управління безпеки інформації департаменту безпеки Національного банку України
Проскуровський Роман Васильович _____

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

Студент _____

Київ — 2023

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

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

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2023 р.

ЗАВДАННЯ
на дипломну роботу

Студент: Куценко Андрій Ігорович

1. Тема роботи: «*Порівняльний аналіз схем доказів з нульовими знаннями та блокчейнів як ядер побудови криптографічних протоколів довільної складності*», науковий керівник роботи: професор кафедри, доктор технічних наук, старший науковий співробітник Кудін Антон Михайлович,

затверджені наказом по університету №__ від «__» _____ 2023 р.

2. Термін подання студентом роботи: «__» _____ 2023 р.

3. Об'єкт дослідження: інформаційні процеси в системах криптографічного захисту.

4. Предмет дослідження: протоколи інтерактивних і неінтерактивних доказів з нульовими знаннями та протоколи з блокчейном.

5. Перелік завдань: 1. Аналіз існуючих інтерактивних та неінтерактивних протоколів з нульовими знаннями. 2. Аналіз техніки евристичного методу Фіата-Шаміра перетворення інтерактивних протоколів з нульовими знаннями на неінтерактивні на прикладі протоколу знання дискретного логарифму. 3. Аналіз недоліків та обмежень застосування евристики Фіата-Шаміра, визначення вимог до

хеш-функцій, які можуть застосовуватись. 4. Аналіз можливостей та розробка модернізованої схеми заміни окремих елементів неінтерактивних протоколів з нульовими знаннями блокчейном PoS-типу.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: презентація доповіді.

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

8. Дата видачі завдання: 10 вересня 2022 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	01-15 вересня 2022 р.	Виконано
2	Огляд опублікованих джерел за тематикою дослідження	Вересень-жовтень 2022 р.	Виконано
3	Аналіз літератури і розробка теоретичного матеріалу для оглядової частини	Листопад-грудень 2022 р.	Виконано
4	Оформлення і захист оглядової частини	Грудень-січень 2022-2023 р.	Виконано
5	Пошук і аналіз практичних рішень проблем за тематикою дослідження та практична розробка алгоритмів	Січень-квітень 2023 р.	Виконано
6	Виконання практики і проведення експериментальних досліджень	17 квітня - 21 травня 2023 р.	Виконано
7	Оформлення та захист дипломної роботи	Травень-червень 2023 р.	Виконано

Студент _____ Андрій Куценко

Керівник _____ Антон Кудін

РЕФЕРАТ

Кваліфікаційна робота містить: 53 стор., 19 джерел.

Ми розглядали проблематику аксіоматичної побудови стійких криптографічних протоколів довільної складності, вона пов'язана із вибором базових криптографічних блоків, з яких можна побудувати будь-який алгоритм. Досліджували певні інформаційні процеси в системах криптографічного захисту. Застосовували протоколи інтерактивних і неінтерактивних доказів з нульовими знаннями та протоколи з блокчейном.

Основні результати, одержані в ході дослідження, полягають в тому, що при використанні деяких конкретних хеш-функцій схема Фіата-Шаміра стає незастосовною та при накладенні певних умов на блокчейн, ми можемо ним замінити хеш-функцію в протоколі.

КЛЮЧОВІ СЛОВА: СИМЕТРИЧНА КРИПТОГРАФІЯ, АСИМЕТРИЧНА КРИПТОГРАФІЯ, БЛОКЧЕЙН, ІНТЕРАКТИВНІ ДОКАЗИ, НЕІНТЕРАКТИВНІ ДОКАЗИ, НУЛЬОВЕ ЗНАННЯ, ХЕШ-ФУНКЦІЯ, СХЕМА ФІАТА-ШАМІРА

ABSTRACT

The qualification work contains: 53 pages, 19 sources.

We have explored the issues related to the axiomatization of secure cryptographic protocols of arbitrary complexity. It involves the choice of basic cryptographic blocks that can be used to construct any algorithm. We have investigated certain information processes in cryptographic security systems. We have applied protocols of interactive and non-interactive zero-knowledge proofs, as well as protocols with a blockchain.

The main results obtained during the research are as follows: when using certain specific hash functions, the Fiat-Shamir scheme becomes inapplicable, and by imposing certain conditions on the blockchain, we can replace the hash function in the protocol with it.

KEYWORDS: SYMMETRIC CRYPTOGRAPHY, ASYMMETRIC CRYPTOGRAPHY, BLOCKCHAIN, INTERACTIVE PROOFS, NON-INTERACTIVE PROOFS, ZERO KNOWLEDGE, HASH FUNCTION, FIAT-SHAMIR SCHEME

ЗМІСТ

ВСТУП

Актуальність дослідження. Актуальність даного дослідження полягає у тому, що досі існує проблема аксіоматичної побудови стійких криптографічних протоколів довільної складності, вона пов'язана із вибором базових криптографічних блоків, з яких можна побудувати будь-який алгоритм. Сьогодні, в якості примітивних блоків, чи інакше – протоколів, разом з традиційними протоколами інтерактивних і неінтерактивних доказів з нульовими знаннями, розглядають блокчейн.

Метою дослідження є побудова стійкого криптографічного протоколу будь-якої складності з примітивних криптографічних блоків, основними з яких є протоколи інтерактивних і неінтерактивних доказів та блокчейн. Для досягнення мети необхідно розв'язати **задачу дослідження**, яка полягає у аналізі перетворення інтерактивного протоколу з нульовими знаннями в неінтерактивний протокол з нульовими знаннями за допомогою евристики Фіата-Шаміра та можливостей заміни окремих елементів протоколу блокчейном. Для розв'язання задачі необхідно вирішити такі завдання:

1) Аналіз існуючих інтерактивних та неінтерактивних протоколів з нульовими знаннями;

2) Аналіз техніки евристичного методу Фіата-Шаміра перетворення інтерактивних протоколів з нульовими знаннями на неінтерактивні на прикладі протоколу знання дискретного логарифму;

3) Аналіз недоліків та обмежень застосування евристики Фіата-Шаміра, визначення вимог до хеш-функцій, які можуть застосовуватись;

4) Аналіз можливостей та розробка модернізованої схеми заміни окремих елементів неінтерактивних протоколів з нульовими знаннями блокчейном PoS-типу.

Об'єктом дослідження є інформаційні процеси в системах криптографічного захисту.

Предметом дослідження є протоколи інтерактивних і неінтерактивних доказів з нульовими знаннями та протоколи з блокчейном.

При розв'язанні поставлених завдань використовувались такі *методи дослідження*: методи теоретичного аналізу, розробки та аналізу алгоритмів, лінійної та абстрактної алгебри, математичного аналізу, теорії імовірностей, математичної статистики, комбінаторного аналізу, теорії кодування, теорії складності алгоритмів.

Наукова новизна отриманих результатів полягає в тому, що, на сьогодні, немає робіт які б досліджували конкретні хеш-функції, коли, при яких умовах на хеш-функцію, по-перше, схема Фіата-Шаміра стає нестійкою, тобто неможливо її застосувати і, по-друге, при яких умовах на блокчейн ми можемо замінювати хеш-функцію блокчейном з відповідними якостями.

Практичне значення результатів полягає в тому, що на базі блокчейнів, зараз, можуть відбудовуватись криптографічні протоколи які будуть набагато більш компактними, більш стійкими, ніж протоколи на хеш-функції. Ми, по-іншому, застосовуємо схему Фіата-Шаміра на конкретних об'єктах.

1 ПОНЯТТЯ ДОКАЗУ З НУЛЬОВИМ РОЗГОЛОШЕННЯМ ТА БЛОКЧЕЙНУ ЯК ОСНОВ ДЛЯ ПОБУДОВИ КРИПТОГРАФІЧНИХ ПРОТОКОЛІВ

У цій частині піде мова про основні поняття з теорії систем доказів з нульовими знаннями та блокчейну, моделі, ідеї та принципи побудови протоколів на їх основі.

1.1 Загальні поняття

Криптологія – це математична наука про шифри, і, до речі, її етимологією є грецька (*kriptós logia*), що означає «вивчення таємниць». Криптологія – це термін для криптографії та криптоаналізу двох об'єктів. Ці об'єкти пов'язані між собою так само, як гра кота з мишею; поки криптографія – це мистецтво побудови алгоритмів та протоколів; криптоаналіз приймає суперечливу точку зору криптології при руйнуванні чи аналізі безпеки криптографічних конструкцій.

Поняття *нульове розголошення* було вперше запропоноване у 1980-х фахівцями: MIT Шафі Голдвассером, Сільвіо Мікалі та Чарльзом Ракоффом (англ. $Shafi\ Goldwasser$, $Silvio\ Micali$ та $Charles\ Rackoff$).

Ці дослідники працювали над проблемою, яка відноситься до інтерактивних систем доказу – теоретичних систем, в яких перший учасник, назвемо його «Випробовувач» (англ. $Prover$) обмінюється повідомленнями з другим учасником «Контролер» (англ. $Verifier$), щоб переконати контролера, що деяке математичне твердження вірне.

Чи можливо переконати когось у тому, що Ви вмієте розв'язувати задачу, не повідомляючи способу її розв'язку, або в тому, що Ви знаєте розв'язок задачі, не розкриваючи його? Ці дві, на перший погляд, взаємовиключні вимоги, задовольняють протоколи доведення з нульовим

розголошенням.

Докази з нульовим розголошенням є захоплюючими і надзвичайно корисними конструкціями. Їхня захоплююча природа обумовлена їх, здавалося б, суперечливим визначенням; докази з нульовим розголошенням переконливі і водночас не дають нічого, крім достовірності твердження, що доводиться. Їх застосовність у сфері криптографії велика; вони зазвичай використовуються, щоб змусити зловмисників поводитися відповідно до заздалегідь визначеного протоколу.

Крім безпосереднього застосування в криптографії, докази з нульовим розголошенням служать гарною відправною точкою для вивчення різних проблем, що стосуються криптографічних протоколів (наприклад, «збереження безпеки при різних формах складу протоколу» та «використання програми супротивника в рамках доказу безпеки») [1, 15, 17, 18].

Теорія доведення з нульовим розголошенням достатньо точно відображає вимоги, які повинні задовольняти протоколи аутентифікації, тому вона насамперед призначена для дослідження цих протоколів. Проте виникають проблеми з ефективністю: задовгий таємний ключ, значна кількість раундів, неприйнятно велика комунікаційна складність протоколу та дуже значна обчислювальна складність, тому при розробці протоколів аутентифікації розробники основну увагу приділяють їх ефективності. Крім того, поняття нульового розголошення дає змогу формалізувати інтуїтивне уявлення протоколу, під час роботи якого не повинен відбуватися витік секретної інформації, тому воно стало корисним і для багатьох інших типів криптографічних протоколів.

Проблема аксіоматичної побудови стійких криптографічних протоколів тісно пов'язана із вибором базових криптографічних блоків, з яких можна побудувати криптографічний протокол довільної складності. Будемо називати такі блоки примітивними криптографічними протоколами. Поряд із традиційним вибором в якості примітивних

протоколів розподілу секрету (англ. secret sharing) та протоколів неінтерактивних доказів (англ. non-interactive) сьогодні як примітивний криптографічний протокол розглядають блокчейн.

