НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ
КАФЕДРА МАТЕМАТИЧНОГО АНАЛІЗУ ТА ТЕОРІЇ ЙМОВІРНОСТЕЙ

«На правах рукопису»
УДК _____

«До захисту допущено»
Завідувач кафедри
_____ О. І. Клесов
«23» березня 2018 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 111 «Математика»

на тему:
«Дослідження криптографічних алгоритмів для генерації ключів»

Виконав:
студент VI курсу, групи ОМ-61м
Ткачук Володимир Володимирович _____

Керівник:
завідувач кафедри математичного аналізу та теорії
ймовірностей, доктор фізико-математичних наук, професор
Клесов О. І. _____

Рецензент:
завідувач кафедри інформаційних систем і технологій,
доктор фізико-математичних наук, професор
Гавриленко В.В. _____

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

Київ – 2018 року

Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»

Фізико-математичний факультет

Кафедра математичного аналізу та теорії ймовірностей

Рівень вищої освіти – другий (магістерський) за освітньо-науковою програмою

Спеціальність – 111 «Математика»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ О. І. Клесов
(підпис)

«14» травня 2018 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Ткачуку Володимиру Володимировичу

1) Тема дисертації «Дослідження криптографічних алгоритмів для генерації ключів», науковий керівник дисертації Клесов О.І., доктор. фіз.-мат. наук, професор, затверджені наказом по університету від «23» березня 2018 р. № 1016-с.
2) Термін подання студентом дисертації: 04 травня 2018 р.
3) Об'єкт дослідження: сучасна ІТ технологія Blockchain, на базі якої побудовані більшість з існуючих наразі криптовалют.
4) Предмет дослідження: сучасна криптографія та технологія Blockchain.
5) Перелік завдань, які потрібно розробити:
   1.1. дослідити технологію Блокчейн та принцип роботи криптовалюти Bitcoin.
   1.2. проаналізувати які криптографічні алгоритми використовує система BTC.
   1.3. розробити комп'ютерну програму, яка генерую одну з важливих частин системи – bitcoin-address.
6) Орієнтовний перелік графічного (ілюстративного) матеріалу: 49 слайдів.
7) Дата видачі завдання 05 лютого 2018 року.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Ознайомлення з літературою за темами: криптографія, інформаційна безпека, економіка, фінанси	05.02.2018 – 17.02.2018	виконано
2.	Дослідження сучасної IT технологія Blockchain і її реалізації в цифровій валюті - Bitcoin	17.02.2018 - 05.03.2018	виконано
3.	Аналіз криптографічних алгоритмів в криптовалюті Bitcoin	05.03.2018 – 18.03.2018	виконано
4.	Розробка першої частини програми	18.03.2018 – 01.04.2018	виконано
5.	Аналіз технології блокчейн у різних сферах діяльності	01.04.2018 – 13.04.2018	виконано
6.	Реалізація комп’ютерної програми для створення біткоїн-адреси в системі Bitcoin	13.04.2018 – 29.04.2018	виконано
7.	Підготовка магістерської дисертації	29.04.2018 – 02.05.2018	виконано

Студент _____ В. В. Ткачук

(підпис)

Науковий керівник дисертації _____ О. І. Клесов

(підпис)

Реферат

Магістерська дисертація виконана на 64 сторінках, містить 18 ілюстрацій, 5 таблиць та 1 додаток.

Дисертацію присвячено актуальній темі – дослідженню криптографічних алгоритмів для генерації ключів в блокчейні Bitcoin.

Мета дослідження – на основі криптографічних алгоритмів, які містить криптовалюта Bitcoin, розробити на мові програмування Java, програму, котра генерує біткоїн-адресу.

Об’єкт дослідження – технологія Blockchain на базі якої побувдовані більшість із криптовалют.

Предмет дослідження – сучасна криптографія та технологія блокчейн.

Методи дослідження – аналіз технології блокчейн, концепції створення криптовалюти та сам принцип роботи BTC.

Наукова новизна – було розроблено програму, задача якої полягала у генерувані біткоїн-адреси, якою користувач системи біткоїн міг користуватись, здійснюючи власні транзакції.

Вдосконаливши продукт, створивши тим самим власний крипто-гаманець, можна прив’язати його до будь-якого API українського банку та використовувати його як обмінний пункт з цифрової валюти на будь-які фіатні кошти.

Ключові слова: Blockchain, Bitcoin, BTC, cryptocurrency, information security, цифрова валюта, електронні гроші, криптовалюта, цифровий підпис, Java.

Реферат

Магистерская диссертация выполнена на 64 страницах, содержит 18 рисунков, 1 приложение.

Диссертация посвящена актуальной теме - исследованию криптографических алгоритмов для генерации ключей в блокчейне Bitcoin.

Цель исследования – на основании криптографических алгоритмов системы биткоин, реализовать на языке программированнии Java, программу генерации биткоин-адресса.

Объект исследования – технология блокчейн, на базе которой построенны большинство из криптовалют.

Предмет исследования – современная криптография и технология блокчейн.

Методы исследования – анализ технологии блокчейн, концепция создания криптовалюты, принцип работы BTC.

Научная новизна – была разработана программа, задача которой в генерации-биткоин адресса, которой пользователь системы биткоин мог использовать, осуществляя некии транзакции с ней.

Усовершенствовав продукт, тем самым создавая крипто-кошелек, можно привязать его к любому API украинского банка и использовать его в качестве обменного пункта: с цифровой валюты на любые фиатные деньги.

Ключевые слова: Blockchain, Bitcoin, BTC, cryptocurrency. information security, цифровая валюта, электронные деньги, криптовалюта, цифровая подпись, Java.

Abstract

Master's thesis performed at 64 pages, contains 18 illustrations, 1 supplement.

The thesis is devoted to the actual topic – Researching on cryptographic algorithms for key generation of in Bitcoin blockchain.

The aim - is to use the cryptographic algorithms of the bitcoin system, to implement the bitcoin address generation program in the Java programming language.

The object of study – is the technology of blockchain, on the basis of which most of the crypto-currencies are built.

Purpose of the study - modern cryptography and technology of blockchain.

Methods - analysis of blockchain technology, the concept of creating a crypto currency, the principle of BTC.

Scientific novelty - a program was developed whose task is to generate a bitcoin address, which the user of the bitcoin system could use, carrying out some transactions with it.

Having improved the product, thereby creating a crypto-purse, you can bind it to any API of the Ukrainian bank and use it as an exchange: from a digital currency to any fiat money.

Keywords: Blockchain, Bitcoin, BTC, cryptocurrency. information security, digital currency, electronic money, crypto currency, digital signature, Java.

Зміст

Перелік скорочень, умновних позначень, термінів

Blockchain	Ланцюжок з формованих блоків транзакцій, який побудований за певними правилами.
Bitcoin	P2P платіжна система, яка використовує однойменну розрахункову одиницю і однойменний протокол передачі даних [1].
Біткоїн-адреса	Біткоїн-адреса подібна до фізичної чи електронної адреси. Це єдина інформація, яку необхідно передати тому, хто надсилатиме Вам біткоїни, важлива відмінність полягає у тому, що кожна адреса повинна використовуватись тільки для однієї транзакції [1].
Блок	Це частина ланцюжку блоків, що містить та підтверджує багато транзакцій, що очікують підтвердження [1].
Криптовалюта	Цифрові рахункові одиниці, облік яких децентралізований.
Майнінг	Процес, що передбачає використання апаратних ресурсів комп'ютера з метою виконання математичних розрахунків для підтвердження транзакцій і забезпечення безпеки мережі Біткоїн. Винагородою за свої послуги майнери можуть збирати комісії за підтверджені ними транзакції - новостворені біткоїни [1].
Ланцюжок блоків	Публічний запис біткоїн-транзакцій у хронологічному порядку. Ланцюжок блоків єдиний

	для усіх біткоїн-кристувачів [1].
Підпис	Криптографічний цифровий підпис - це математичний механізм, що дозволяє підтвердити право власності [1]
BTC	Загальноприйнята одиниця, що використовується для позначення одного біткоїну [1]
Приватний ключ	Це секретна послідовність певних даних, що дає вам право витрачати цифрову валюту з конкретного гаманця за допомогою криптографічного підпису[1].
P2P	Термін peer-to-peer означає системи, що працюють як організована спільнота, надаючи можливість кожному учаснику безпосередньо взаємодіяти з іншими [1].

ВСТУП

За декілька останніх місяців технологія Blockchain стрімко почала набирати популярності в нашій країні.

Хоча у світі вже існують та успішно працюють ефективні рішення побудовані на базі Blockchain, такі як наприклад Bitcoin - інноваційна мережа платежів та цифрова валюта, Brave – браузер, який має можливість проводити анонімні платежі власникам сайтів, та багато інших успішних реалізацій.

Блокчейн в найпростішому розумінні це просто розподілена база даних, до якої кожен може безпечно підключитись та виконати транзакційний код. Всі транзакції зберігаються в блоках даних, які створюються таким чином, що ними досить складно маніпулювати після того, як вони вже потрапили до системи Blockchain. Для того, щоб блок потрапив до Blockchain потрібно здійснити верифікацію цього блоку і додати його до системи, дана процедура має назву - майнінг. Блокчейн має змогу вирішувати проблеми такі як: безпека, висока доступність та швидкість виконання транзакцій.

Як було вже зазначено, досить часто користувачами даної технології є цифрові валюти, невийняток і найвідоміша з них – біткоїн. Ринок цифрових валют досить швидко розвивається, так максимальна капіталізація ринку криптовалют складала 190 млрд. доларів, а курс біткоїна перетнув відмітку 20000 доларів.

Проте це не єдине ефективне застосування цієї технології, так, ринок стартапів на базі використання технології blockchain, за оцінками експертів, залучить у 2018 році інвестицій на суму 3 млрд. доларів, що цілком впевнено робить технологію альтернативою традиційним венчурним інвестиціям.

Варто звернути увагу, що використання технології блокчейн викликає зацікавлення і в Україні. Так стало відомо, що Україна уклала угоду з міжнародною технологічною компанією Bitfury Group про переведення всіх електронних державних даних на блокчейн [2].

Таким чином стає зрозуміло, що технологія Blockchain є певною мірою революційною та може бути використана в різних сферах людської діяльності. Зокрема, мова йде про такі реалізації як: криптовалюти, різного роду реєстри, корпоративного, або ж державного значення.

Розділ 1. Blockchain

Цілком зрозуміле питаня щодо терміну «*Blockchain*». Що це за технологія? Як працює? І чому її значущість прирівнють як до появи, свого часу, Internet. В даному розділі, ми розглянемо концепцію блокчейн, її основні можливості, а також її конкурентноспроможні переваги перед іншими, існуючими на даний момент, системами. І так, що ж таке блокчейн?

1. Блокчейн – це розподілена система управління базами данних.
2. Блокчейн – це спеціальна струткура для запису групи транзакцій.
3. Блокчейн – це загальнодоступна технологія, що використовує криптографічне хешування, та разом з ним цифровий підпис задля забезпечення безпечних транзакцій, які вже потрапишви до системи не можна змінити.
4. Блокчейн – це технологія, яка отримала великої популярності саме завдяки її використанню при побудові криптовалюти Bitcoin.
5. Блокчейн – інноваційна «книга обліку», де кожний із користувачів може вносити або ж читати записи інших.
6. Блокчейн – це технологія з трьома функціями безпеки: незмінність, прозорість і автономія.
7. Блокчейн – це принципово нова надійна технологія зберігання записів, яка може кардинально змінити підхід до формування і зберігання баз даних.
8. Блокчейн – це програмний продукт, який дозволяє зберігати будь-які дані використовуючи Інтернет захищеним і прозорим способом, не маючи при цьому центрального керуючого органу.
9. Блокчейн – описує ланцюжок блоків, в яких зберігається інформація будь-якого виду: транзакції, контракти, документи про власність, твори мистецтва, патенти і т.п.
10. Блокчейн – це не біткоїн.