Світ блокчейна дуже сильно фрагментований – і велика кількість можливостей імплементації, технічна складність теми, а також медійний ажіотаж навколо індустрії дуже заважають відокремити найголовніше.

Блокчейн — одна з найбільш трендових технологій сьогодення, яка дає змогу відкрити дані й продемонструвати, що вони не зазнали змін, не розкриваючи при цьому персональні дані користувачів. Через прозорість і незалежність від урядів та корпорацій його називають проривом XXI ст.

Блокчейн — це надійний спосіб зберігання даних про угоди, контракти, транзакції, про все, що необхідно записати і перевірити. Сьогодні блокчейн проник практично в усі сфери життєдіяльності, готовий змінити фінансову систему держави і в декілька разів спростити роботу середнього і великого бізнесу. Блокчейн — не таємна технологія, знаходиться у вільному доступі, причому існує величезна кількість статей про те, як він влаштований і за яким принципом працює [2, 12]. Сьогодні блокчейн перестає асоціюватися з біткоіном і стає самостійною технологією, яка є основою нових додатків і систем. До того ж, якщо раніше про блокчейн говорили, як про сховище даних, то тепер його можливості стають набагато ширше, тому що він також може виконувати програми. Деякі блокчейни дозволяють кожному факту містити міні-програму. В основі блокчейну лише математика, але ця властивість не обмежує сферу реалізації даної технології.

1.2 Основні відомості про нульове розголошення

Грубо кажучи, докази з нульовим розголошенням – це докази, які нічого не дають, крім достовірності твердження. Тобто перевіряючий, отримавши такий доказ, лише переконується у справедливості твердження. Це формулюється так: все, що можна вирахувати з

доведення з нульовим розголошенням, також можна вирахувати з самого (дійсного) твердження (за допомогою так званого симулятора).

Сучасна криптографія займається побудовою ефективних схем, котрим неможливо порушити функцію безпеки. Така сама проблема лежить в основі основних визначень нульового знання [3, 11]. Отже, спочатку нам потрібне поняття ефективних обчислень, і навіть поняття неприпустимих. Обчислення законних користувачів схеми мають бути ефективними, тоді як порушення функцій безпеки (через зловмисника) має бути неможливим.

Ефективні обчислення зазвичай моделюються як обчислення з поліноміальним часом у параметрі безпеки. Поліном, що обмежує час виконання стратегії легітимного користувача, є фіксованим та, як правило, явним (і невеликим). Тут (тобто коли йдеться про складність легітимних користувачів) ми знаходимося в тій самій ситуації, що й у будь-якому алгоритмічному налаштуванні [4, 13].

Інша справа, коли йдеться про наші припущення щодо обчислювальних ресурсів супротивника. Звичайний підхід полягає в тому, щоб постулювати, що останні також мають поліноміальний час, коли поліном не вказано апріорі. Іншими словами, противник обмежений класом ефективних обчислень, і все, що виходить за його межі вважається неможливим.

Насправді, щоб спростити наш виклад, ми часто вважатимемо нездійсненними будь-які обчислення, які можуть бути виконані з допомогою (можливо, неоднорідного) сімейства схем поліноміального розміру. Для простоти розглянемо сімейства схем Cn, де для деяких поліномів p і q кожна Cn має рівно $p(n)$ вхідних бітів і має розмір не більше за $q(n)$.

Рандомізовані обчислення грають центральну роль у визначенні нульового розголошення (а також у криптографії в цілому) [3, 14]. Тобто, ми дозволяємо легітимним користувачам використовувати рандомізовані обчислення, а також розглядаємо зловмисників, які використовують

рандомізовані обчислення.

Це піднімає проблему ймовірності успіху: зазвичай ми вимагаємо, щоб законні користувачі досягли успіху (у досягненні своїх законних цілей) з ймовірністю 1 (або зневажливо близької до неї), тоді як зловмисники досягли успіху (у порушенні функцій безпеки) із знехтовно малою ймовірністю. Таким чином, поняття знехтовно малої ймовірності відіграє важливу роль у нашому викладі.

Одна особливість, що потрібна від визначення незначної ймовірності, полягає в тому, щоб дати надійне поняття рідкості: рідкісна подія має відбуватися рідко, навіть якщо ми повторюємо експеримент допустиму кількість разів. Так само ми вважаємо, що дві події відбуваються «так само часто», якщо абсолютна різниця між їхніми відповідними ймовірностями виникнення дуже мала.

Найпривабливіша особливість доказу з нульовим розголошенням полягає в його здавалося б суперечливій унікальній природі: доказуючий може довести правильність твердження перевіряючому без витоку будь-якої додаткової інформації [5]. Він може змусити зловмисних учасників криптографічного протоколу виконати певні кроки для забезпечення безпеки протоколу. Таким чином він має широку перспективу застосування. Говорячи образно, передбачається, що верифікатор, який отримує доказ твердження з нульовим розголошенням, має бути повідомлений Богом, що воно є істинним. Основні характеристики системи доказу з нульовим розголошенням включають повноту, коректність та нульове знання.

Повнота. Якщо твердження правильне, верифікатор «завжди» прийме його.

Коректність. Якщо твердження неправильне, то верифікатор «завжди» відхиляє.

Нульове знання. Жоден (зловмисний) верифікатор не може отримати жодної додаткової інформації від процедури доказу, крім правильності твердження.

1.3 Системи інтерактивних та неінтерактивних доведень з нульовим розголошенням

Перш ніж визначати інтерактивні докази з нульовим розголошенням, ми маємо визначити докази. Стандартне поняття статичних (тобто неінтерактивних) доказів не підходить, тому що статичні докази з нульовим розголошенням існують лише для наборів, які легко вирішити [3].

Тобто тут доказ є (багатораундовим) рандомізованим протоколом для двох сторін, званих верифікатором та доказуючим, у якому доказуючий хоче переконати перевіряючого у достовірності даного твердження. Такий інтерактивний доказ повинен дозволяти доводячому переконати перевіряючого у достовірності будь-якого справжнього твердження, тоді як жодна стратегія доказувача не може обдурити перевіряючого, змусивши його прийняти помилкові твердження. Наведені вище умови повноти і коректності повинні виконуватися з високою ймовірністю (тобто допускається невелика ймовірність помилки).

Таким чином, «доказ» у цьому контексті – це не фіксований та статичний об'єкт, а рандомізований та динамічний (тобто інтерактивний) процес, у якому верифікатор взаємодіє з доказуючим. Інтуїтивно можна уявити собі цю взаємодію як таку, що складається з «підступних» питань, що задаються перевіряючим, на які доказуючий повинен відповісти «переконливо».

Простіше кажучи, інтерактивний доказ – це гра між обчислювально обмеженим перевіряючим та обчислювально необмеженим доказуючим, мета якої — переконати перевіряючого у справедливості певного твердження.

Потрібно, щоб якщо твердження виконується, верифікатор завжди приймав його (тобто при взаємодії з відповідною стратегією перевіряючого). З іншого боку, якщо твердження є хибним, то

верифікатор повинен відхилити його з «помітно» ймовірністю, незалежно від того, яку стратегію використовує той, хто доводить.

Нехай $\{0,1\}^n$ – множина n-бітних рядків, а $\{0,1\}^*$ – множина всіх рядків. Два ймовірнісних ансамблі називаються обчислювально нерозрізнюваними (позначається $\approx_c$), якщо жодна ймовірнісна машина Тьюрінга за поліноміальний час не може розрізнити їх із знехтовно малою ймовірністю. Два ймовірнісних ансамблі називаються статистично нерозрізнюваними, або статистично близькими (позначається $\approx_s$), якщо їхня статистична відстань незначна [5].

Твердження 1.1. *Система інтерактивного доведення з нульовим розголошенням.*

Для мови $L \subseteq \{0,1\}^*$ *і пари інтерактивних машин Тьюрінга* (P,V)*, в яких* P *має необмежені обчислювальні можливості, а* V *є ймовірнісною поліноміальною за часом,* (P,V) *називають системою інтерактивного доведення з нульовим розголошенням на мові* L*, якщо виконуються наступні три умови:*

Повнота*:* $\forall x \in L$ *і полінома* $p(\cdot)$*,*

$$Pr[(P,V)(x) = 1] \geqslant 1 - \frac{1}{p(|x|)}.$$

Коректність*:* $\forall x \notin L$ *і будь-якої інтерактивної машини Тьюрінга* P' *та полінома* $p(\cdot)$*,*

$$Pr[(P',V)(x) = 1] < 1 - \frac{1}{p(|x|)}.$$

Нульове знання*: для будь-якої ймовірнісної поліноміальної машини Тьюрінга* V^**, існує ймовірнісний поліноміальний алгоритм* M^* *такий, що* $\forall x \in L$*,*

$$(P,V^*)(x) \approx_c M^*(x).$$

P *є доказуючим, а* V *– верифікатором.*

Інтуїтивно кажучи, повнота відображає правильність системи, а це означає, що для допустимих вхідних даних $x \in L$, доказуючий завжди може успішно завершити доказ, який приймає перевіряючий. Коректність визначається проти зловмисного доказувача, що означає, що для невірних вхідних даних $x \notin L$, жоден доказуючий P' не може побудувати дійсну систему доказів, яку приймає верифікатор. У той час як для верифікатора нульове знання означає, що зловмисний верифікатор не може отримати додаткові знання з процесу взаємодії [5].

Неінтерактивна система доказу з нульовим розголошенням містить лише повідомлення, що надсилається доказуючим перевіряючому, що може бути краще використано при побудові криптографічних протоколів. Після цього послідовно почалися дослідження теорії та додатків системи неінтерактивних доказів з нульовим розголошенням, включаючи доказ проблем NP та неінтерактивне статистичне (досконале) нульове знання, а також застосування неінтерактивного доказу до схеми безпечного шифрування CCA, анонімна автентифікація та побудова групових та кільцевих підписів.

Твердження 1.2. *Система неінтерактивного доведення з нульовим розголошенням.*

Для пари ймовірнісних машин Тьюрінга (P,V)*, в яких* P *– є ймовірнісною поліноміальною за часом, а* V *– детермінованою поліноміальною за часом,* (P,V) *називається неінтерактивною системою доказу з нульовим розголошенням для мови* L*, якщо виконуються наступні умови:*

Повнота*:* $\forall x \in L$ *і полінома* $p(\cdot)$*,*

$$Pr[V(x,R,P(x,R)) = 1] \geqslant 1 - \frac{1}{p(|x|)}.$$

Коректність*:* $\forall x \notin L$ *і будь-якої інтерактивної машини Тьюрінга* P' *та полінома* $p(\cdot)$*,*

$$Pr[V(x,R,P'(x,R)) = 1] < \frac{1}{P(|x|)}.$$

Нульове знання*:* $\forall x \in L$ *існує ймовірнісний поліноміальний алгоритм* M *такий, що*

$$V(x) = (x,R \in \{0,1\}^{c(|x|)},P(x,R)) \approx_c M(x)_{x\in L}.$$

Нерозрізнюваність свідка – найслабше поняття нульового знання, але його достатньо для забезпечення безпеки криптографічного протоколу в деяких додатках. Варто зазначити, що нерозрізнюваність свідків закривається при паралельній композиції.

Твердження 1.3. *Неорозрізнюваність свідка*

Нехай L *буде* NP*-мовою, а* (P,V) *буде системою інтерактивного доказу для* L*, нехай* R_L *буде відношенням-свідком для* L*, і* $z \in \{0,1\}^*$ *буде додатковим виходом для* V*.* (P,V) *називається свідком, нерозрізнюваним для* R_L*, якщо для будь-якої ймовірнісної поліноміальної інтерактивної машини Тьюринга* V^* *і будь-яких* ω_1*,* $\omega_2 \in R_L(x)$ *наступні ймовірнісні ансамблі обчислювально нерозрізнювані:*

$$\{(P(\omega_1), V^*(z))(x)\}_{x\in L, z\in\{0,1\}^*} \approx_c \{P(\omega_2), V^*(z))(x)\}_{x\in L, z\in\{0,1\}^*}.$$

1.4 Побудова протоколів з нульовим розголошенням

Природне питання щодо доказів із нульовим розголошенням (і аргументів) у тому, чи зберігається умова з нульовим розголошенням при різних операціях композиції. У літературі розглядаються три типи операцій композиції: послідовна композиція, паралельна композиція та одночасна композиція (англ. $sequential, parallel, concurrent$). Зазначимо, що збереження нульового розголошення не тільки цікаве саме собою, а й «проливає світло» на збереження безпеки загальних протоколів за цих

форм композиції [3].

Підкреслимо, що коли ми говоримо про склад протоколів (або систем доказів), ми маємо на увазі, що чесні користувачі повинні дотримуватись запропонованої програми (зазначеної в описі протоколу, що відноситься до одноразового виконання. Тобто дії чесних сторін у кожне виконання не залежить від повідомлень, отриманих ними в інших виконаннях. Противник, однак, може координувати дії, які він робить у різних виконаннях, і, зокрема, його дії в одному виконанні можуть залежати від повідомлень, отриманих ним в інших виконаннях.

Координація дій у різних виконаннях зазвичай складна, але не неможлива. Таким чином, бажано використовувати композицію (як визначено вище), а не використовувати протоколи, що включають дії з координації між виконаннями, які вимагають від користувачів відстеження всіх виконуваних ними рішень. Насправді спроба координувати чесні виконання протоколу ще проблематичніша, ніж здається, тому що може знадобитися координувати рішення різних чесних сторін (наприклад, усіх співробітників великої кооперації або агентства, що зазнав нападу), що в багатьох випадках є вкрай нереальним. З іншого боку, зловмисник, який атакує систему, може бути готовий взяти на себе додаткові зусилля щодо координації своєї атаки у різних виконаннях протоколу.

Для $T \in \{sequential, parallel, concurrent\}$ ми говоримо, що протокол є T-нульовим розголошенням, якщо він є нульовим розголошенням при композиції типу T. Визначення T-нульового розголошення виводяться з урахуванням відповідних супротивників (тобто верифікаторів); тобто противники, які можуть ініціювати поліноміальну кількість взаємодій з тим, хто доводить, де ці взаємодії заплановані відповідно до типу T. Відповідний симулятор (який ні з ким не взаємодіє) потрібен для отримання вихідних даних, що обчислювально не відрізняються від вихідних даних такого противника типу T.

Твердження 1.4. *Послідовна композиція*

У цьому випадку протокол викликається (поліноміально) багато разів, причому кожен виклик слідує за завершенням попереднього. Принаймні, безпека (наприклад, нульове розголошення) повинна зберігатися при послідовній композиції, інакше застосування протоколу дуже обмежене (оскільки його не можна безпечно використовувати більше одного разу).

Твердження 1.5. *Паралельна композиція*

У цьому випадку (поліноміально) багато екземплярів протоколу викликаються одночасно і виконуються в одному темпі. Тобто ми припускаємо синхронну модель зв'язку і розглядаємо (поліноміально) безліч виконань, які повністю синхронізовані, так що і-те повідомлення у всіх примірниках відправляється точно (або приблизно) одночасно.

В загальному випадку, виходить так, що нульове розголошення не закривається при паралельній композиції. Проте, при стандартних припущеннях про нерозв'язність (наприклад, нерозв'язність факторизації) існують докази з нульовим розголошенням, що є закритими при паралельній композиції. Крім того, ці протоколи мають постійну кількість раундів.

Твердження 1.6. *Одночасна композиція*

Одночасна композиція узагальнює як послідовну, так і паралельну композицію. Тут (поліноміально) багато екземплярів протоколу викликаються у довільний час і виконуються у довільному темпі. Тобто, ми припускаємо асинхронну (а не синхронну) модель спілкування.

1.5 Доведення знання розв'язку певної задачі Діффі–Хеллмана з нульовим розголошенням

Припустимо, що G_1 – це скінченна адитивна підгрупа точок еліптичної кривої E з N елементів, P – базова точка цієї кривої, aP та bP елементи G_1, за якими гравець P розв'язав задачу Діффі–Хеллмана, тобто знайшов точку кривої abP. Гравцю P потрібно довести перевіряльнику V, що він справді знає розв'язок цієї задачі, не допомагаючи йому знайти це значення. При цьому припустимо, що перевіряльнику V відомий порядок q групи G_1. Послідовність кроків, які повинні зробити гравці, така:

1) Гравець P випадково вибирає число e та посилає V значення $B' = ebP$;

2) V вибирає випадковий біт α. Якщо $\alpha = 1$, то гравець P повинен розкрити значення e і гравець V може перевірити, що дійсно $B' = ebP$.

3) Якщо $\alpha = 0$, то гравець P повинен обчислити та передати V значення $eabP$, гравець V повинен перевірити рівність $e(P,eabP) = e(aP,B')$.

4) Кроки $1-3$ повторюються доти, доки гравець V не переконається, що P дійсно знає значення abP.

Властивість повноти тривіальним чином випливає з конструкції самого протоколу, оскільки виникає тотожність:

$$e(P,eabP) = e(P,P)^{eab} = e(aP,P)^{eb} = e(aP,B').$$

Якщо гравець P насправді не знає значення abP, то він може давати правильну відповідь не більше ніж при одному варіанті значення α. Так, якщо виконуючи крок 1 гравець P сподівається, що α прийме значення 1, то він може, не знаючи abP, послати V значення $B' = ebP$. Якщо P сподівається, що α прийме значення 0, то він може послати значення $B' = eP$ (тоді на кроці 3 він повинен замість $eabP$ послати

значення eaP, тому що $e(P,eaP) = e(aP,eP) = e(aP,B'))$. Однак якщо в цьому випадку α прийме значення 1, гравець P не зможе надати правильну відповідь, бо не знає значення eb^{-1}, яке потрібно скалярно помножити на точку bP, щоб отримати $B'(B' = eP = eb^{-1}bP)$. Ймовірність того, що гравець V прийме доведення дорівнює $\frac{1}{2m}$ при m ітераціях протоколу, це обґрунтовує властивість коректності [6].

1.6 Модель узагальненого еталонного рядку посилань

Це теоретична модель інтерактивної системи, яка допускає моделювання інтерактивною ймовірнісною машиною Тьюрінга, в якій усі учасники системи (в тому числі – зі спеціальною роллю зловмисника) отримують доступ до деякого випадкового рядку символів str із визначеним розподілом D, сформованим за спеціальним правилом. Як реалізація моделі може розглядатись послідовність $u_1,u_2,\ldots,u_n$ випадкових чисел довжини k (параметру стійкості), які є результатами обчислень односторонньої функції. Відповідно до неї обчислюється проміжний випадковий рядок $w_1,w_2,\ldots,w_n$, де $w_i = f^{-1}(u_i)$, та значення ядра односторонньої функцій $s_i = B(f^{-1}(u_i))$, які прувер в змозі обчислити, оскільки його обчислювальні можливості є необмеженими [8]. При практичній реалізації це забезпечується існуванням пар відкритих та таємних ключів асиметричної системи, стійкою до атак обраного відкритого тексту. Розкриття значення s_i для верифіканта означає, що прувер надає значення s_i разом із w_i. Верифікант перевіряє $u_i = f(w_i)$ та $s_i = B(w_i)$.

1.7 Місце протоколів неінтерактивних доказів із нульовими знаннями між примітивними протоколами

Аналізу аксіоматичних методів криптографічних протоколів присвячено багато робіт [1,3-8]. Нагадаємо, що центральною ідеєю є побудова криптографічного протоколу будь-якої складності з примітивних криптографічних протоколів, основними з яких є протоколи розподілу секрету (англ. *secret sharing*), протоколи інтерактивних і неінтерактивних доказів (аргументації), протоколи прив'язки до біту (англ. *bit commitment*), протоколи передачі з забування (англ. *oblivious transfer*), протоколи прив'язки із лазівкою (англ. *trapdoor commitment*).

Подальша декомпозиція самих примітивних протоколів дозволила виокремити декілька криптографічних та обчислювальних базових теоретичних моделей, спираючись на які за аксіоматичним підходом будувати практично будь-який криптографічний протокол, стійкість якого спирається на властивості введеної моделі. До таких моделей (в порядку історичного виникнення) відносились [4,7,8]: модель випадкового маяку, випадкового оракула, функції односторонньої перестановки, модель випадкової функції, що перевіряється, модель загального випадкового рядку, узагальнена модель випадкового рядку, модель «правильного» блокчейну.

Паралельно з аналізом теоретичних моделей, за цим же аксіоматичним підходом відбувалось зведення будь-якого криптографічного протоколу до як можна меншої кількості примітивних протоколів. На цьому шляху прогрес практично досяг своєї мети, з розробкою теорії неінтерактивні протоколи з нульовими знаннями, за допомогою яких вважається можна побудувати будь-який криптографічний протокол, та спираючись на стійкість теоретичної моделі, яка покладена в його основу, формально довести стійкість протоколу за аксіоматичним підходом.

Згодом, однак виявились деякі проблеми: по-перше, існування неінтерактивних протоколів з нульовими знаннями неможливе у стандартній моделі обчислень [4]; по-друге, теоретична складність, яка полягає в необхідності генерації достатньо довгого випадкового рядку в довірений для учасників протоколу спосіб, навіть поліноміальна кількість доказів, які використовують один і той самий еталонний випадковий рядок, порушують властивість «нульового знання» [1]; по-третє, складність практичної реалізації, всі відомі реалізації засновані на властивостях білінійних спарюваннях Вейля [6]; в-четверте, порушення припущення про властивості випадкових функцій, що верифікуються або еталонного загального рядку порушує всі гарантії стійкості протоколу.