Зробивши висновок вищесказаного, можна охарактеризувати blockchain ще й так.

Поява, а затім і стрімка популярність bitcoin сприяла розповсюдженню технології blockchain, на якій, власне і побудована система цифрових валют.

Спосіб зберігання даних, який ще називається цифровим реєстром будь-яких операцій, впорядкований у блоки за ланцюговим принципом має назву – «Blockchain» (англійською **block** — блок, **chain** — ланцюг).

Поняття «*blockchain»* упроваджене анонімним Satoshi Nakamoto у 2008 році, а рік по тому ним же, реалізована відповідна технологія в рамках цифрової валюти — біткоїна. Саме дана технологія стала першою, яка змогла вирішити інформаційну проблему, таку як забезпечення довіри між сторонами до отриманої інформації без залучення зовнішніх гарантів — банків, посередників тощо.

Основа технології blockchain - в розподіленому зберіганні інформації. Це дозволяє зберігати важливу інформацію одночасно на багатьох серверах, при цьому зберігати їх відкрито і безпечно. Наприклад, на базі цієї технології можна зберігати як історію банківських транзакцій клієнтів, результати голосувань, так і базу контрактів, відбитків пальців або історій хвороб. Та інформація, яка зберігається одночасно у багатьох місцях, неможливо вкрасти, тому що у будь-якому випадку їх можна буде відновити із оригінальних джерел.

Як уже зазначалось, блокчейн, себто блок транзакцій — це структура для запису групи транзакцій. Транзакція здійснюються лише тоді, коли вважається підтвердженою. Це надійно та зручно, коли говориться про проведення платежів або ж передачі конфіденційних даних. Так щоб транзакція вважалася підтвердженою, її формат та її підписи мають бути перевірені. Після цього групу транзакцій записують в спеціальний блок. В даних блоках всі дані швидко можна перевірити. А ще в кожному наступному зберігається інформація про попередній. Наприклад, при операціях над криптовалютами, у ланцюжку блоків міститься інформація про всі виконані коли-небудь дії з біткоїнами.

В block входять заголовок та список транзакцій. Заголовок блоку має в собі свій власний хеш, хеш попереднього блоку, хеші транзакцій та іншу додаткову службову інформацію. Першою, що вказується в транкакційному блоці це отримання комісії, яка стане як нагорода, тому користувачеві, який власне і створить даний блок. Для проведення транзакцій в блоці використовують деревоподібне хешування.

Так як результат функції SHA-256 (хешування) непередбачуваний, немає алгоритму отримання бажаного результату, окрім випадкового перебору. Якщо хеш не задовольняє певній умові, то за замовченням змінюється блок службової інформації в заголовку — вже тоді хеш перераховується. Після того як співпали варіанти, вузол розсилає отриманий блок всім іншим підключеним вузлам, які перевіряють блок. При наявності, що блок помилок немістить, тоді він вважається доданим в ланцюжок і наступний блок повинен включити в себе вже його хеш. А тоді все починається спочатку.

1.1 Різновиди систем керувань : централізовані, децентралізовані, розподілені

Раніше було зазначено, система блокчейн – це ***розподілена система управління*** базами данних. Що розуміють під розподіленою системою? Які існують інші системи керування, у чому їх плюси та мінуси. Саме в даному підрозділі ми дамо відповідь на ці питання.

Наразі відомі три системи керування : централізовані, децентралізовані та розроділені

Централізовані системи

Централізовані системи мають тільки одну точку управління, в якій зосереджений весь контроль за системою (рис. 1.1). Всі процеси виконуються тільки в цій точці, в ній же приймаються і всі рішення. Однак, це робить систему надзвичайно слабкою, адже у будь-який збій - цей єдиний центр управління може обрушити всю систему [5].

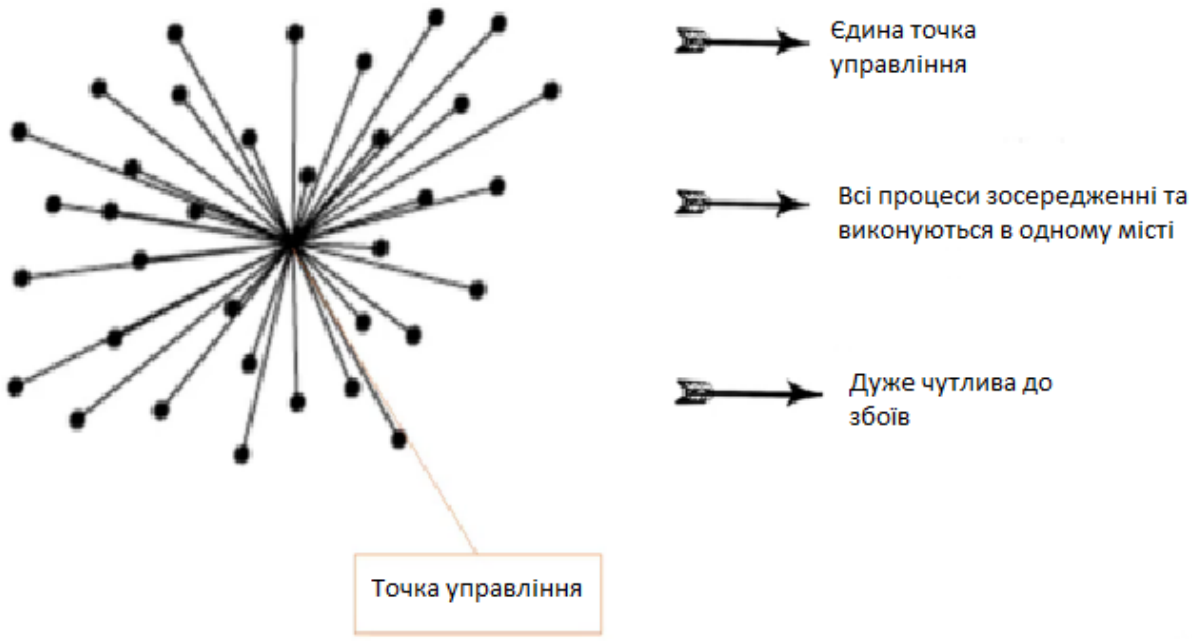


Рис.1.1

Централізовані системи :

Плюси централізованої системи:

1. Її легко реалізувати і всі рішення приймаються набагато швидше. Так як в ній тільки одна точка управління, в якій зосереджений весь контроль за системою, консенсусу не потрібно.

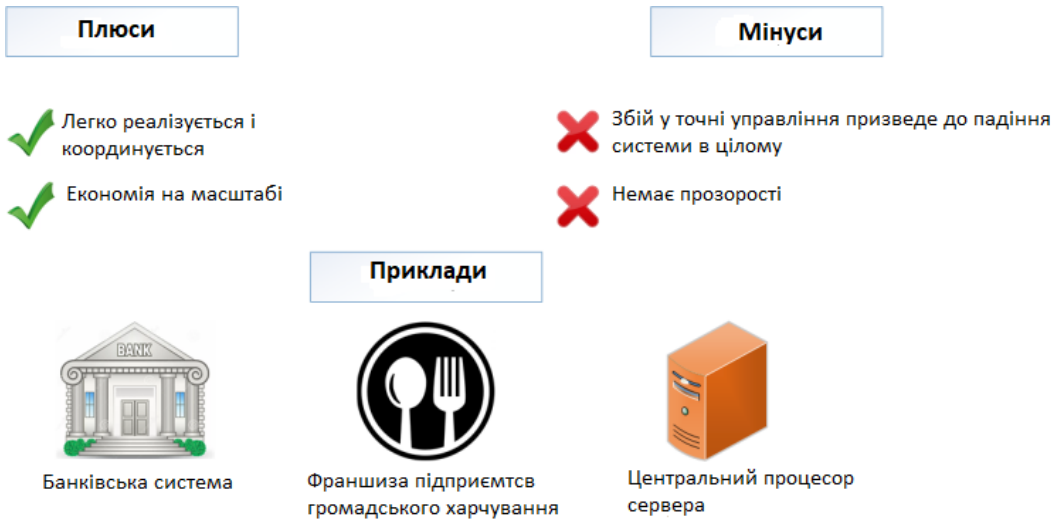
2. При великому масштабі, система позбавляє від необхідності подвійної роботи, яка іноді робиться при наявності декількох точок управління. Так як в системі тільки одна точка управління, не потрібно змушувати безліч точок виконувати одні і ті ж функції, що дає економію при великих масштабах системи.

Мінуси централізованої системи:

1. Залежність від однієї точки управління. Наявність лише однієї точки управління робить систему уразливою, так як будь-яка атака на цю єдину точку управління дестабілізує всю систему. Нескладно уявити собі ситуацію з сервером, коли атакується єдине джерело інформації і немає ніяких резервних копій, - інформація в цьому випадку буде втрачена.

2. Система з однією точкою управління бюрократична за своєю суттю, що додає в неї безліч шарів та ієрархій.

3. Ця система непрозора і тому має схильність до шахрайства.



Приклади централізованих систем: банківські системи; франшизи підприємств громадського харчування ("McDonalds"); центральний процесор сервера.

Децентралізовані системи

Децентралізовані системи - системи, в яких існує кілька точок управління і повноваження диверсифіковані (рис. 1.2). Це робить систему менш чутливою до збоїв, так як вихід з ладу однієї точки управління не призведе до падіння всієї системи. Ієрархія такої системи більше наближена до горизонтальної в порівнянні з централізованою системою[5].

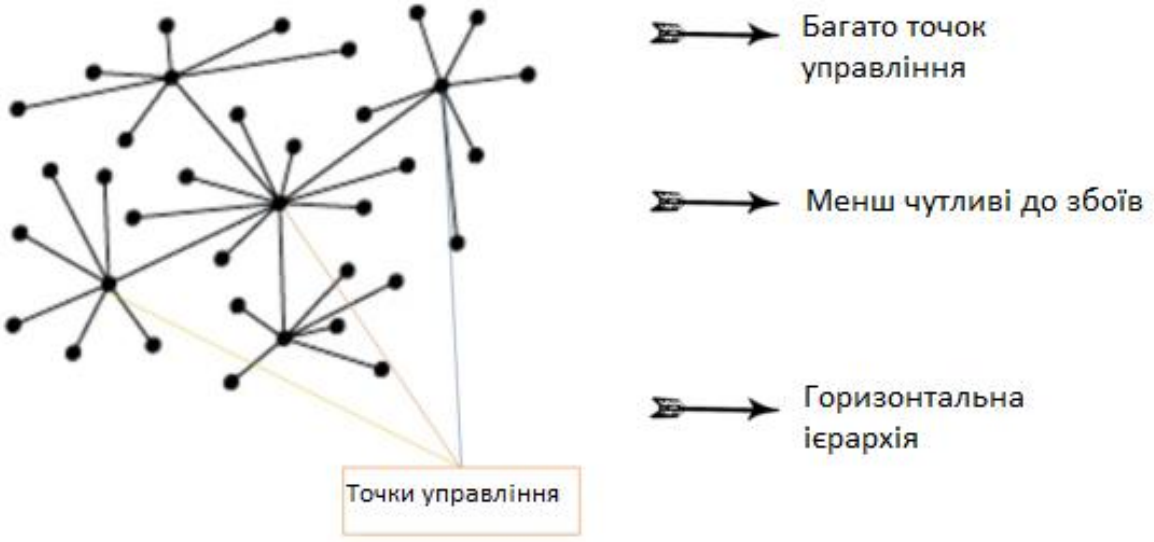


Рис. 1.2

Децентралізовані системи :

Плюси децентралізованої системи:

1. В даній системі рішення ухвалюються на рівні, більш наближеному до користувача. Таким чином, у органів (точок), які приймають рішення, набагато більше інформації про кінцевого користувача (якщо мова йде про продукт) або про людей (якщо мова йде про уряд).