1.8 Блокчейн як альтернатива безпечної ініціалізації

З появою блокчейнів швидко виникла ідея використання їх як ядра принципово нової формальної теорії криптографічних протоколів [8], а саме, використання блокчейнів замість концепції еталонного випадкового рядку. Життєздатність такої ідеї полягає в тому, що стійкість системи блокчейну базується на інших принципах, ніж інші криптографічні протоколи, а його стан в певні моменти часу може розглядатися подібно до результатів роботи випадкової функції, яка верифікується.

У протоколі блокчейну метою всіх сторін є підтримка (послідовного) глобального упорядкованого набору записів, тому блокчейн розглядається як альтернатива «довіреної безпечної ініціалізації» (англ. *Trusted Setup Assumptions*) [8]. Довірена безпечна ініціалізація є загальним доступним (можливо конфіденційним) значенням, яке є доступним для усіх учасників криптографічного протоколу до його початку. Мета блокчейну є в тому числі усунення єдиної централізованої точки довіри, і з цього боку будь-який протокол з таким елементом може бути перероблений під блокчейн технологію.

Отже, стійкість криптографічних протоколів при такому

аксіоматичному підході може розглядатись шляхом верифікації зведень стійкості протоколу до стійкості протоколу блокчейну при припущенні існування блокчейну із певними властивостями. Головною вимогою до такого блокчейну є стійкість до централізації з плином часу. На жаль відомо, що для «*proof-of-work*» та «*proof-of-stake*» типів блокчейну така властивість не є притаманною. Розглянемо ідею і практичну побудову блокчейну нового типу, який є стійким до централізації із плином часу.

1.9 Ідеї побудови протоколу узгодження на основі блокчейну

Основною ідеєю нового протоколу консенсусу є зміна підходу до розрахунку консенсусної функції та знищення взаємозв'язків між цінним ресурсом, що використовується в консенсусному протоколі та винагороди за перемогу у консенсусі, а саме вибору такого цінного ресурсу, накопичення якого було б не доцільно в економічному або технологічному сенсі. Найбільш підходить для цього «біла» IP-адреса учасника консенсус-протоколу. Зрозуміло, що монополізація всіх білих (зовнішніх) IP-адрес є недоцільною, а для підвищення стійкості протоколу до децентралізації в рамках протоколу можна обмежити кількість IP-адрес учасників за допомогою механізму фільтрації IP-адрес. За допомогою цих ідей запропонований протокол є більш стійким до централізації ніж існуючі протоколи типу «*proof-of-stake*».

До того ж, ймовірність атаки «розгалуження» та атаки «подвійної трати» нижче, ніж для інших консенсусних протоколів типу «*proof-of-work*» та «*proof-of-stake*». Зауважимо, що прив'язка до білої IP-адреси використовується не для майнінгу, а для організації випадкового вибору майнерів, як частина невизначеності інформації, невідомої до початку процесу консенсусу. Цим ми знищуємо зв'язок між

цінним ресурсом, що використовується в консенсусному протоколі та консенсусній винагороді.

Далі важливо, щоб майнер не створив новий блок, поки не буде отримана повна та точна інформація про це у співпраці з іншими майнерами. Точніше, обчислювальна задача, вирішення якої необхідне для досягнення консенсусу, не може бути вирішена з необхідною точністю, поки не відбудеться певна подія в консенсусному протоколі. Такою подією є отримання одним з учасників протоколу повної і точної інформації, необхідної для вирішення обчислювальної проблеми. За основу ідеї побудови нового протоколу пропонується взяти найкращі ідеї від протоколів типу «*proof-of-work*» та «*proof-of-stake*», зокрема від протоколів першого типу – ідею змагання між майнерами за якнайшвидше вирішення складної обчислювальної задачі, від протоколів другого типу – залежність виграшу майнера в змаганні за право генерації наступного блоку від наявної у майнера інформації, необхідної для генерації блоку.

Для унеможливлення згенерувати довільну кількість цінних ресурсів для учасника протоколу, до початку протоколу застосовується два типи обмеження: по-перше, регулюється чисельність учасників, які можуть прийняти участь в протоколі; по-друге, окремі дані рейтингу учасників формуються тільки на поточний сеанс протоколу (як сеансові ключі в схемі Діффі-Хеллмана).

Визначається наступний підхід до побудови протоколу узгодження: пропонується змінити обчислення алгоритму додавання нового блоку в блокчейн при застосуванні протоколу угоди «*proof-of-works*» таким чином, щоб необхідна для роботи алгоритму вихідна інформація була задана неповно і неточно. Значення, яке обчислюється алгоритмом і яке може бути перевірено іншими учасниками протоколу, визначається з точністю, що задається деяким порогом. Вихідна інформація розташовуються на декількох ресурсах, за доступ до яких конкурують учасники протоколу угоди.

1.10 Опис протоколу консенсусу

Наведемо покроковий опис протоколу консенсусу для блокчейну, на основі ідей, запропонованих вище. На першому етапі визначаємо поточну кількість k_i учасників протоколу з усіх активних в даний момент часу n учасників, де IP_1,IP_2,...,IP_n – «білі» IP-адреси учасників.

Для i-го сеансу обираємо сеансові порогові значення ефективності протоколу $0 < l_i < 1$ та $k_i < n$. Вибір значення k_i визначається співвідношенням необхідної швидкості транзакцій і стійкості протоколу до атак типу «*double spend attack*». Вибором значення l_i визначається ймовірність обчислення предиката, який визначається кількістю спроб генерації нового блоку для кожного учасника протоколу. Обираємо k_i учасників (наприклад, за найменшими значеннями IP-адрес, або деякою процедурою випадкового вибору учасників) та за допомогою групового протоколу Діффі-Хеллмана генеруємо спільне випадкове число $R_i < 2^B$, де B обрано із міркувань стійкості до криптоаналізу.

На другому етапі відбувається обчислення рейтингової функції k_i учасниками протоколу. Функцію $F_{l_i} : \{0,1\}^* \to \{0,1\}$ визначаємо як предикат такий, що ймовірність $P(F_{l_i}(^*,R_i) = 1) = l_i$. Побудову предикату для j-го учасника протоколу здійснюємо наступним чином: $X = H_{i-1} \| IP_j \| T_j \| R_i$, де H_{i-1} – геш-код попереднього блока ланцюжків блоків транзакцій, $\|$ – операція конкатенації, T_j – поточне значення лічильника.

Лічильником j-го учасника протоколу є постійно зростаючий ряд натуральних чисел від 1. Обчислюємо геш-код $H(X)$. У випадку $l_i = \frac{1}{2}$ визначаємо парність $H(X)$ та приймаємо $F_{l_i}(H(X)) = 1$, якщо парність $H(X)$ дорівнює 1, та $F_{l_i}(H(X)) = 0$ – інакше. Для $l_i \neq \frac{1}{2}, l_i \approx \frac{C}{D}$ – визначаємо $H(X) \bmod D$, де C та D – деякі цілі числа.

Якщо $l_i < \frac{1}{2}$ і $H(X) \bmod D < C$, приймаємо $F_{l_i}(H(X)) = 1$, та $F_{l_i}(H(X)) = 0$ – інакше. Якщо $\frac{1}{2} < l_i < 1$ і $H(X) \bmod D \geqslant C$, то

$F_{l_i}(H(X)) = 1$ та $F_{l_i}(H(X)) = 0$ – інакше.

Для i-го блоку кожен з k_i учасників обчислює до m значень:

$$F_{l_i,j_1}(H_{i-1}\|IP_j\|1\|R_i), \ldots, F_{l_i,j_m}(H_{i-1}\|IP_j\|m\|R_i).$$

Переможцем стає той, у кого для деякого значення лічильника $T_j = s$ значення предикату $F_{l_i,j_s}(H_{i-1}\|IP_j\|s\|R_i) = 1$. Якщо для деякого $j_1 \neq j_2$ отримано:

$$F_{l_i,j_1,T_{j_1}}(H_{i-1}\|IP_j\|T_{j_1}\|R_i) = F_{l_i,j_2,T_{j_2}}(H_{i-1}\|IP_j\|T_{j_2}\|R_i),$$

тоді переможцем стає той, у кого значення лічильника є меншим. Переможець підписує блок транзакцій.

Висновки до розділу 1

У цій частині було розглянуто основні поняття про нульове розголошення, інтерактивні та неінтерактивні системи доказів з нульовими знаннями, а також блокчейнів, моделі та принципи побудови протоколів на їх основі, їх місце серед криптографічних протоколів довільної складності. Блокчейн – відносно, нова технологія, яка продовжує розвиватися і стрімко набирати популярність, якщо говорити про систему доказів з нульовим розголошенням, то за останні 30 років її дослідження та пов'язаної з нею теорії поступово покращувалися. Нещодавні дослідження переважно зосереджені на застосуванні та підвищенні ефективності неінтерактивних систем доказів. Далі планую проілюструвати техніку перетворення інтерактивного протоколу з нульовими знаннями в неінтерактивний протокол з нульовими знаннями за допомогою евристики Фіата-Шаміра на прикладі найпростішого протоколу знання дискретного логарифму та проаналізувати її стійкість за різних умов, так як дана задача ще не була розв'язана.

2 ОГЛЯД ПРОБЛЕМАТИКИ ЗАДАЧІ ЗНАННЯ ДИСКРЕТНОГО ЛОГАРИФМУ

У цій частині ми описуємо протокол, який дозволяє A переконати B, що він знає розв'язок проблеми дискретного логарифму – тобто, що він знає x, для якого виконується $\alpha^x \equiv \beta \pmod{N}$, – не повідомляючи B нічого про x.

2.1 Огляд задачі дискретного логарифму

Розглянемо проблему:

1) Аліса (сторона A) знає розв'язок проблеми дискретного логарифму для конкретних α, β і N, вона знає такий показник степеня x, що виконується $\alpha^x \equiv \beta \pmod{N}$.

2) Аліса хоче переконати Боба (сторона B), що вона знає x.

3) Аліса не бажає розкривати значення x.

4) Боб приймає експоненційно малу ймовірність того, що Аліса обманює, тобто вона вдає, що знає x, але насправді не знає. Точніше, ймовірність того, що Алісі вдасться обдурити, не виявивши її Бобом, дорівнює 2^{-T}, де T обирається пропорційно необхідному часу та простору.

У цьому документі представлено протокол, який вирішує цю проблему як для випадків, коли N є простим числом, так і для випадків, коли $N = P_1 \cdot P_2$, де P_1 та P_2 є простими числами приблизно однакового розміру. У другому випадку припускається, що A знає факторізацію N. Проте, якщо A не знає цієї факторизації, наш протокол все ще становить інтерес, оскільки за даних α і N вона може вибрати $x \in \{1, \ldots, N-1\}$ навмання, а потім обчислити β просто шляхом піднесення до степеня (або третя сторона може надати A необхідні x і β). Припускається, що B має лише поліноміальну (в $\log N$) обчислювальну потужність, тоді як на

обчислювальні ресурси A не накладено жодних обмежень. Також немає ймовірнісного поліноміального алгоритму для знаходження x за заданими α, β і N, якщо N є простим, або складеним, яке важко розкласти на множники.

2.2 Деякі узагальнення проблеми дискретного логарифму

1) Множинний дискретний логарифм (англ. *Multiple Discrete Log* (MDL)): A показує B, що маючи α та $\beta_1, \ldots, \beta_k$, вона знає $x_1, \ldots, x_k$ такі що $\alpha^{x_1} \equiv \beta_1, \ldots, \alpha^{x_k} \equiv \beta_k$. Цей протокол більш ефективний, ніж застосування базового DL-протоколу для пар $(x_1, \beta_1), \ldots, (x_k, \beta_k)$, так як він дає B однакову ймовірність спіймати злоумисника A. Коли третя сторона створює x_i випадковим чином і надає A необхідні x_i та β_i, цей протокол також пропонує можливість використовувати DL як основу для схеми автентифікації, подібна до *Fiat end Shamir*, чия схема заснована на складності факторизації.

2) Розслаблений дискретний логарифм (англ. *Relaxed Discrete Log* (RDL)): A показує B, що маючи $\alpha_1, \ldots, \alpha_k$ і β, вона знає $x_1, \ldots, x_k$ такі що $\alpha_1^{x_1} \cdot \alpha_2^{x_2} \ldots \alpha_k^{x_k} \equiv \beta$.

3) Одночасний дискретний логарифм (англ. *Simultaneous Discrete Log* (SDL)): A показує B, що маючи $\alpha_1, \ldots, \alpha_k$ і $\beta_1, \ldots, \beta_k$, вона знає x такий що $\alpha_i^x \equiv \beta_i$ для $i = 1, \ldots, k$.

Проблема дискретного логарифму описана вище в Z_N^* (мультиплікативній групі класів лишків за модулем N цілих чисел, взаємно простих з N) з N простим, або складеним. Проте проблему дискретного логарифму можна сформулювати в будь-якій скінченній групі: нехай G – скінченна група, $< \alpha >$ підгрупа породжена $\alpha \in G$ і $\beta \in < \alpha >$; потім знайдемо такий x, щоб $a^x = \beta$. Протоколи, представлені тут, можуть бути виконані в будь-якій групі G, в якій як A, так і B можуть застосовувати групову операцію ефективним способом, наприклад, за поліноміальний час у логарифмі порядку G. (Для

RDL-протоколу ми також маємо припустити, що G є комутативною). Властивості DL-протоколу над Z_N^*, які дозволяють A переконати B з високою ймовірністю, що вона знає дискретний логарифм β відносно α без розкриття будь-яких знань про цей дискретний логарифм, залишаються вірними для DL-протоколу над будь-якою групою G, такою, що A знає (позитивний кратний) порядок α в G, а B знає «хороше» наближення (позитивного кратного) порядку α, тобто якщо m є деяким кратним порядку α, тоді B знає таке ціле число m', що $|m - m'| \leqslant m^c$, де c – деяке число з $0 \leqslant c < 1$. Наприклад, якщо $G = Z_N^*$, то B знає точний порядок G, якщо N — просте, а якщо $N = P_1 \cdot P_2$, де P_1 та P_2 є простими числами порядку $O(N^{\frac{1}{2}})$, то G має порядок $\phi(N) = (P_1 - 1)(P_2 - 1)$, B знає N і $|N - \phi(N)| = O(N^{\frac{1}{2}})$. DL-протокол можна використовувати також, якщо B не знає хорошого наближення для (кратного) порядку α; однак B може отримати таке наближення, досліджуючи повідомлення, які він отримує від A під час участі в протоколі [9]. Крім того, з невеликою модифікацією DL-протокол все ще можливий, якщо А не знає кратного порядку α в G, але тоді протокол пропускає інформацію про x.

Звичайно, протокол представляє інтерес лише в тому випадку, якщо не існує ефективного алгоритму для обчислення дискретного логарифму в G. Окрім випадку $G = Z_N^*$, з N простим або складеним, ми можемо взяти K-кратний прямий добуток Z_N^*, що породжує протокол одночасного дискретного логарифму, або набору точок еліптичної кривої над $GF(P)$ для деякого простого P.

2.3 Базовий протокол дискретного логарифму

Для того, щоб протокол мав сенс, потрібно припустити, що не існує ефективних (поліноміальних у $\log N$ за часом) алгоритмів для обчислення дискретних логарифмів за модулем N для N простих, або складених. Загальноприйнято вважати, що для великих простих чисел N, які задовольняють певні слабкі обмеження, неможливо обчислити

дискретні логарифми в Z_N^*. Ми припускаємо, що обчислення дискретних логарифмів також є важким, коли N є добутком двох простих чисел, який важко розкласти на множники. Наша мотивація, що стоїть за цим припущенням, полягає в тому, що будь-який швидкий метод обчислення для кожної пари $\alpha \in Z_N^*$ і $\beta \in< \alpha >$ цілого числа x з $\alpha^x \equiv \beta \pmod N$ дозволяє ефективно знайти факторізацію N з високою ймовірністю. Дійсно, виберіть γ навмання з Z_N^* і виберіть «ймовірне просте число» p між N і $2N$. Обчисліть $\alpha :\equiv \gamma^{2p}, \beta :\equiv \gamma^2$. Тоді з високою ймовірністю p є простим числом, взаємно простим з $\phi(N)$, де $\beta \in< \alpha >$. Припустимо, що алгоритм дискретного логарифму обчислює з $\beta \equiv \alpha^x$. Тоді $\gamma^{2(px-1)} \equiv 1$, отже γ^{px-1} є квадратним коренем з 1. З імовірністю $\frac{1}{2}$ цей квадратний корінь не дорівнює 1 або -1 і дає факторізацію N. Насправді можна довести наступне більш сильне (і з криптографічної точки зору) твердження [9]. Нехай N – заданий добуток двох великих простих чисел і існує випадковий поліноміальний часовий алгоритм (тобто алгоритм, час роботи якого є поліноміальним за довжиною вхідних даних і який може виконувати незалежне підкидання монет) із такою властивістю: коли алгоритму надається пара α, β як вхідні дані, де α рівномірно розподілено на Z_N^*, а β рівномірно розподілено на $< \alpha >$, тоді ймовірність того, що цей алгоритм виведе ціле число x з $\alpha^x \equiv \beta$ принаймні $1/Q(\log N)$ для деякого полінома Q. Тоді існує випадковий поліноміальний часовий алгоритм, який виводить факторізацію N з ймовірністю принаймні $\frac{1}{2}$.

2.4 Протокол 1: Дискретний логарифм: $\alpha^x \equiv \beta \pmod N$

Крок 0*: A* і *B* відомі наступні дані: $\alpha \in Z_N^*$, $\beta \in< \alpha >$, N;

Повторюємо T разів:

A обчислює: $\gamma :\equiv \alpha^r$; $r \in_R \{1, \ldots, \phi(N)\}$;

Крок 1*: A* відправляє *B* значення γ; *B* обирає b таке, що $b \in_R \{0, 1\}$;

Крок 2*: B* відправляє *A* значення b; *A* обчислює:

$$y :\equiv r + bx \pmod{\phi(N)};$$

Крок 3*: A* відправляє *B* значення y; *B* перевіряє: $\alpha^y = \gamma\beta^b$.

Зауваження. Ми говоримо, що *A* обманює, якщо вона створює свої повідомлення за допомогою деякого ймовірнісного алгоритму іншим способом, ніж описаний у протоколі. Наприклад, якщо *A* не знає дискретного логарифма, то вона може спробувати побудувати свої повідомлення таким чином, щоб вони все ще задовольняли перевірки *B*. *B* обманює, якщо він генерує свої біти на кроці 2 за допомогою випадкового алгоритму поліноміального часу, який не вибирає їх випадковим чином [8,9].

Яким би способом *B* не намагався обдурити, дані, які він отримує під час участі в протоколі, не допомагають йому знайти розв'язок будь-якого рівняння $f(\alpha, \beta, N, z) = 0$ щодо невідомого z. Перед запуском протоколу *B* отримує α, β і N. На кроці 1 *B* отримує $\gamma \in Z_N^*$ від *A*. На кроці 2 *B* генерує біт b. Якщо *B* обманює, то він генерує b іншим способом, ніж просто вибираючи його навмання; він може використовувати всі повідомлення, які він обчислив або отримав раніше (у першому раунді протоколу це лише N, α, β та γ). Під час виконання алгоритму, який створює b, *B* може отримати проміжні результати, деякі з яких він хотів би зберегти для подальших цілей; нехай $\mathbf{b}$ містить проміжні результати, збережені *B*. Нарешті, на кроці 3 *B* отримує ціле число y від *A* таке, що $\alpha^y = \gamma\beta^b$. Таким чином *B* отримує кортеж $(\gamma, \mathbf{b}, b, y)$. Після того, як кроки 1, 2 і 3 були виконані T разів, *B* отримав кортеж $W_B = (\gamma_1, \mathbf{b}_1, b_1, y_1, \ldots, \gamma_T, \mathbf{b}_T, b_T, y_T)$, що містить усі дані, отримані *B* під час його участі в протоколі. Зауважте, що W_B є стохастичним і його розподіл ймовірностей залежить від ініціалізаційної інформації $I_A = (\alpha, \beta, N, x)$.

Припустимо, що *B* має ймовірнісний алгоритм M_f, який обчислює розв'язок рівняння $f(\alpha, \beta, N, z) = 0$ з деякою позитивною ймовірністю.

Далі, припустимо, що існує «симулятор» S з малим (поліноміальним у $\log N$) часом роботи, який створює кортеж W_B^* із приблизно таким самим розподілом ймовірностей, як W_B, на вході $I_A^* = (\alpha, \beta, N)$. Цей симулятор може залежати від способу шахрайства B. Нехай M_f^* алгоритм, який спочатку обчислює W_B^* так само, як S, на вхідних даних I_A^*, а потім обчислює рішення для $f(\alpha, \beta, N, z) = 0$, застосовуючи M_f до I_A^*, і W_B^*. M_f^* виводить рішення для $f(\alpha, \beta, N, z) = 0$ приблизно з тією ж ймовірністю, що й M_f (починаючи з W_B і W_B^* мають приблизно такий самий розподіл ймовірностей) і M_f^* має приблизно такий самий час роботи, як і M_f. Це показує, що протокол не розкриває жодних корисних знань B: алгоритм M_f при введенні даних, зібраних B під час роботи протоколу, не виводить рішення $f(\alpha, \beta, N, z) = 0$ швидше, або з вищою ймовірністю, ніж алгоритм M_f^* при введенні даних лише з I_A^* [9]. Отже, щоб протокол був безпечним, достатньо мати симулятор із малим часом роботи для кожного способу шахрайства B.