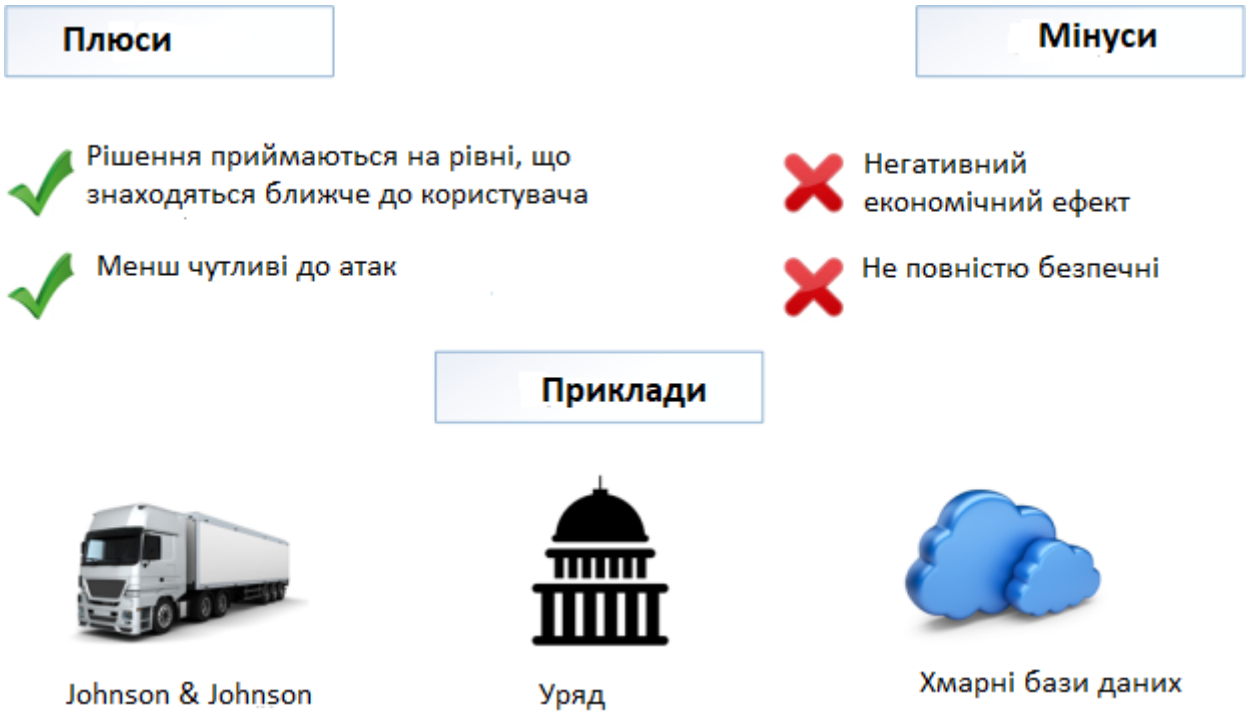
2. Ця система менше схильна до атак та збоїв, так як тепер в ній кілька точок управління. Збій в одній точці не призведе до дестабілізації всієї системи, як у випадку з централізованою системою.

Мінуси децентралізованої системи:

1. Негативний економічний ефект, пов'язаний зі збільшенням масштабів системи. У такій системі через збільшення кількості точок управління можна отримати «проблему дублювання завдань».

2. Незважаючи на те, що децентралізовані системи надійніше централізованих, вони все одно схильні до збоїв, тому їх не можна назвати повністю надійними.

Приклади децентралізованих систем: системи постачання, такі як "Johnson & Johnson"; уряду, де є центральні та місцеві органи влади; хмарні бази даних.



Розподілені системи

У розподілених системах будь-яка точка - це точка управління (рис. 1.3). Тому такі системи фактично несприйнятливі до падінь. Це не означає, що їх не будуть зламувати, однак, щоб вивести з ладу таку систему, зловмисник повинен взяти під контроль або змінити більше 50% точок управління. Витрати на те, щоб зробити подібне самостійно, зведуть нанівець більшу частину прибутку і зроблять економічно недоцільним спроби злому. Ієрархія таких систем повністю горизонтальна. Кожна точка управління дорівнює будь-якій іншій точці управління, і будь-який суб'єкт, будь-який учасник системи є точкою управління.[5] Отже, усі рівні, що і призводить до горизонтальної ієрархії.

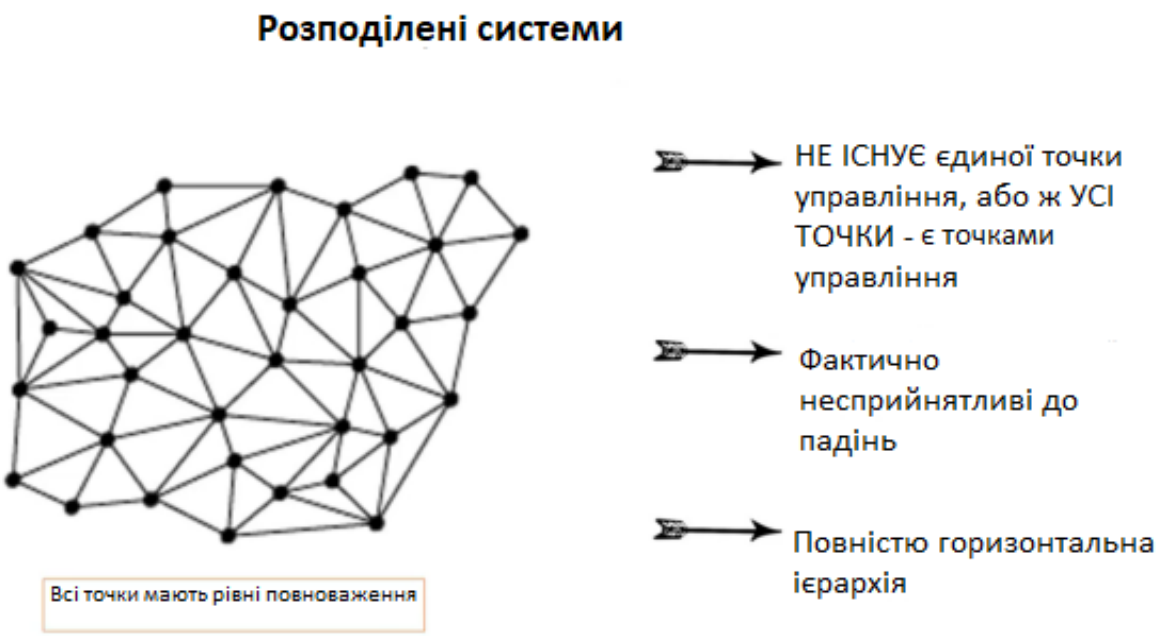


Рис. 1.3.

Розподілені системи :

Плюси розподіленої системи:

1. Відсутність необхідності посередників.

2. Цю систему економічно недоцільно зламувати, що робить її надзвичайно надійною і найбезпечнішою з трьох вище названих систем.

3. Розподілена система повністю прозора, завдяки чому вчинення шахрайства стає малоймовірним, так як це досить складно.

Мінуси централізованої системи:

1. Такі системи до цих пір вважаються новими, і їх технології знаходяться в процесі постійної розробки.

2. Для стабілізації таких систем буде потрібно багато часу і значна капіталізація.

Приклади розподілених систем: криптовалюта; мережі блокчейн.

Пояснивши сутність системи блокчейн і вже знаючи принцип її роботи, можна зробити певний підсумок. Що ж, блокчейн - це цифровий децентралізований реєстр, що дозволяє реєструвати транзакції без участі фінансового посередника, наприклад, банку. Наведемо деякі суттєві переваги блокчейну у порівнянні з існуючими платіжними системами:

- **Прозорість**
- **Зниження транзакційних витрат**
- **Децентралізація**

1.2. Концепція блокчейн

Структура даних blockchain - це упорядкований «назад» пов'язаний між собою список блоків транзакцій. Blockchain може зберігатися у будь-якому файлі або просто в базі даних. Клієнт Bitcoin Core зберігає метадані blockchain використовуючи БД LevelDB від Google [2]. Блоки пов'язані "назад", це означає що кожен посилається на попередній блок в ланцюзі. Blockchain часто візуалізується як вертикальна стопка, з блоками, що лежать один на одному і першим блоком, який є своєрідним фундаментом для блоків, які знаходяться вище. Подібна візуалізація у вигляді блоків, складених один на одного призводить до використання таких термінів, як "висота" для позначення відстані від першого блоку, і "вершина", яка вказує на нещодавно доданий блок.

Кожен блок в blockchain ідентифікується хешем, який генерується з використанням криптографічного алгоритму SHA256, застосованого до заголовка блоку. Кожен блок також посилається на попередній блок, відомий як батьківський блок, через поле "хеш попереднього блоку" в заголовок блоку. Іншими словами, кожен блок містить хеш свого батька всередині власного заголовка. Послідовність хешей, що зв'язують кожен блок з його батьком створює ланцюг, що тягнеться до самомого першого блоку з коли-небудь створених, відомому як блок генезису.
Хоча блок має тільки одного з батьків, він може тимчасово мати кілька дочірніх блоків. Кожен з дочірніх блоків посилається на один і той же батьківський блок і містить той же хеш в поле "хеш попереднього блоку".

Хеш дочірнього блоку змінюється, якщо змінюється хеш батьківського. Коли батьківський блок отримує будь-які зміни, змінюється його хеш. Змінений хеш батьківського блоку вимагає зміни посилання "хешу попереднього блоку" в дочірньому блоці. Це в свою чергу змінює хеш самого дочірнього блоку, яке, в свою чергу, змінює посилання у свого попереднього блоку, який, в свою чергу змінює хеш вже свого попереднього блоку, і так далі. Це каскадний ефект гарантує, що якщо за блоком було багато поколінь, він не може бути змінений без перерахунку всіх наступних блоків. Так як для подібного перерахунку потрібна величезна кількість обчислень. Довгий ланцюг блоків робить глибоку історію в блокчейні незмінної, що є ключем до безпеки біткоїна.

1.2.1. Структура блоку транзакцій

Базовою складовою блокчейна є блок. Блок являє собою структуру даних, контейнер, який об'єднує транзакції для включення в загальнодоступну бухгалтерську книгу, в blockchain. Блок складається із заголовка (Head), що містить метадані, далі за ним іде довгий список транзакцій(Payload), (Рис.1.4.), які займають більшу частину всього обсягу блоку. Розмір блоку займає 80 байт, в той час як середня транзакція займає не менше 250 байтів, а середньому блок містить понад 500 угод [7]. Відповідно, повністю заповнений транзакціями блок в 1000 разів більше заголовка.