Ми розглядаємо криптографічні протоколи з двома сторонами, «доказником» A і «верифікатором» B. І A, і B використовують ймовірнісні машини Тьюрінга T_A і T_B, відповідно, з робочою стрічкою, випадковою стрічкою та «поштовою скринькою». Машини використовують той самий алфавіт $\sum$. Кожна машина може читати лише зі своєї робочої стрічки, випадкової стрічки та поштової скриньки, але вона може писати на своїй робочій стрічці, а також у поштовій скриньці іншої машини. Кожен крок, який виконує машина, визначається станом машини та вмістом її трьох стрічок і не залежить від стану іншої машини. Кожного разу, коли машині потрібно надіслати повідомлення іншій машині, вона копіює це повідомлення з власної робочої стрічки в поштову скриньку іншої машини; тоді інша машина може скопіювати це повідомлення зі своєї поштової скриньки на робочу стрічку. Для зручності ми припускаємо, що машини не працюють одночасно. Таким чином, після того, як машина записує повний рядок повідомлення в поштову скриньку іншої машини, вона зупиняється і знову активується

лише після отримання повідомлення від іншої машини.

Перед запуском протоколу обидві машини знаходяться у фіксованому стані ініціалізації, а робочі стрічки цих машин заповнюються певними даними ініціалізації I_A^*. Крім того, робоча стрічка T_A містить секрет x. Покладіть $I_A = (I_A^*, x)$, тоді I_A є рядком довжини 1, скажімо, над $\sum$. Крім того, на початку обидві випадкові стрічки заповнюються нескінченною кількістю символів, кожен рівномірно вибраний з $\sum$. У кінці протоколу обидві машини мають бути в кінцевому стані. Ми припускаємо, що кількість кроків, виконаних T_B між станом ініціалізації та кінцевим станом, обмежена вище поліномом від l; для наших цілей не має значення, чи кількість кроків, виконаних T_A між станом ініціалізації та кінцевим станом, поліноміально обмежена в l.

Позначимо W_B вміст робочої стрічки T_B у кінцевому стані. W_B містить усі дані, збережені T_B під час роботи протоколу; ці дані можуть містити повідомлення, надіслані та отримані T_B, і деякі кінцеві чи проміжні результати обчислень T_B. Через використання випадкових стрічок W_B є стохастичною змінною, розподіл ймовірностей якої залежить від I_A. Ми припускаємо, що для кожного I_A, W_B приймає свої значення в деякій перелічуваній множині Ω; нехай P_{I_A} позначає розподіл ймовірностей W_B на Ω. Симулятор, заснований на машині T_B, визначається як ймовірнісна машина Тьюрінга, яка створює кортеж W_B^* з майже таким самим розподілом ймовірностей, як W_B (але залежить лише від I_A^*); точніше, якщо $P_{I_A^*}$ позначає розподіл ймовірностей W_B^*, то для кожного I_A з достатньо великою довжиною l маємо [9]:

$$\sum_{\omega \in \Omega} |P_{I_A}(W_B = \omega) - P_{I_A^*}(W_B^* = \omega)| \leqslant C^{-l},$$

де C – деяка абсолютна константа з $C > 1$.

2.5 Протокол 2: Множинний дискретний логарифм:

$\alpha^{x_1} \equiv \beta_1 \pmod N, \ldots, \alpha^{x_k} \equiv \beta_k \pmod N$

Крок 0*:* A і B відомі наступні дані: $\alpha \in Z_N^*$, $\beta_1, \ldots, \beta_k \in < \alpha >$, N;

Повторюємо T разів:

A обчислює: $\gamma :\equiv \alpha^r$; $r \in_R \{1, \ldots, \phi(N)\}$;

Крок 1*:* A відправляє B значення γ;

B обирає $b_1, \ldots, b_k$ такі, що $b_i \in_R \{0, 1\}$;

Крок 2*:* B відправляє A значення $b_1, \ldots, b_k$; A обчислює:

$$y :\equiv r + b_1 x_1 + \ldots + b_k x_k \pmod{\phi(N)};$$

Крок 3*:* A відправляє B значення y; B перевіряє: $\alpha^y = \gamma \beta_1^{b_1} \cdot \ldots \cdot \beta_k^{b_k}$.

Якщо B не обманює і якщо A не знає хоча б одного з дискретних логарифмів $x_1, \ldots, x_k$, тоді будь-яке шахрайство з боку A в протоколі 2 буде виявлено B з ймовірністю $\geqslant 1 - 2^{-T}$.

Зауваження. Можна використовувати протокол 2 як інтерактивну «схему ідентифікації», концепцію, введену Фіатом і Шаміром [10, 16]. Припустимо, що не A, а якийсь «центр», якому взаємно довіряють, генерує випадковим чином x_i, передає їх A (але нікому більше) і зберігає відповідні β_i в якомусь публічному каталозі. Тоді A може ідентифікувати себе B, показавши, що вона знає дискретні логарифми числа β_i, не відкриваючи жодних знань про їхні значення, використовуючи протокол 2. Таким чином, дані, отримані від його взаємодії з A, не дозволять B ідентифікувати себе третім особам як A. Схема Фіата-Шаміра використовує загальнодоступне складене число, факторізація якого відома лише центру. У цій схемі числа β_i для користувача є квадратами за модулем цього складеного, побудованого центром, і має переконати , що вона має квадратні корені з цих β_i. На противагу нашій схемі, яка використовується з простим модулем, у схемі Фіата-Шаміра центр повинен зберігати в секреті деяку інформацію про

люки (а саме факторизацію модуля). З іншого боку, Фіат і Шамір стверджували, що ця схема дозволяє центру формувати β_i деякого користувача A шляхом застосування певної загальнодоступної функції до імені й адреси A. Таким чином, будь-який верифікатор B може самостійно обчислити числа β_i, і їх не потрібно зберігати в загальнодоступному файлі. Функція, яка використовується для побудови β_i, має бути такою, щоб лише центр, знаючи розклад модуля, міг обчислити квадратний корінь із деякого результату функції. Однак наразі невідомо, як довести, що будь-яка така публічна функція заважає людям будувати імена, для яких вони можуть самостійно знаходити відповідні квадратні корені. Схема Фіата і Шаміра ефективніша за нашу, тому що вона вимагає лише зведення у квадрат, тоді як наша схема вимагає піднесення до степеня $logN$-розрядних чисел [9].

Також можна побудувати неінтерактивні протоколи на основі задачі дискретного логарифму. Один приклад такого протоколу – це протокол ідентифікації Шнорра, який заснований на задачі дискретного логарифму для забезпечення безпеки.

Протокол ідентифікації Шнорра дозволяє доводячому переконати перевіряючого у своїй ідентичності, не розкриваючи при цьому будь-якої додаткової інформації. Проте спочатку розглянемо його базовий інтерактивний варіант.

2.6 Протокол 3: Інтерактивна схема ідентифікації Шнорра

Крок 0*:* Вибирається циклічна група простого порядку p, і вибирається генератор цієї групи g.

Крок 1*:* Доводячий генерує приватний ключ x – випадкове ціле число за модулем порядку групи і обчислює відповідний публічний ключ y як $y = g^x \bmod p$ (де g – генератор).

Крок 2*:* Доводячий генерує випадкове значення r і обчислює значення c як $c = g^r \bmod p$.

Крок 3*:* Перевіряючий вибирає випадкове значення виклику e і надсилає його доводячому.

Крок 4*:* Доводячий обчислює значення відповіді s як

$$s = r + e \cdot x \bmod p - 1.$$

Крок 5*:* Перевіряючий перевіряє, чи виконується рівняння $g^s \equiv c \cdot y^e$ за модулем порядку групи. Якщо рівняння виконується, ідентичність доводячого вважається дійсною.

Зауваження. Безпека протоколу ідентифікації Шнорра ґрунтується на припущенні, що обчислення дискретного логарифму є обчислювально складною задачею. Протокол Шнорра є інтерактивним протоколом ідентифікації, що дозволяє довести перевіряючому свою ідентичність, не розкриваючи додаткову інформацію. Однак, застосовуючи перетворення Фіата-Шаміра, інтерактивний протокол Шнорра можна перетворити на неінтерактивну систему доведень. Перетворення Фіата-Шаміра заміняє інтерактивну фазу виклику протоколу хеш-функцією. Це перетворення дозволяє використовувати протокол Шнорра в сценаріях, де потрібна неінтерактивність, наприклад, у цифрових підписах, або доведеннях нульового знання. Варто зазначити, що протокол ідентифікації Шнорра – це лише один приклад неінтерактивного протоколу на основі задачі дискретного логарифму. Інші протоколи, такі як ті, що базуються на варіантах задачі Діффі-Хеллмана або криптографії на еліптичних кривих, також можуть бути використані для побудови неінтерактивних систем доведень.

Висновки до розділу 2

У цій частині була проілюстрована проблематика доведень з нульовими знаннями на основі базового протоколу дискретного логарифму та його модифікацій. Наведено інтерактивний протокол ідентифікації Шнорра, який в наступній частині буде використаний в техніці перетворення інтерактивного протоколу з нульовими знаннями в неінтерактивний протокол з нульовими знаннями за допомогою евристики Фіата-Шаміра на прикладі найпростішого протоколу знання дискретного логарифму, а також подальший аналіз і його перетворення на основі блокчейну.

3 ПЕРЕТВОРЕННЯ ІНТЕРАКТИВНОГО ПРОТОКОЛУ З НУЛЬОВИМИ ЗНАННЯМИ

У цій частині ми проілюструємо набуті навички в застосуванні схеми Фіата-Шаміра, проаналізуємо роль хеш-функції, проаналізуємо можливість та техніку застосування блокчейну, як заміну хеш-функції.

3.1 Протокол 4: Неінтерактивна схема ідентифікації Шнорра

Крок 0*:* Вибирається циклічна група простого порядку p, і вибирається генератор цієї групи g.

Крок 1*:* Доводячий генерує приватний ключ x – випадкове ціле число за модулем порядку групи і обчислює відповідний публічний ключ y як $y = g^x \bmod p$ (де g – генератор).

Крок 2*:* Доводячий генерує випадкове значення r і обчислює значення c як $c = g^r \bmod p$.

Крок 3*:* Замість того, щоб перевіряючий надавав випадкове викликове значення e, виклик обчислюється детерміністично за допомогою хеш-функції, як $e = Hash(c, y)$.

Крок 4*:* Доводячий обчислює значення відповіді s як

$$s = r + e \cdot x \bmod p - 1.$$

Крок 5*:* Перевіряючий перевіряє, чи виконується рівняння $g^s \equiv c \cdot y^e$ за модулем порядку групи. Якщо рівняння виконується, ідентичність доводячого вважається дійсною.

Зауваження. У контексті протоколу Шнорра «$Hash(c, y)$» означає застосування хеш-функції до конкатенації або комбінації значень

c та публічного ключа y. Конкретна хеш-функція може варіюватися, але часто використовуються криптографічні хеш-функції, такі як $SHA-256$, або $SHA-3$. Перевіряючий може незалежно обчислити хеш-функцію на своєму боці, використовуючи спільний референсний рядок та отриманий незалежний доказ. Порівняння обчисленого перевіряючим значення хешу зі значенням, наданим доводячим, дозволяє перевірити валідність доказу. Схема Фіата-Шаміра дозволяє перетворити багато інтерактивних систем доведень, таких як системи, засновані на нульових знаннях, на неінтерактивні версії. Це має практичне застосування в криптографічних протоколах, де неінтерактивні докази можуть бути більш ефективними та зручними у використанні. Схему ідентифікації Шнорра можна поліпшити, включивши концепцію блокчейну для забезпечення додаткової безпеки та прозорості. Далі буде наведена загальна схема, як це може бути реалізовано. Значення c можна поєднати з блокчейном. Блокчейн є публічно відомим значенням b, отриманим з інформації останнього блоку (такої як хеш блоку чи мітка часу блоку).