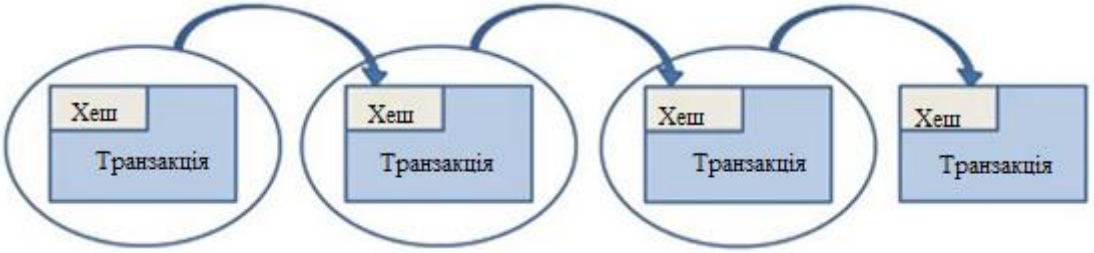


Рис. 1.4.

Ті блоки, які вже записані в блокчейн, змінити неможливо! Взагалі, будь-яке редагування в блокчейн інформації (транзакцій) заборонено. Можна тільки дописувати нові блоки. Це важлива властивість блокчейна, як розподіленого реєстру фінансових транзакцій.

При цьому блокчейн набагато надійніше будь-якої бухгалтерської книги або будь-якого банківського реєстру операцій, оскільки копії блокчейна, як вже зазначалось зберігаються на безлічі комп'ютерів (серверів).

1.2.2. Ідентифікатори блоку: хеш заголовка блоку

Заголовок блоку містить наступну інформацію:

- Версія блоку
- Дата і час створення блоку
- Хеш-код заголовка блоку
- Хеш-код попереднього блоку
- Хеш-код всіх транзакцій в блоці
- Спеціальні параметри nonce і bits, які записуються при Майнінгу

Сам хеш-код заголовка блоку - це і є те, що пов'язує попередній блок з подальшим в ланцюжку блокчейна. Хеш-код записується в наступний блок як хеш-код попереднього і так далі.

Також в заголовку зберігається хеш-код транзакцій поточного блоку. Він вираховується за допомогою алгоритму, відомого, як дерево Меркла (Merkle tree), він ще має назву бінарне дерево хешів. (Рис.1.5.)

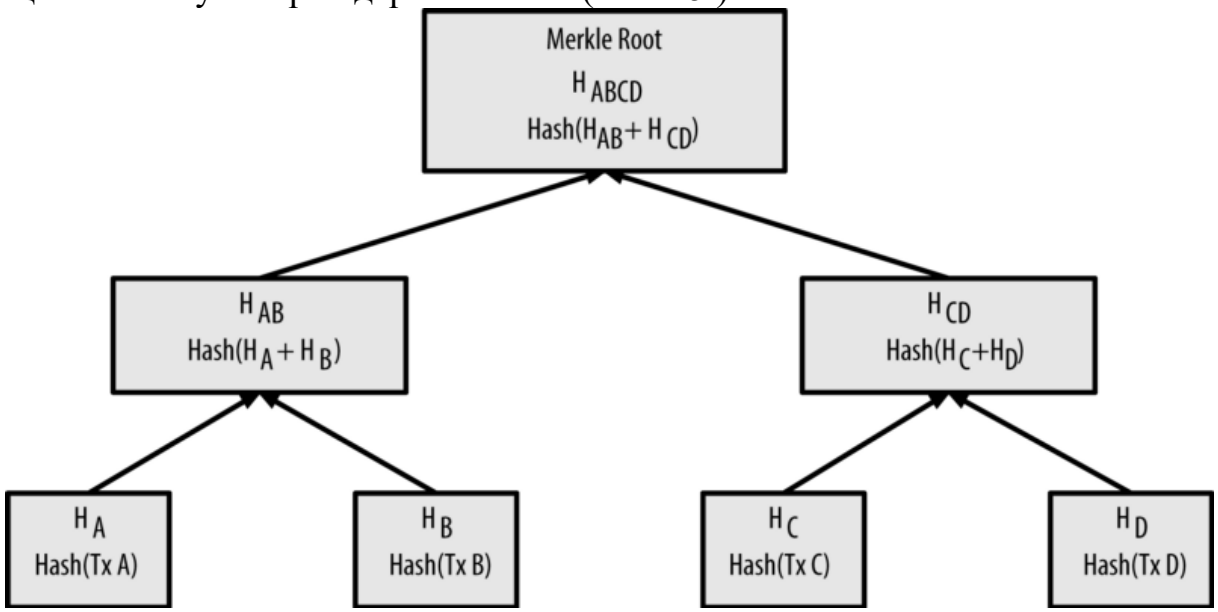


Рис. 1.5.

Працює це так:

1. Спочатку підраховуються хеші всіх транзакцій

2. Потім рахується сума від усіх хешей пар транзакцій
3. Далі рахуються хеші від суми отриманих пар хешей і так далі по тій же системі, поки не отрмаємо один єдиний хеш-код, який і буде хешем транзакцій в блоці.

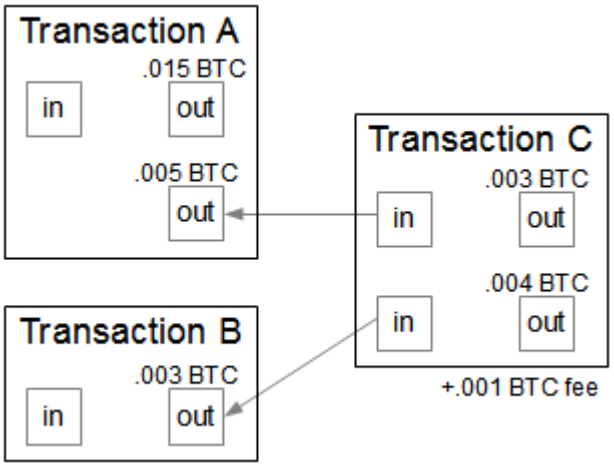
Саме заголовки дозволяють відслідковувати цілісність вмісту самих блоків.

Як вже говорилося раніше, в блокчейні біткоїна записуються транзакції. Власне сама транзакція має вигляд:

З **< address 1>** відправник **<N>** біткоїнів на **< address 2>**

Насправді транзакція, записана в блок блокчейна, дещо складніша, оскільки протокол біткоїна оперує такими поняттями, як Входи (Inputs або In) і Виходи (Outputs або Out) [11].

В цифровій валюті нові транзакції через Входи (один або кілька) посилаються на Виходи (один або кілька) попередніх транзакцій і формують Виходи (також один або кілька) для використання вже у наступних транзакціях.



Нова транзакція С має посилання на дві вхідні транзакції- А і В. У результаті на вході у транзакції виходить 0.008 BTC (0.005 + 0.003), які потім розділяються на два виходи - на першу адресу відправляється 0.003 BTC, а на другий 0.004 BTC. Залишок (0.001 BTC) - комісія Майнеру.

1.2.3. Блок Генезиса

Перший блок в blockchain називається блоком генезиса. Він був створений в 2009 році і є спільним предком всіх блоків в blockchain. Це означає, що якщо ви виберете будь-який блок і простежте ланцюжок назад в часі, то в кінцевому підсумку прийдете до блока генеза. Кожен вузол завжди починається з blockchain щонайменше з одного блоку так, як блок генезису закодований статично в клієнтському ПО, це означає що він не може бути модифікований. Кожен вузол

завжди "знає" хеш блоку генезису і його структуру, фіксований час, коли він був створений, і навіть єдину транзакцію у цьому блоці. Таким чином, кожен вузол має відправну точку для blockchain, безпечний "корінь", з якого можна побудувати надійний blockchain. Блоку генезису відповідає наступний ідентифікує хеш [2]:

```
000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f
```

Рис.1.6.

Використовуючи базовий клієнт Bitcoin Core з комомандним рядком Windows:

```
$ bitcoind getblock 000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f
```


```
{
    "hash" : "000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f",
    "confirmations" : 308321,
    "size" : 285,
    "height" : 0,
    "version" : 1,
    "merkleroot" : "4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b",
    "tx" : [
        "4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b"
    ],
    "time" : 1231006505,
    "nonce" : 2083236893,
    "bits" : "1d00ffff",
    "difficulty" : 1.00000000,
    "nextblockhash" : "00000000839a8e6886ab5951d76f411475428afc90947ee320161bbf18eb6048"
}
```


Блок генезиса містить приховане повідомлення всередині себе. Вхід coinbase-транзакції містить текст "Таймс 03 / Січ / 2009 Міністр економіки на межі другого раунду у порятунку банків." Перший блок – є символічним, так як має посилання на заголовок британської газети The Times. Він служить посилом важливості створення незалежної монетарної системи, разом із запуском біткоїна в той час, коли в усьому світі безпрецедентна валютна криза. Повідомлення було закладено в перший блок Сатоши Накамото, творцем біткоїна.

1.2.4. Ланцюжок блоків в блокчейн

Вузли містять локальну копію блокчейн, починаючи з блоку генезиса. Локальна копія blockchain постійно оновлюється і поповнюється з мірою появи нових блоків. Як тільки вузол приймає вхідні блоки з мережі, він перевіряє їх, потім їх зв’язує з існуючим ланцюжком блоків на підставі "хешу попереднього блоку".

Припустимо, наприклад, що вузол містить 277,314 блоків в локальній копії blockchain. Останній блок, про який вузол знає - це 277,314 має хеш заголовка 00000000000000027e7ba6fe7bad39faf3b5a83daed765f05f7d1b71a1632249 [2].

Потім біткоїни-вузол отримує новий блок з мережі, який він проаналізує наступним чином:

```
{
    "size" : 43560,
    "version" : 2,
    "previousblockhash" :
        "00000000000000027e7ba6fe7bad39faf3b5a83daed765f05f7d1b71a1632249",
    "merkleroot" :
        "5e049f4030e0ab2debb92378f53c0a6e09548aea083f3ab25e1d94ea1155e29d",
    "time" : 1388185038,
    "difficulty" : 1180923195.25802612,
    "nonce" : 4215469401,
    "tx" : [
        "257e7497fb8bc68421eb2c7b699dbab234831600e7352f0d9e6522c7cf3f6c77",

 # [...   пропущено багато інших транзакцій    ...]

        "05cfd38f6ae6aa83674cc99e4d75a1458c165b7ab84725eda41d018a09176634"
    ]
}
```

Вузол «дивиться» в вміст поля previousblockhash цього нового блоку, в якому міститься хеш «батьківського» блоку. Цей хеш вже відомий вузлу як хеш останнього блоку в ланцюжку на висоті 277314. Таким чином, цей новий блок є дочірнім останнього блоку в ланцюжку і розширює поточний blockchain. Вузол додає цей новий блок в кінець ланцюжка, що робить blockchain довше з новою висотою 277315 [1].

1.3. Приклади використання технології блокчейн на практиці

Ми розглянули концепцію технології блокчейн. Як дана технологія реалізована, які має властивості, та які переваги має у порівнянні з іншими технологіями.

В даному розділі, буде дана відповідь на питання : «Чи є блокчейн революціною технологією саме тільки для криптовалют?» Відразу відповідь – ні. Що ж, розглянемо приклади застосування технології блокчейн у різних сферах.