3.2 Формальні критерії для слабкої хеш-функції та стійкості на блокчейні

Ми показуємо, що будь-яка хеш-функція h (або сімейство H), яка створює екземпляр евристики Фіата-Шаміра для будь-якого з широкого класу протоколів, повинна бути стійкою до колізій. Відповідно це означає, що h можна використовувати для побудови односторонньої функції.

Формальні критерії для слабкої хеш-функції з точки зору ймовірності колізій можна сформулювати наступним чином:

1) Слабка хеш-функція має високу ймовірність колізій, що означає, що вона більш схильна до виникнення колізій для різних вхідних значень. Іншими словами, існує більша ймовірність того, що два різних вхідних значення дають однаковий хеш-вихід.

2) Слабка хеш-функція може мати недостатній обсяг вихідного простору, що означає, що кількість можливих хеш-значень, які вона може виробити, обмежена. Це обмеження збільшує ймовірність колізій, оскільки кількість різних вхідних значень зазвичай набагато більша, ніж кількість можливих хеш-вихідних.

3) Слабка хеш-функція може не володіти рівномірністю у розподілі хеш-значень по вихідному простору. Ця нерівномірність може призводити до утворення груп хеш-значень, збільшуючи ймовірність колізій всередині цих груп.

4) Лавинний ефект відноситься до властивості хеш-функції, при якій невелика зміна вхідних даних повинна призводити до значної зміни вихідних даних. Слабка хеш-функція може показувати слабкий лавинний ефект, що означає, що невеликі зміни вхідних даних можуть не належним чином впливати на вихід, збільшуючи ймовірність колізій.

А тепер наведемо формальні критерії для стійкості на блокчейні, зокрема відносно ймовірності виникнення відгалуження від зловмисника:

1) Стійкий блокчейн повинен мати високу складність для успішного створення відгалуження зловмисником. Відгалуження відноситься до створення окремої гілки блокчейну з різною історією вхідних даних. Чим вища складність відгалуження, тим безпечніший блокчейн від атак, спрямованих на зміну історії вхідних даних.

2) Стійкий блокчейн повинен використовувати надійний механізм консенсусу, який забезпечує згоду учасників щодо валідної версії блокчейну. Розповсюджені механізми консенсусу включають $Proof\ of\ Work$ (PoW) та $Proof\ of\ Stake$ (PoS). Ці механізми вимагають від учасників вкладання обчислювальної потужності для участі у процесі консенсусу, що робить невигідним спробу відгалуження зловмисником.

3) Стійкий блокчейн повинен підтримувати узгодженість мережі, що означає, що всі учасники мережі мають доступ до однакової історії вхідних даних. Вузли мережі повинні регулярно комунікувати і синхронізуватися, щоб запобігти виникненню розбіжностей або відгалужень.

4) Стійкий блокчейн повинен мати широке розподілення вузлів по всій мережі. Ця децентралізація допомагає запобігти одній єдиній сутності, або групі, отримати контроль над більшістю обчислювальної потужності мережі, зменшуючи ймовірність успішних спроб відгалуження.

5) Стійкий блокчейн зазвичай слідує «правилу найдовшого ланцюжка», де валідна версія блокчейну – це та, яка має найбільший ланцюжок підтверджених блоків. Це правило забезпечує, що зловмисникові необхідно контролювати більшість обчислювальної потужності мережі, щоб успішно створити довший ланцюжок і переписати існуючу історію вхідних даних.

Загалом, ймовірність колізії хеш-функції подібна до існування ланцюга блокчейну певної довжини який може сформувати зловмисник.

3.3 Аналіз стійкості схеми Фіата-Шаміра з використанням слабкої хеш-функції

Якщо буде використовуватись слабка хеш-функція в схемі Фіата-Шаміра, безпека схеми може бути скомпрометована у декількох аспектах:

1) Атаки на колізії: Слабка хеш-функція може бути вразлива до атак на колізії, коли різні вхідні дані дають однаковий хеш-вихід. Якщо використовується слабка хеш-функція, атакуючий може знайти два різних випадкових значення, які мають однаковий хеш, що дозволить підробити доказ, не знаючи секрету.

2) Атаки на пошук образу: Слабка хеш-функція може бути вразлива до атак на пошук образу, коли стає вигідним обчислити вхідні дані, які генерують певний хеш-вихід. У схемі Фіата-Шаміра хеш-функція використовується для виведення виклику за допомогою публічної інформації. Якщо використовується слабка хеш-функція, атакуючий може зламати процес генерації виклику та сконструювати дійсний доказ

без володіння секретом.

3) Атаки на оберненість: У деяких випадках слабка хеш-функція може не мати необхідної властивості оберненості, коли обчислення початкового вхідного значення з хеш-виходу є обчислювально складним завданням. Ця оберненість є важливою в схемі Фіата-Шаміра для забезпечення правильності перевірки неінтерактивного доказу. Якщо використовується слабка хеш-функція, атакуючий може легше підробити доказ або зманіпулювати процес перевірки.

Загалом, використання слабкої хеш-функції в схемі Фіата-Шаміра може призвести до вразливостей, що дозволяють атакуючому підробити докази, зманіпулювати процес генерації виклику або скомпрометувати цілісність та безпеку неінтерактивних доказів. Важливо використовувати сильну та безпечну хеш-функцію, яка має властивості, такі як стійкість до колізій, стійкість до пошуку образу та оберненість, для забезпечення надійності схеми.

Проте, слабка хеш-функція не є обов'язковою умовою для слабкості евристики Фіата-Шаміра. Про це детально можна прочитати в роботі [19]. Концептуальний момент цієї роботи: можна довести значущі поняття надійності для протоколу $FiatShamir$, використовуючи властивості безпеки самого неінтерактивного протоколу замість властивостей безпеки хеш-функції.

3.4 Протокол 5: Неінтерактивна схема ідентифікації Шнорра з блокчейном

Крок 0*:* Вибирається циклічна група простого порядку p, і вибирається генератор цієї групи g, також маємо значення блокчейну b.

Крок 1*:* Доводячий генерує приватний ключ x – випадкове ціле число за модулем порядку групи і обчислює відповідний публічний ключ y як $y = g^x \bmod p$ (де g – генератор). Публічний ключ y пов'язується з адресою

доводячого в блокчейні.

Крок 2*:* Доводячий генерує випадкове значення r і обчислює значення c як $c = g^r \bmod p$.

Крок 3*:* Виклик(e) обчислюється детерміністично за допомогою блокчейну (тобто ми маємо: Функцію генерації псевдовипадкового числа $VRF_{val}(Key_{prv} = x,\ c)$; Функцію вироблення доведення $VRF_{prove}(Key_{prv} = x,\ c)$, яка для val обчислює $proof(c)$, що підтверджує «коректність» val; Функцію $VRF_{ver}(Key_{pub} = y,\ c,\ val,\ proof(c))$ верифікації доведення $proof(c)$; Подібно до псевдовипадкових функцій – зловмисник не здатний відрізнити значення псевдовипадкової функції від випадкового числа, навіть за умови, що йому відомі її значення в інших точках. Доказ може бути перевірений будь-ким, хто знає відкритий ключ).

Крок 4*:* Доводячий обчислює значення відповіді s як

$$s = r + e \cdot x \bmod p - 1.$$

Крок 5*:* Перевіряючий перевіряє, чи виконується рівняння $g^s \equiv c \cdot y^e$ за модулем порядку групи. Якщо рівняння виконується, ідентичність доводячого вважається дійсною.

Зауваження. Інтеграція блокчейну в схему ідентифікації Шнорра забезпечує зв'язок підписів зі станом блокчейну на момент їх створення. Це гарантує, що підписи не можуть бути підроблені або змінені без виявлення, оскільки будь-яка модифікація блокчейну призведе до недійсності підписів. Інтеграція блокчейну додає додатковий рівень прозорості та безпеки до схеми ідентифікації Шнорра, що робить її придатною для застосувань, де потрібна децентралізована та перевірена підписова схема, наприклад, у блокчейн-системах та криптовалютах.

Якщо блокчейн має використовуватись замість хеш-функції в криптографічних алгоритмах, він повинен мати певні властивості для забезпечення потрібного рівня безпеки та функціональності. Ось деякі

властивості, яким повинен володіти блокчейн:

1) Визначений вихід та випадковість: Блокчейн повинен давати випадковий визначений вихід, що означає, що для певного вхідного значення воно завжди повинно генерувати однаковий вихід. Визначеність забезпечує послідовність та дозволяє перевірити цілісність даних.

2) Відсутність колізій та непередбачуваність: Блокчейн повинен мати алгоритм, що є стійким до колізій, щоб уникнути ситуацій, коли різні вхідні дані дають однаковий результат, який до того ж не має бути прогнозованим. Це важливо для забезпечення цілісності даних та запобігання несанкціонованим змінам.

3) Односторонність: Блокчейн повинен забезпечувати односторонність, тобто після підтвердження блоку ми вже не зможемо його змінити.

4) Криптографічна стійкість: Блокчейн повинен мати високий рівень криптографічної стійкості проти різних криптографічних атак, таких як атаки пошуку образу, атаки на оберненість та атаки колізії.

3.5 Неінтерактивний протокол з нульовим знанням через блокчейн

Більшість відомих конструкцій неінтерактивних доказів з нульовим знанням належать до так званої моделі загального еталонного рядка (CRS), де існує надійна третя сторона, яка публікує деякі публічні параметри. Ми показуємо, що вже існуючі системи блокчейнів, потенційно можуть бути використані як основа замість концепції загального еталонного рядоку (CRS) для реалізації систем неінтерактивних доказів.

У блокчейнах на основі POS кожен учасник (крім зберігання локального блокчейну B) також має право на певну ставку в системі, яку можна виміряти як позитивну раціональну величину. Ідеологія майнінгу в

системі на базі POS полягає в тому, що ймовірність того, що будь-яка сторона зможе створити наступний блок, пропорційна її ставці. Крім того, кожна сторона, яка генерує блок, повинна надати підтвердження ставки, яке може використовуватися як сертифікат іншими сторонами для перевірки правильності.

Припустимо, що всі сторони, які виконують протокол блокчейну, мають однакову суму ставки (тобто кожен новий блок містить підтвердження ставки фіксованої суми). Крім того, супротивник контролює лише міноритарну ставку в системі (скажімо α).