Блокчейн стрімко розвивається в безлічі інших областей, які прагнуть скоротити витрати, підвищити ефективність і відмовитися від застарілих інформаційних інфраструктур. Таких як:

- ***Управління ідентифікаційної інформацією***
- ***Цифрові активи і токенізація***
- ***Міжнародні платежі***
- ***Захист авторського права***
- ***Смарт-контракти***
- ***Електронне голосування***

1.5. Висновок

Таким чином, у розділі 1. Blockchain, ми провели дослідження і дали відповідь на питання, що являє собою технологія блокчейн, які криптографічні алгоритми вона використовує задля безпеки, автономності й надійності. Задопомогою командного рядку Windows та клієнстького Bitcoin Core згенерували й продемонстрували перший блок, блок Генезиса, системи блокчейн біткоїна. Провели аналіз ланцюжків блоків. Навели актульні компанії які використовують технологію блокчейн у своїх проектах.

Розділ 2.Bitcoin

Біткоїн - це набір концептів і технологій, які спільно утворюють основу для екосистеми цифрових грошей.

Різні автори по-різному класифікують біткоїн. Найчастіше зустрічаються варіанти: криптовалюта, цифрова валюта, віртуальна валюта, електронна готівка.

Біткоїни, використовуються задля зберіганя та їх передачі між учасниками мережі. Користувачі біткоїнів комунікують один з одним за допомогою протоколу біткоїн через мережу Інтернет, однак, протокол може використовуватися і всередині будь-якої іншої мережі. Стек протоколу біткоїна доступний у вигляді ПО з відкритим вихідним кодом і може бути запущений на різних пристроях, включаючи ноутбуки і смартфони, що робить технологію легко доступною.

Користувачі мають можливість використовувати біткоїни в мережі так само, як це відбувається зі звичайними валютами: купувати і продавати товари, відправляти гроші людям або організаціям. Біткоїни можуть бути куплені, продані або обмінені на інші валюти на спеціалізованих біржах. В певному сенсі, біткоїни є ідеальної грошової технологією для інтернету, так як він швидкий, безпечний і не має меж[9].

На відміну від традиційних валют, біткоїни повністю віртуальні. Не існує фізичних монет або навіть монет в цифровому форматі. Користувачі біткоїнів володіють ключами, які дозволяють довести права володіння в транзакціях біткоїн-мережі, дозволяючи витрачати кошти і відправляти їх новому власнику. Ці ключі часто зберігаються в цифрових гаманцях на комп'ютерах кожного користувача.

Біткоїн - це розподілена одноранговая (P2P) мережа, що означає, що в ній не існує "центрального" сервера або контрольного вузла [2]. Біткоїни виникають в процесі "Майнінга": конкурентного рішення математичної задачі. Будь-який учасник мережі біткоїн (тобто хто завгодно, із запущеним повним стеком протоколу) може бути Майнером, тобто використовувати обчислювальні потужності свого комп'ютера для підтвердження транзакцій. В середньому, кожні 10 хвилин, хтось підтверджує транзакції за попередні 10 хвилин і винагороджується за це новоствореними біткоїном. По суті, "Майнінг" децентралізована емітацію валюти і клірингова функція центрального банку.

2.1. Види платіжних систем. Причини появи концепції криптовалюти

Вище, були коротко наведені означення цифрової валюти – біткоїна. Але перед тим, як зрозуміти її рівень значущості, її революційність, нам слід розібратись в концепції грошей, згадати їх історію, які існували грошові системи до появи біткоїна, проаналізувати їх і на основі цього навести їх позитивні і негативні сторони.

Саме після цього стане зрозуміло, що призвело до ідеї Bitcoin і чому все ж таки, цю платіжну систему називають революційним з усіх коли-небудь створених.

Бартерна платіжна система

Бартерну систему люди використовували 6 тисяч років до нашої ери. Потім з'явилися монети - приблизно в 650 році до н. е. Люди почали застосовувати перші форми друкованої валюти приблизно в 960 році, а перша форма чека з'явилася приблизно в 1717 році н. е. Вперше ідея "електронних грошей" була запропонована американським фахівцем з теорії складності Девідом Чоумом ще в кінці 70-х років. Перша система електронних грошових коштів PayPal була представлена в 1998 році, в Росії - WebMoney також з'явилася в 1998 році. І, нарешті, в 2008 році з'явилася ідея біткоіни [3].

Система бартеру - перший засіб обміну, відомій людині, - в своєму зародженні була досить простою і досить логічною: будь-яка людина виробляє той товар або послугу, на якій він спеціалізувався, і обмінював свій товар або послугу на ті, які йому були потрібні. Ця система відмінно працювала деякий час, але в ній були присутні недоліки, про які стало відомо пізніше. Розглянемо приклад, щоб це зрозуміти. представимо дві особи - А і Б; А виробляє пшеницю, а у Б є стадо кіз. А та Б вирішили обмінятися - 10 центнерів пшениці за одну козу. В даний конкретний момент А і Б вирішили, що одна коза по вартості дорівнює 10 центнерів пшениці. Однак завтра це може змінитися, і пшениця особи А може не коштувати стільки ж у третьої особи. Припустимо, особі А потрібні банани, він шукає того, хто їх виробляє, і тепер йому доведеться пропонувати різну загальну вартість кожного разу, коли він хоче придбати різні товари. Тому першим і головним недоліком системи бартеру була відсутність загальної одиниці вартості.

Наступний недолік полягав у тому, що товари легко псуються. Кози, пшениця - все це продукти, піддані псування. У них короткий термін життя. Однак, якщо сьогодні є пшениця, а потрібна коза, але через рік, то пшениця не доживе до цього моменту. Тому в даній системі завжди існував розрив у часі між попитом і пропозицією.

Наступним недоліком була неефективна система перевезення товарів. Наприклад, перевезення великих товарів завжди ставала проблемним моментом.

І, нарешті, в бартерній платіжній системі не забезпечувалася безпека - товари могли бути вкрадені і використані кимось іншим, і у власника цих товарів не було докази того, ресурси належать йому.

Монетна платіжна система

Проте люди швидко зрозуміли, що товаром-посередником може служити золото, яке цінувалося усіма і прекрасно вирішило одну з основних проблем

бартерної системи - відсутність одиниці загальної вартості. До того ж золоті монети не швидко псуються і інфляції. Золоті запаси в світі завжди обмежені, а це значить, що попит на золото завжди буде підвищуватися, в той час як пропозиція завжди буде обмежена. Це робить будь-які ресурси не піддаються інфляції. Їх вартість буде тільки зростати і ніколи не буде падати. Нарешті, золото було легко транспортувати. Монети були дрібними за розміром і найчастіше однакової форми. Людям більше не доводилося перевозити два різних за розміром товару.

Однак у золотих монет були і мінуси, через які люди перестали їх використовувати. Перший мінус: видобуток золота - витратна і нелегка справа. Другий мінус: незважаючи на те, що золоті монети були порівняно дрібними за розміром і їх легко було перевозити, порівняння з сучасними паперовими грошима вони не витримували. І, нарешті, в системі платежів золотими монетами також не забезпечувалася безпека: їх міг вкрасти і використовувати хто завгодно. Це означає, що дана система не вирішувала саму основну проблему бартеру - не забезпечувала безпеку[13].

Отже, виникла необхідність вирішити питання безпеки платіжної системи з використанням золотих монет, тому люди стали віддавати монети на зберігання третій стороні, яка в наші дні відома як банки. В обмін на це золото банки випускали боргові розписки, які підтверджували отримання золота. Будь-яка людина, що має таку розписку, міг де завгодно обміняти її на товари. Таким чином, боргові розписки були нічим іншим як першою формою паперових грошей. Однак їхні переваги спиралися на фактично існуюча кількість золота.

Паперова платіжна система

Існувало безліч плюсів паперової платіжної системи. По-перше, такі гроші були легше золотих монет і їх було простіше виробляти. По-друге, ці гроші підкріплювалися чимось: на перших порах - фізично існуючим цінним ресурсом - золотом, а пізніше - недосяжним ресурсом, таким як довіру уряду, яке їх випускало.

Однак і у цієї системи є свої недоліки. І один з них полягає в тому, що в системі паперових грошей не забезпечується безпека. Банкноти можна вкрасти і використовувати. Другий недолік - паперові гроші схильні до інфляції. Отже, паперові гроші - це валюта, підкріплена нематеріальним товаром, наприклад, довірою уряду. Але через те, що якогось відчутного ресурсу, підкріплює банкноти, не існує, уряд може випустити стільки паперових грошей, скільки можливо. Але через таких ринкових сил, як попит і пропозиція, більшу кількість випущених банкнот з часом втрачає свою вартість.

Ще один недолік паперової платіжної системи - це так звані "брудні" гроші і тіньова економіка. У той час не було способу відстежити всі грошові потоки, визначити, на які цілі використовуються гроші, і хто ними володіє.

І, нарешті, з золотомонетной системою в 1970-х було покінчено, тому що вона знижувала гнучкість центрального банку в здійсненні кредитно-грошової і фіскальної політик, потрібних для регулювання економіки країни.

Чекова платіжна система

Чекова платіжна система і система біткоіни мають багато спільного. По-перше, в чекової платіжній системі нарешті покращилася ситуація з безпекою в засобах обміну: на чеках вказувалося ім'я чекодавця, ім'я чекодержателя, а також їх адреси. Цей момент має дуже важливе значення для розуміння системи біткоіни. По-друге, в системі чеків використовувалася базова форма криптографії, іменована "підписом", щоб підтвердити, що саме чекодавець доручає провести операцію по чеку (ще одна важлива властивість з точки зору системи біткоіни). І, нарешті, користувач міг поміняти суму на чеку - люди більше не спиралися на певну суму, як у випадку з паперовими грошима[6].

Недоліки чекової платіжної системи пов'язані з появою форми чеків під назвою "чек на пред'явника" - той, хто пред'являв даний чек, міг отримати суму, зазначену в ньому. Це знижувало рівень безпеки. По-друге, та форма криптографії, яка використовувалася в чекової платіжній системі, була надто елементарної, і підписи можна було легко підробити. Нарешті, найбільшим недоліком чекової платіжної системи було час верифікації переказу грошових коштів. Верифікація відбувалася з великою затримкою. Чекодавець вручав чекодержателю чек, чекодержатель відправлявся в свій банк в будь-який зручний для нього час і передавав чек банку. Банк чекодержателя зв'язувався з роздільною інстанцією, яка, в свою чергу, пов'язана з банком чекодавця, щоб підтвердити, що у чекодавця є на рахунку необхідна сума і операція з переказу коштів може відбутися. Проблема ж полягала в тому, що під час переказу грошей чекодержатель був змушений віддавати чекодавцеві товар в обмін на його чек, не маючи при цьому ніяких гарантій того, що у чекодавця є на рахунку сума для завершення угоди.

Платіжна система електронних гаманців

На перший погляд ця система здається ідеальною - вона не вимагає наявності "живих" грошей, людям більше не потрібно носити з собою гаманець. З розвитком індустрії смартфонів і Інтернету і підвищенням рівня застосовуваної криптографічного захисту безпеку електронних гаманців стала набагато вище. В наші дні практично в будь-якому смартфоні є сканер сітківки і сканер відбитків пальців, що може служити засобом підтвердження банківських операцій, а це набагато більш технічно просунуті способи, ніж підпис. Завдяки швидкості, ефективності та зручності цих способів верифікація операції займає кілька секунд, що вирішує найбільшу проблему системи платежів банківськими чеками. Нарешті, можливість обліку операцій за допомогою електронних гаманців у фінансовій системі набагато вище. Електронні гаманці дозволяють регулюючим органам відслідковувати грошові потоки, контролювати, скільки у кого грошей і для чого ці

гроші використовуються. Ця система дає можливість знизити кількість "брудних" грошей і грошей, що використовуються в протизаконних цілях[5].

Отже, з першого погляду система електронних гаманців здається ідеальною. Але виникає питання: чому виникла необхідність в альтернативній системі - нової, вдосконаленої системи під назвою біткоіни? Тому що головна проблема системи електронних гаманців - це те, що не очевидно навіть зараз: ця система є частиною існуючої фінансової інфраструктури. У 2008 році світ побачив недоліки і недоліки існуючої фінансової інфраструктури. Виявилося, що діюча фінансова система неефективна і корумпована. Відбулася криза субстандартного іпотечного кредитування, через якого фінансові ринки обвалилися за принципом доміно, великі інвестиційні банки стали банкрутами, урядам довелося надати фінансову допомогу багатьом банкам. До того ж на світових фінансових ринках був зареєстрований найбільший з 1929 року спад ділової активності [6]. Але це були економічні наслідки, а існували ще й емоційні - люди втратили віру в фінансову систему. Вся суть банків полягала в тому, що вони були третьою стороною, яка надійно зберігає матеріальні цінності і є кращою альтернативою, ніж зберігання грошей вдома. . Люди довіряли банкам збереження своїх коштів, але банки не впоралися, - вони неправильно використовували ці гроші і втратили все. Тоді люди зрозуміли, що альтернативної та надійної системи зберігання грошей не існує. Почався масовий вивід коштів з банківських рахунків, і стало зрозуміло, що заміни існуючої системи немає. Люди опинилися перед вибором: або зберігати всі гроші вдома, що в сьогоднішніх обставинах не найкраща ідея, або віддавати їх банку, якому вони більше не довіряли. Виникла потреба в альтернативній системі. В кінці 2008 року особа або група осіб під псевдонімом Сатоси Накамото опублікував документ під назвою "біткоіни: одноранговая валютна система". Цей революційний документ кидав виклик самій концепції грошей і всієї ідеї існування посередника, який займається грошима. В цьому і полягає причина того, що біткоіни - це найбільш продумана і підриває все підвалини технологія з усіх коли-небудь створених.

2.2. Bitcoin – цифрова валюта та її властивості

Біткоїн - це перша в світі децентралізована цифрова криптовалюта, з'явився Bitcoin (BTC) в 2009 році. Дана валюта створена у протиріч всім раніше створеним еликтронним валютам та платіжних систем. Вона не прив'язана ні до яких фізичних активів або «офіційних» валют, а ціна цифрової монети біткоїна, регулюється виключно ринковим попитом і пропозицією. У біткоїнів є одна дуже значна схожість з золотом - обмеженість в кількості / запасу. Bitcoin строго обмежений у кількості 21 000 000 [13].

Як наслідок біткоіни є всесвітньою платіжною системою без емісії та інфляції, через яку можна проводити операції з даної валютою. Головна перевага у порівнянні до традиційних платіжних систем в тому, що система біткоіни не має ніякого керуючого і процесингового центру - всі операції відбуваються виключно

в мережі рівноправних клієнтів.

Біткойн - перша і найбільша децентралізована криптовалюта. Існують сотні інших альткойнов (альтернативних криптовалюта), наприклад Litecoin або Dogecoin, але на біткойн припадає близько 90% ринкової капіталізації всіх криптовалюта, і він став фактичним стандартом. Біткойни використовуються псевдонімного (а не анонімно).

У термінах легко заплутатися, тому що слова« біткойн »і« блокчейн »можуть позначати будь-яку з трьох частин концепції: базову блокчейн-технологію, протокол і клієнта, щоб забезпечити виконання транзакцій, і власне криптовалюта (гроші). Крім того, ці терміни можуть застосовуватися для позначення і концепції криптовалюта. Це все одно що називати терміном «PayPal» сам інтернет, через який працює протокол PayPal, службовець для перекладу валюти PayPal. У блокчейн-індустрії ці терміни часто змішуються, оскільки поки не завершився процес формування загальновизнаного багаторівневого стека технологій.

2.2.1. Стек технологій: блокчейн, протокол, валюта

Термін «біткоїн» може ввести в оману, оскільки під біткоїном прийнято вважати три різні речі.

По-перше, біткоїн - це базова платформа блокчейн-технології.

По-друге, так як біткоїн працює на основі даної базової технології, його також називуйть й протоколом, що описує, як саме відбуваються операції над активам в ланцюжку блоків.

По-третє, біткоїн - це цифрова криптовалюта, найперша і найпопулярніша з відомих на сьогодні криптовалют.

У таблиці 1-1 показано, чим відрізняються ці поняття.

Нижній рівень - це базова блокчейн-технологія. Блокчейн як ланцюжок блоків транзакцій - це розподілений, загальнодоступний і спільно-використований усіма вузлами мережі реєстр або журнал записів, що містить дані про транзакції. Журнал оновлюється Майнером і відстежується всіма бажаючими, але при цьому його ніхто не контролює. Він подібний до гігантської загальнодоступної таблиці, яка періодично оновлюється і підтверджує унікальність цифрових операцій переказу грошових коштів.

Середнім рівнем стека є протокол - пакет программ, який переводить кошти шляхом внесення транзакцій в блокчейн (журнал записів). Нарешті, третій рівень - це сама валюта, в транзакціях і на біржах використовується позначення BTC або Btc. Серед сотні криптовалюта біткойн - не тільки найперша, але і найпопулярніша. Серед інших слід відзначити Litecoin, Dogecoin, Ripple, NXT, і Peercoin [2].

Криптовалюта	Біткоїн (BTC), Litecoin, Dogecoin
Біткоїн-протокол і клієнт	Програма, виконуюча певні операції
Блокчейн біткоїна	Базовий децентралізований журнал записів

Таблиця 1-1. Рівні стека блокчейн-технологій на прикладі біткоїна

Важливо розуміти, що загальна структура будь-якої сучасної кріптовалютної системи формується всіма трьома рівнями (блокчейн, протокол і валюта). Кожна монета являє собою одночасно валюту і протокол, вона може мати власний розподілений журнал записів або використовувати розподілений блокчейн біткоїнів. Наприклад, криптовалюта Litecoin використовує Litecoin-протокол, який працює з блокчейном Litecoin, - по суті, це клон біткоїна, в якому дещо змінені деякі функції.

Окремий блокчейн означає, що у монети є власний децентралізований журнал записів з такою ж структурою і форматом, що і розподілений журнал записів біткоїнів.

Подвійні витрати та задача візантійських генералів

Навіть якщо не брати до уваги потенціал використання біткоїнів і блокчейн-технології, біткоїн, безумовно, є серйозним фундаментальним проривом в галузі інформатики - результатом 20 років досліджень в області цифрових валют і 40 років досліджень в області криптографії, над якими працювали тисячі вчених усього світу [4]. Біткоїн став рішенням давньої проблеми фіатних грошей - проблеми подвійних витрат (double-spend problem). До появи криптографії блокчейна цифрову готівку (digital cash), як і будь-який інший цифровий актив, можна було нескінченно копіювати - як, наприклад, ми можемо сьогодні незліченну кількість разів копіювати різні файли в електронній пошті. При цьому без спеціального посередника неможливо було підтвердити, що та чи інша частина грошей не була вже витрачена раніше. Функцію посередника виконувала довірена третя сторона: банк або платіжна система на зразок PayPal, яка зберігала журнал записів, що гарантує, що кожна одиниця цифрових грошей може бути витрачена тільки один раз, тим самим запобігаючи подвійне витрачання.

Проблема подвійних витрат аналогічна давно сформульоваій математичній проблемі - так званої «Задачі візантійських генералів», суть якої полягає в тому, що кілька генералів перед боєм, не довіряючи один одному, повинні якось узгодити свої дії.

Блокчейн вирішує проблему подвійних витрат, об'єднуючи технологію однорангового обміну файлами BitTorrent і шифрування з відкритим ключем, тим самим створюючи новий вид цифрових грошей. Власність на монети реєструється у відкритому журналі записів і підтверджується криптографічними протоколами і співтовариством Майнерів. Блокчейн не вимагає довіри в тому сенсі, що в процесі транзакції користувачеві немає потреби довіряти контрагенту або посереднику.

Необхідно лише довіряти системі - програмної реалізації блокчейн-протоколу[1].

2.2.2. Принцип роботи Bitcoin

Біткоїн - це цифрові гроші, за допомогою яких можна купувати і продавати товари через інтернет. Ланцюжок доданої вартості біткоїнів формується кількома групами: розробниками, Майнерами, біржами, сервісами обробки платежів, операторами інтернет-гаманців і кінцевими користувачами / споживачами. Для початку роботи з криптовалютою «користувачу потрібно лише біткойн-адреса, секретний ключ і програма-гаманець. Біткоїн-адреса - це ідентифікатор на зразок номера рахунку, на який інші користувачі можуть відправляти біткоїни, а секретний ключ - це криптографічний ключ, за допомогою якого можна відправляти отримані біткоїни іншим користувачам. Для того щоб оперувати біткоїнами, програма-гаманець встановлюється на комп'ютері або смартфоні (Рис. 1.7). При цьому не потрібно відкривати ніякого «розрахункового рахунку» у будь-якої компанії або банку - після установки програма автоматично генерує зв'язку з секретного ключа і біткоїн-адреси, і ви можете відразу ж розпоряджатися коштами, прив'язаними до даною адресою. Гаманець може містити копію блокчейна - записи всіх транзакцій, коли-небудь виконаних з даної валютою. Це дозволяє самостійно верифікувати буд-які транзакції в рамках децентралізованої системи Біткоїн.

Рис.1.7 – Додаток – електнонний біткоїн-гаманець

Блокчейн вже зайняв нішу «валюти інтернету», став глобальною цифровою платіжною системою і має потенціал розвинутися в цілий «інтернет грошей», який об'єднує фінанси так само, як «інтернет речей» об'єднує різні пристрої. Першою і найбільш очевидною областю застосування блокчейна стали грошові розрахунки. Сенс існування альтернативних систем грошових розрахунків виправданий вже одними тільки міркуваннями економії: зниження комісій за платежі кредитними картами в усьому світі з 3% хоча б до 1% стане величезною вигодою для економіки. Особливо це стосується міжнародного ринку грошових переказів об'ємом в 514 млрд доларів щорічно, де комісії за переклад можуть становити від 7% до 30% [2]. Крім того, блокчейн доставляє кошти негайно, користувачі не очікують перекладу

кілька днів. Використання біткойнов і інших криптовалюта може привести до повного перегляду уявлень про гроші, торгівлі і комерції. Біткойн - не просто поліпшена версія системи VISA, він дозволяє робити те, про що люди навіть не замислювалися, адже валюта і платежі - це лише перша область його застосування. Основна особливість грошових розрахунків на основі блокчейна полягає в тому, що вони дозволяють здійснювати будь-які угоди через інтернет без посередників. За допомогою альткойнов можна здійснювати грошові перекази і вести комерційну діяльність повністю децентралізованим, розподіленим і глобальним чином. Тому криптовалюта може стати відкритою програмованої мережею для децентралізованого обміну будь-якими ресурсами - навіть без урахування валюти і платежів.