Спочатку ми визначимо поняття форка щодо блокчейнів. Нехай B – деякий блокчейн. Форк відносно B – це послідовність дійсних блоків, яка розширює деякий префікс блокчейну B замість розширення B безпосередньо з його кінця. Іншими словами, форк (або розгалуження) – це послідовність дійсних блоків, яка починає розширювати ланцюжок з деякого блоку, який не є останнім доданим блоком у B. Блокчейни будь-яких двох чесних сторін у будь-яких двох (можливо, різних) раундах під час виконання протоколу можуть відрізнятися лише в останніх блоках, з майже незначною ймовірністю.

Припустимо, $Com(\cdot)$ є неінтерактивною статистичною схемою зобов'язань. Нехай B позначає поточний стан блокчейну, а супротивник контролює не більше α ставки загальної ставки в мережі блокчейну. Схема працює наступним чином. Довідник будує неінтерактивну систему з нульовим знанням як:

1) Обчислення зобов'язань $c_1 \leftarrow Com(w)$ і $c_2 \leftarrow Com(f)$, де w є свідком для даного оператора $x \in L$, а f є просто рядком із усіма нулями відповідної довжини.

2) Обчислення неінтерактивної нерозрізнюваності свідка ($NIWI$) за допомогою свідка w, підтверджуючи, що:

а) c_1 є зобов'язанням дійсного свідка $x \in L$, або

б) c_2 – це зобов'язання довгого форку щодо блокчейну B (тобто іншої послідовності дійсних блоків), так що кількість доказів ставки,

присутня у форку, становить явну більшість (від загальної ставки).

Повнота безпосередньо випливає з правильності базових примітивів. Щоб довести властивість нульового знання, нам потрібно було б побудувати симулятор, який не мав би свідка w, але міг би створити докази, які неможливо відрізнити від чесно згенерованих доказів.

Більш формально:

Для пари ймовірнісних машин Тьюрінга (P,V), в яких P і V – є ймовірнісними поліноміальними за часом, (P,V) називається неінтерактивною системою доказу з нульовим розголошенням через протокол блокчейну Γ^V для мови $L \in NP$ із відношенням-свідком R, якщо виконуються наступні умови:

Повнота: $\forall(x, w)$ таких, що $R(x, w) = 1$, усіх супротивників A та гравців i,j у середовищі Z, існують незначні функції $negl_1(\cdot)$, $negl_2(\cdot)$ такі що:

$$Pr[V(B', x, \pi) = 1 : view \leftarrow EXEC^{\Gamma^V}(A, Z, 1^\lambda);\ \pi \leftarrow P(B, x, w)] \geqslant \\ \geqslant 1 - negl_1(|x|) - negl_2(\lambda),$$

де $view_i$ та $view_j$ позначають погляди гравців i та j, обидва i,j є чесними, $B = GetRecords(view_i)$, $B' = GetRecords(view_j)$, $GetRecords$ – це алгоритм, що виводить найдовшу впорядковану послідовність дійсних блоків B (або просто блокчейн), що міститься у змінній стану, $EXEC^{\Gamma^V}(A, Z, 1^\lambda)$ – це випадкова змінна, що позначає спільний погляд усіх сторін у виконанні блокчейну, де противник A контролює всі корумповані сторони, λ – параметр безпеки.

Коректність: $\forall x \notin L$, усіх супротивників A та кожного гравця i в середовищі Z існують незначні функції $negl_1(\cdot)$, $negl_2(\cdot)$, такі що:

$$Pr[V(B,x,\pi)=1 : view \leftarrow EXEC^{\Gamma^V}(A(x),Z,1^\lambda);\ \pi \leftarrow A] \leqslant$$
$$\leqslant negl_1(|x|) + negl_2(\lambda).$$

Нульове знання: Існує ймовірнісна поліноміальна машина Тьюрінга Sim, така, що $\forall(x,w)$ підпорядкованих $R(x,w)=1$ і всіх супротивників A та кожного гравця i в середовищі Z маємо наступне:

$$\{(\pi, view_A) : view \leftarrow EXEC^{\Gamma^V}(A, Sim, Z, 1^\lambda);\ \pi \leftarrow Sim(x)\}$$
$$\approx_c$$
$$\{(\pi, view_A) : view \leftarrow EXEC^{\Gamma^V}(A, Z, 1^\lambda);\ \pi \leftarrow P(B,x,w)\},$$

де $EXEC^{\Gamma^V}(A, Sim, Z, 1^\lambda)$ – це випадкова змінна, що позначає спільний погляд усіх сторін у виконанні блокчейну, де противник A контролює всі корумповані сторони, а Sim контролює всі чесні сторони.

Ми зазначаємо, що наведену вище ідею на основі (POS) також потенційно можна перенести на блокчейни на основі (POW) із наступним застереженням: неінтерактивний доказ, створений доводячим, буде дійсним протягом обмеженого періоду часу. Проте безпека тепер буде спиратися на той факт, що будь-який супротивник, який контролює помітно менше половини обчислювальних ресурсів, не може обчислити форк довжини набагато довшої, ніж чесні сторони блокчейна. Інтуїтивно зрозуміло, що цього не може статися, оскільки це означало б, що будь-який супротивник, який має лише міноритарне право голосу, може форкнути блокчейн у будь-якому раунді.

Висновки до розділу 3

У цій частині було проілюстровано перетворення інтерактивного протоколу з нульовими знаннями в неінтерактивний протокол з нульовими знаннями за допомогою евристики Фіата-Шаміра на прикладі найпростішого протоколу знання дискретного логарифму, а також перетворення неінтерактивного протоколу з нульовими знаннями в протокол з блокчейном. Проведений аналіз як впливає вибір хеш-функції на стійкість протоколу і можливість застосування схеми Фіата-Шаміра, на прикладі слабкої хеш-функції. А також показано якими властивостями має володіти блокчейн, щоб ним можна було замінити хеш-функцію.

ВИСНОВКИ

У ході даної роботи був проведений огляд та аналіз опублікованих джерел за тематикою нульове розголошення, інтерактивні та неінтерактивні системи доказів з нульовими знаннями, а також блокчейнів, який показав, що блокчейн – відносно, нова технологія, яка продовжує розвиватися і стрімко набирати популярність, щодо системи доказів з нульовим розголошенням, то за останні 30 років її дослідження та пов'язаної з нею теорії поступово покращувалися. Нещодавні дослідження переважно зосереджені на застосуванні та підвищенні ефективності неінтерактивних систем доказів.

Під час розробки теоретичного матеріалу було викладено моделі та принципи побудови протоколів на основі інтерактивних та неінтерактивних систем доказів з нульовими знаннями, їх місце серед криптографічних протоколів довільної складності.

В основі практичної розробки алгоритмів було проілюстровано перетворення інтерактивного протоколу з нульовими знаннями в неінтерактивний протокол з нульовими знаннями за допомогою евристики Фіата-Шаміра на прикладі найпростішого протоколу знання дискретного логарифму, а також їх перетворення в протокол з блокчейном.

Основні результати, одержані в ході дослідження полягають в тому, що ймовірність колізії хеш-функції подібна до існування ланцюга блокчейну певної довжини який може сформувати зловмисник. Загалом, використання слабкої хеш-функції в схемі Фіата-Шаміра може призвести до вразливостей, що дозволяють атакуючому підробити докази, зманіпулювати процес генерації виклику, або скомпрометувати цілісність та безпеку неінтерактивних доказів. Проте, слабка хеш-функція не є обов'язковою умовою для слабкості евристики Фіата-Шаміра.

ПЕРЕЛІК ПОСИЛАНЬ

1. Blum M. Noninteractive Zero-Knowkedge / Manuel Blum, Alfredo De Santis, Silvio Micali, Giuseppe Persiano., 1991. – (SIAM J. COMPUT).

2. Myronets I. Cryptographic algorithms and features of their use in blockchain systems [Електронний ресурс] / Myronets I., Shkrebtii A. // Ukrainian Scientific Journal of Information Security. – 2019. — Режим доступу до ресурсу: http://jrnl.nau.edu.ua/index.php/Infosecurity.

3. Goldreich O. Zero-Knowledge twenty years after its invention / Oded Goldreich. – Rehovot, Israel, 2002. – (Department of Computer Science Weizmann Institute of Science).

4. Bellare M. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols / Mihir Bellare, Phillip Rogaway., 1995.

5. Wu H. A Survey of Noninteractive Zero Knowledge Proof System and Its Applications / Huixin Wu, Feng Wang., 2014. – (Hindawi Publishing Corporation).

6. Kulaga A. Zero-Knowledge proof of Diffie–Hellman problem solution [Електронний ресурс] / Kulaga A.. – 2012. — Режим доступу до ресурсу: http://ekmair.ukma.edu.ua/bitstream/handle/123456789/1894/Kulaha_Protokol_dovedennia_znannia.pdf?sequence=3&isAllowed=y.

7. Кудін А.М. Асиметричні криптографічні протоколи з блокчейн-ядром: проблеми побудови та їх рішення / Фізико-математичне моделювання та інформаційні технології. / Кудін А.М., Селюх П.В.. – Львів, 2021.

8. Goldreich O. Foundations of Cryptography. Volume 1. Basic Tools. / Oded Goldreich. – London, 2001. – (Cambridge University Press).

9. Chaum D. An improved protocol for demonstrating possession of discrete logarithms and some generalizations / D. Chaum, J. Evertse. – Amsterdam: Centre for Mathematics and Computer Science Kruislaan.

10. Fiat A. How To Prove Yourself: Practical Solutions to Identification and Signature Problems / A. Fiat, A. Shamir. – Israel: Department of Applied Mathematics The Weizmann Institute of Science Rehovot.

11. Feige U. Multiple non-interactive zero knowledge proofs under general assumptions / U. Feige, D. Lapidot, A. Shamir., 1999. – (SIAM Journal of Computing).

12. Goya R. Overcoming cryptographic impossibility results using blockchains / R. Goya, V. Goyaly.

13. Goldwasser S. Knowledge complexity of interactive proof systems / S. Goldwasser, S. Micali, C. Rackoff., 1989. – (SIAM Journal on Computing).

14. Blum M. Non-interactive zeroknowledge and its applications / M. Blum, P. Feldman, S. Micali., 1988. – (Annual ACM symposium on Theory of computing (STOC '88)).

15. Bellare M. Certifying permutations: noninteractive zero-knowledge based on any trapdoor permutation / M. Bellare, M. Yung., 1996. – (Journal of Cryptology).

16. Bellare M. New paradigms for digital signatures and mes-sage authentication based on non-interative zero knowledge proofs / M. Bellare, S. Goldwasser., 1989. – (Lecture Notes in Computer Science).

17. Micali S. Noninteractive zeroknowledge with preprocessing / S. Micali, G. Persiano., 1990. – (Lecture Notes in Computer Science).

18. Brassard G. Minimum disclosure proofs of knowledge / G. Brassard, D. Chaum., 1988. – (Journal of Computer and System Sciences).

19. Chen Y. Does Fiat-Shamir Require a Cryptographic Hash Function? / Y.Chen, A. Lombardi, F. Ma, W. Quach., 2021. – (Defense Advanced Research Projects Agency).