2.3. Транзакції

Простими словами, транзакція зазначає мережі, що власник певної кількості біткоїнів уповноважив передачу певної кількості з них іншому власнику. Тепер новий власник може витрачати ці біткоїни шляхом створення іншої транзакції, яка в свою чергу дозволяє передачу вже іншому власнику, і так далі, по ланцюжку зміни власників. Транзакції аналогічні записам у звичайній бухгалтерської книзі прибутків та витрат. Простими словами, кожна транзакція містить один або кілька "входів", з яких надходять кошти. З іншого боку в транзакції є один або більше

"виходів", куди гроші надходять. Транзакція містить в собі докази володіння у вигляді цифрового підпису власників сум на кожному з входів. Цифрові підписи власників можуть бути незалежно перевірені ким завгодно. У термінах біткоїнів, "витратити" означає підписання транзакції, що переміщує цінність з будь-якої іншої попередньої транзакції новому власнику, ідентифікованому біткоїн-адресою.

Нагадаємо, що *Bitcoin-адреса* - ідентифікатор (номер рахунку), що починається з 1 або 3 і містить 26-35 буквено-цифрових латинських символів (крім 0, O, I). Адреса так само може бути представлена у вигляді QR-коду, є анонімна і не містить інформації про власника. Її можна згенерувати безкоштошктовно, використовуючи, наприклад, програмне забезпечення системи Bitcoin.

Приклад біткоїн-адреси [16]:

```
1njrRcKQtfjjLuQxFYCeMXcth77m5TAYo
```

Приклад транзакції біткоїнів з одним входом і одним виходом:
Дані:

```
Input:
Previous tx: f5d8ee39a430901c91a5917b9f2dc19d6d1a0e9cea205b009ca73dd04470b9a6
Index: 0
scriptSig: 304502206e21798a42fae0e854281abd38bacd1aeed3ee3738d9e1446618c4571d10
90db022100e2ac980643b0b82c0e88ffdfec6b64e3e6ba35e7ba5fdd7d5d6cc8d25c6b241501
```

```
Output:
Value: 5000000000
scriptPubKey: OP_DUP OP_HASH160 404371705fa9bd789a2fcd52d2c580b65d35549d
OP_EQUALVERIFY OP_CHECKSIG
```

Пояснимо послідовність дій:
Вхід в цій транзакції імпортує 50 BTC від виходу # 0 в транзакції f5d8 ..., а потім вихід відправляє 50 BTC на біткойнов адреса (виражений тут в шістнадцятковій системі - 4043 ...) [2]. Коли одержувач захоче витратити свої гроші, то він буде посилатися на вихід # 0 цієї транзакції для входу своєї власної транзакції.

Вхід
Вхід (input) - це посилання на вихід іншої транзакції. У транзакції часто буває кілька входів. Значення цих посилань підсумовуються, і загальна сума біткоїнів може бути використана у виході поточної транзакції. Previous tx - це хеш попередньої транзакції. Index - це певний вихід цієї транзакції. ScriptSig - це перша половина половина скрипта (докладніше про це - нижче).

Скрипт містить дві компоненти: підпис і публічний ключ. Public key належить користувачеві, який використовує виходи транзакції, і підтверджує те, що власник транзакції має право розпоряджатися сумою, що надходить із виходів. Інший компонент - це ECDSA-підпис хешу спрощеної версії транзакції. Об'єднаний з

публічним ключем, підпис підтверджує, що транзакція була створена реальним власником даного Bitcoin адреси.

Вихід
Вихід (output) містить інструкції щодо відправленню біткоїнів. Значення (value) - це кількість Сатоши (1 BTC = 100,000,000 Сатоши). ScriptPubKey - це друга половина скрипта.

Кожен вихід транзакції може бути використаний в якості входу для наступної транзакції тільки один раз, тому сума всіх входів для поточної транзакції повинна бути використана на її виходах.
Наприклад, якщо кількість введених біткоїнів дорівнює 50 BTC, а користувачеві потрібно відправити тільки 25 BTC, то біткоїн створить два виходи по 25 BTC кожен: один відправиться в пункт призначення, а інший відправиться ще раз власнику цих коштів (так звана "здача" - транзакція, яку користувач фактично відправляє сам собі).

Будь-яка сума входів біткоїнів, не використана в виходах, стає комміссією транзакції. Вона дістанеться тому, хто згенерує блок.

2.4. Майнінг

Раніше, ми неодноразово згадували термін «майнер», «майнінг». Що ж, в даному розділі ми розглянемо дані терміни більш детальніше.

Як ми вже знаємо транзакція поширюється по біткоїн-мережі. Але вона не стане записом в загальній бухгалтерській книзі (блокчейн) до тих пір, поки не буде перевірена і включена в блок, саме цей процес і має назву майнінг.

Майнінг (англ., "Mining") - це процес запису транзакцій в блокчейні. За мету Майнінга - є досягнення консенсусу між вузлами мережі щодо того, які транзакції вважати реальними [12].

Крім того, Майнінг це єдиний спосіб емісії біткоїнів, які нараховуються в якості винагороди за рішення Майнеру певних математичних задач за допомогою комп'ютерного обладнання.

Кожен блок повинен містити підтвердження того, що робота по вирішенню певних математичних задач була виконана, і кожен із нод мережі може легко перевірити, чи дійсно блок був закритий по певних правилах. Емісія цифрових валют відбувається в якості винагороди за добування децентралізовано, що можемо цілковито стверджувати, про відсутність контролю над випуском з боку єдиного центру [12].

Що ж, процес видобутку біткоїнів служить одночасно двом цілям:

• Майнінг створює нові біткоїни в кожному новому блоці, майже так само, як центральний банк друкує нові гроші.

• Майнінг створює довіру, гарантуючи, що для потрапляння транзакцій в блок потрібно досить обчислювальної потужності. Більше блоків означає більше обчислень, що, в свою чергу, означає більше довіри.

Випуск нових біткойнів децентралізований, не залежить від будь-якого регулюючого органу, обсяг емісії відомий заздалегідь (Рис 1.8).

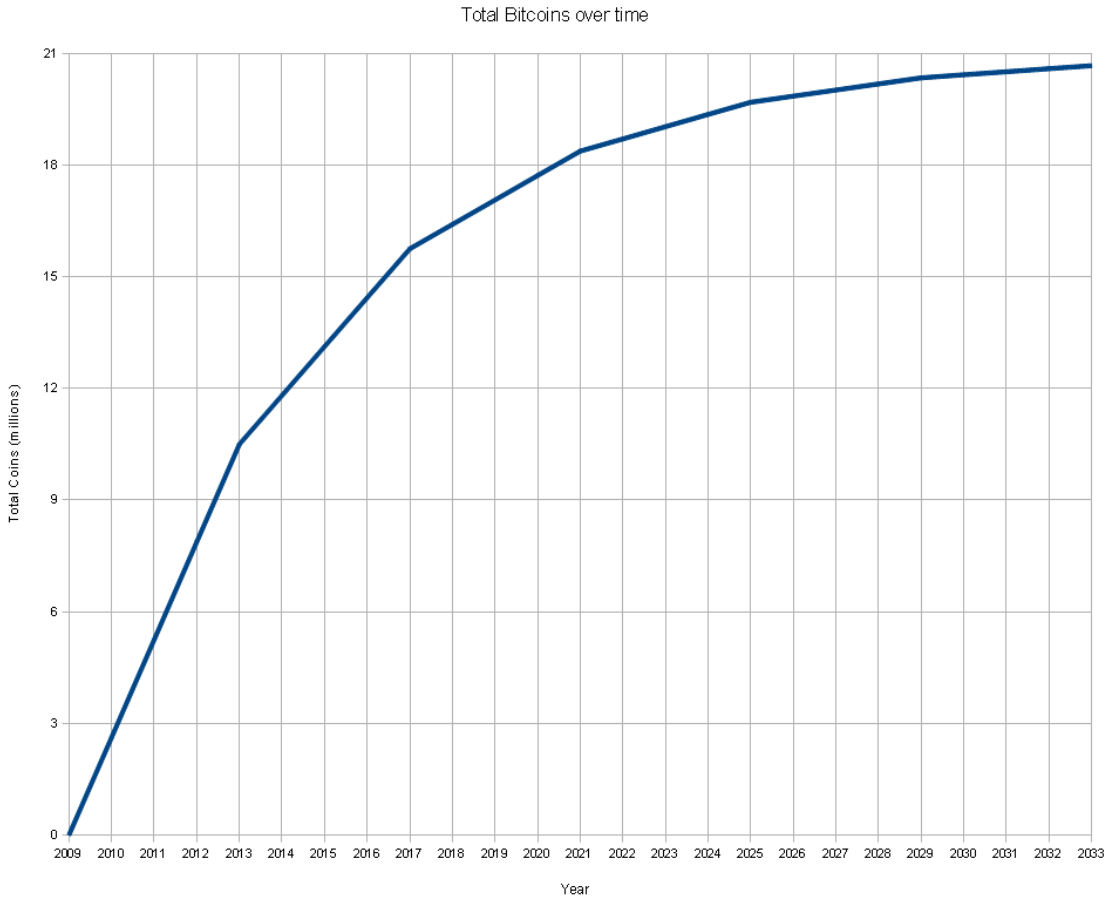


Рис.1.8. - Графік кількості біткойнів до 2033 року

Спочатку розмір емісії при створенні блоку становив 50 біткойнів. 28 листопада 2012 відбулося перше зменшення емісійної нагороди з 50 до 25 біткойнів. 9 липня 2016 року стався другий зменшення емісійної нагороди з 25 до 12,5 біткойнів. У 2031 році розмір емісії при створенні блоку складе менше одного біткойнів і продовжить прагнути до нуля. Передбачається, що емісія зупиниться в 2140 році, оскільки нагорода за блок не зможе перевищувати 10-8 BTC, однак задовго до цього поступово основним джерелом винагороди за формування нових блоків стануть комісійні збори [3].

2.5. Висновок

У цьому розділі був проведений аналіз криптовалюти Bitcoin. Виявлено основні переваги криптовалюти, які наведено нижче.

1. **Нульові або дуже низькі комісії**
2. **24/7/365**
3. **Миттєві перекази**
4. **Безпека, надійність, захищеність**
5. **Технологічне перевага**
6. **Нові можливості**
7. **Можливість відправляти мікротранзакції**
8. **Легкий старт**
9. **Легкий прийом і відправка пожертвувань**
10. **Наднаціональна система**
11. **Відкрита технологія**
12. **Ринкове ціноутворення**
13. **Можливість обійти будь-які санкції**
14. **Розширення покриття фінансовими послугами**
15. **Позбавлення від посередників, таких як банки-кореспонденти**
16. **Повний контроль над своїми засобами**
17. **Недоторканність приватного життя**
18. **Усунення непрозорості**
19. **Контроль там, де необхідно**
20. **Можливості блокчейна**

Розділ 3. Математика криптовалют

У даному розділі ми розглянемо симетричну та асиметричну криптографію і науку, яка лежить в основі криптовалют.

Криптовалюта використовує однорангову децентралізовану систему для проведення своїх транзакцій. Оскільки весь процес проходить в онлайні, є побоювання, що транзакції можуть перериватися або бути зламуваними хакерами. Тому, ми розповімо, як криптовалюта використовує криптографію задля безпеки своїх транзакцій [10].

Криптографія - це метод використання передових математичних принципів для зберігання і передачі даних в певному формі, так що тільки ті, для яких вони призначені, можуть читати і обробляти інформацію. Криптографія використовується вже тисячі років для таємного спілкування. Найперше використання криптографії було зафіксовано в гробниці, знайденої в Стародавньому царстві в Єгипті близько 1900 року до нашої ери.

Шифрування - один з найбільш важливих інструментів, що використовується в криптографії. Це засіб, за допомогою якого повідомлення перетворюється в закодований набір символів. Тільки відправник і одержувач знають, що приховано в листі.

У сучасних технологіях широко використовуються три форми шифрування:

- Симетрична криптографія
- Асиметрична криптографія
- Хешування

3.1. Симетричне шифрування

Симетричне шифрування – спосіб шифрування, в якому для шифрування і дешифрування застосовується один і той же криптографічний ключ. Ключ алгоритму повинен зберігатися в секреті обома сторонами. До появи схеми асиметричного шифрування єдиним існуючим способом було симетричне шифрування.

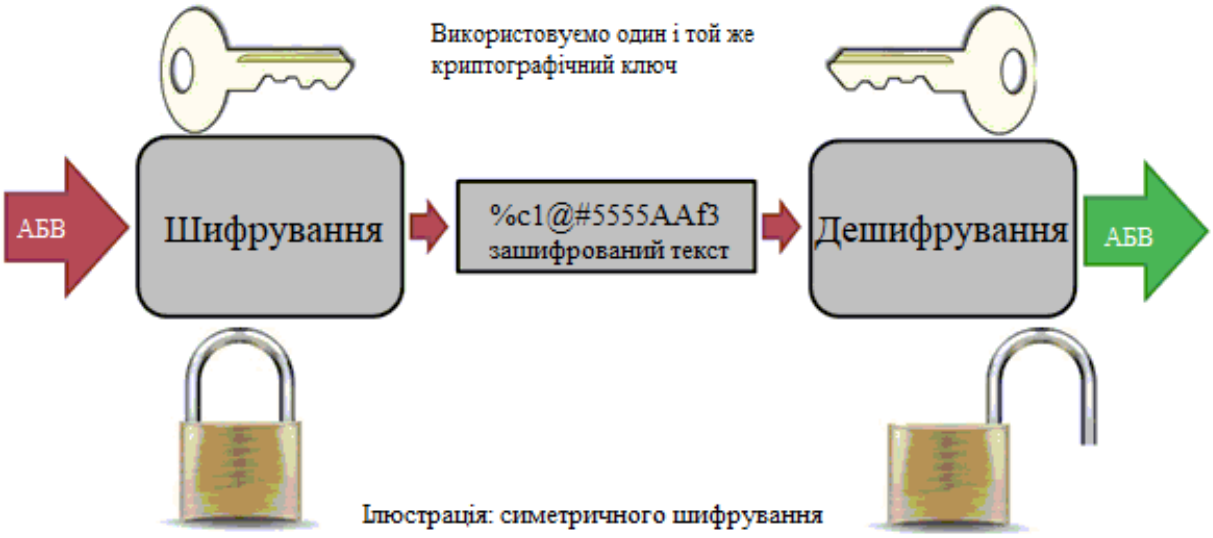
Алгоритми шифрування і дешифрування даних мають популярне місце у застосувані в комп'ютерній техніці в системах приховування конфіденційної і комерційної інформації від не коректного використання сторонніми особами. Головним принципом у них є умова, що особа яка приймає зашифрований текст, заздалегідь знає алгоритм шифрування, а також ключ до повідомлення, без якого ці дані є всього лише набір символів, які не мають жодного змістового сенсу.

Симетричні криптоалгоритми виконують перетворення невеликого (зокрема це 1 біт або 32-128 біт) блоків даних в залежності від ключа таким чином, що отримати оригінал тексту, можна тільки тільки у тому випадко, коли ти знаєш цей секретний ключ.

Концепція дуже проста:

- Є повідомлення А, котре Ви бажаєте відправити своєму другові
- Ви зашифровуєте дане повідомлення за допомогою певного ключа і отримуємте зашифрований текст В
- Ваш друг отримує цей зашифрований текст В
- Потім він розшифровує закодований вами текст використовуючи той же ключ і в результаті отримує текст А

Візуєально можна представити так:



Ілюстрація: симетричного шифрування

Криптографічних алгоритмів існує безліч. В загальному вони призначені для захисту інформації. Симетричні криптоалгоритми відносяться до криптоалгоритмів з ключем.

Вони поділяються на:

1. Потокові шифри – побітна обробка інформації. Шифрування і дешифрування в такому випадку обриваються в довільний момент часу, як тільки з'ясовується, що потік що передається перервався, і також відновлюється при виявленні факту продовження передачі.
2. Блочні шифри – перетворення блоку вхідної інформації фіксованої довжини. і отримують результуючий блок того ж обсягу.

Використовуються різні принципи потокового шифрування, AND, OR або

XOR (виключає АБО).

Розглянемо, приклад поточного шифрування:

Для цього шифрування потрібен ключ, який має таку ж кількість символів, що і повідомлення, і його потрібно використовувати тільки один раз (звідси назва «одноразовий»).

Припустимо, що ми відправляємо повідомлення Бобу «СКОРО УВИДИМСЯ». Але ми не хочемо, щоб хтось перехоплював наше повідомлення. Так що ми з Бобом вирішили використовувати одноразовий блокнот, який виглядає наступним чином [8]:

«Г Н П В К У О Х З Ю Т Э М»

Як бачимо, запис має стільки символів, що і повідомлення, тобто 13.

Це дуже простий приклад застосування одноразового блокнота, щоб краще зрозуміти його принципи.

З'являється ще одна річ: кожна буква з алфавіту замінюється числовим еквівалентом.

А	0	К	11	Х	22
Б	1	Л	12	Ц	23
В	2	М	13	Ч	24
Г	3	Н	14	Ш	25
Д	4	О	15	Щ	26
Е	5	П	16	Ъ	27
Ё	6	Р	17	Ы	28
Ж	7	С	18	Ь	29
З	8	Т	19	Э	30
И	9	У	20	Ю	31
Й	10	Ф	21	Я	32

В ході процесу шифрування буде 6 одиниць даних:

- ОМ вихідне повідомлення: оригінал якого ми написали «СКОРО УВИДИМСЯ».

- NOM числове вихідне повідомлення (його цифровий еквівалент)
- OTP одноразовий блокнот
- NOTP одноразовий блокнот
- NCT числовий зашифрований тест, який є сумою числового вихідного повідомлення і одноразового блокноту. Якщо сума більша, вираховується 33 у нашому випадку, по кількості букв в алфавіті з 0.
- CT зашифрований текст, який є алфавітним еквівалентом пункту який вище

Давайте почнемо по-порядку:

OM	NOM	OTP	NOTP	NCT	CT
С	18	Г	3	21	Ф
К	11	Н	14	25	Ш
О	15	П	16	31	Ю
Р	17	В	2	19	Т
О	15	К	11	26	Щ
У	20	У	20	7	Ж
В	2	О	15	17	Р
И	9	Х	22	31	Ю
Д	4	З	8	12	Л
И	9	Ю	31	7	Ж
М	13	Т	19	32	Я
С	18	Э	30	15	О
Я	32	М	13	12	Л

Так працює даний процес шифрування.

А як же працює процес дешифрування? Процес дешифрування відбувається за тим же самим ключем. Той хто отримав повідомлення ФШЮТЩЖРЮЛЖЯОЛ, має:

- Зашифровану фразу
- Загальний ключ
- Таблицю із числовими еквівалентами

І так, виникає питання. Як Боб дешифрує повідомлення, використовуючи ці дані?

- Він відобразить числові значення ключа, так зашифрованого повідомлення,

так щоб отримати NTC та NOTP

- Потім, підрахує NOM (числові значення вихідного повідомлення), виконав даний підрахунок : NOM = NCT – NOTP mod 33
- Надалі, використаємо таблицю для отримання відповідних букв.

Тепер, варто розібратись з блочним шифруванням.

Блоковий шифр має фіксовану довжину та кілька однакових блоків. Повідомлення розбивається на ті ж блоки, а якщо розмір фрази менше, її доповнюють, щоб повністю зайняти блок [8].

У одноразового блокнота довжина ключа дорівнює довжині повідомлення, а в блочному шифруванні ця властивість не є необов'язковою. Одним ключем можна зашифрувати довгий текст.

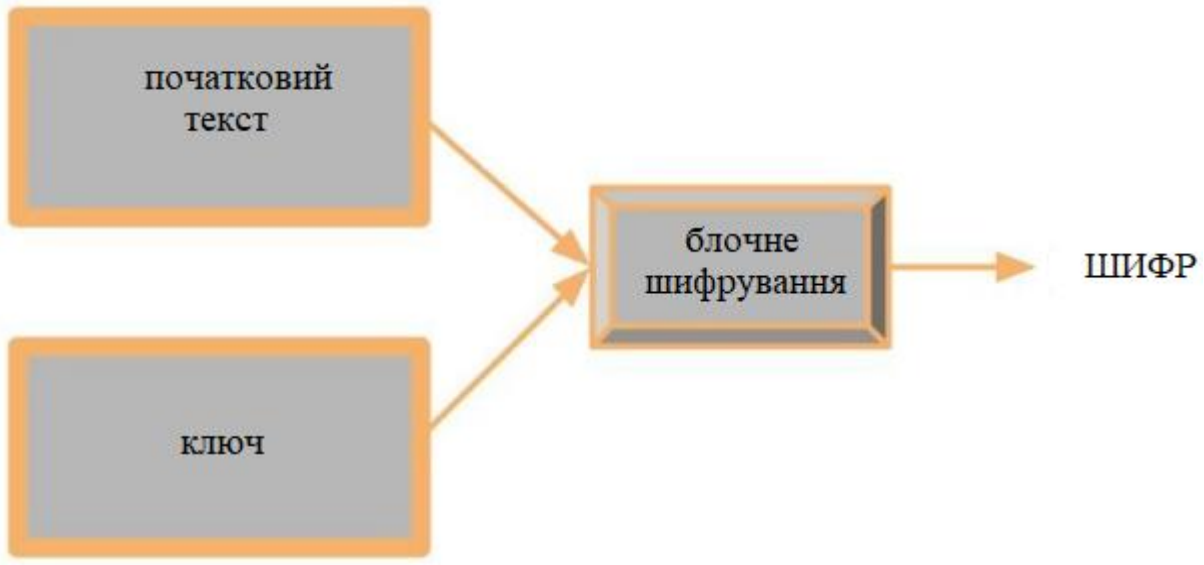
Найпростіший - метод підстановки Атбаш, коли алфавіт просто перевертається: А -> Я і навпаки. Також відомий Шифр Цезаря, коли буква приймає значення котре стоїть за нею (або розташованої на Х).

Наприклад, слово БАК -> ВБЛ, зрушено на 1:

- А Б В Г Д Е Ё Ж З И Й К Л
- Б В Г Д Е Ё Ж З И Й К Л А

Вище наведені приклади є простими, зазвичай шифрування відбується з великими фрагментами даних.

На картинцю, це можна зобразити так:



Плюси та мінуси симетричної криптографії

Незважаючи на те, що симетрична криптографія має деякі серйозні проблеми (що ми обговоримо нижче), проте найбільша її перевага полягає в тому, що вона вимагає дуже значних ресурсів. Все що потрібно, це надати одержувачу один той самий ключ, щоб метод спрацював.

Навіть зараз велика кількість ПО використовує цей метод в поєднанні з асиметричною криптографією для забезпечення швидких і ефективних служб шифрування / дешифрування.

Але наведемо декілька проблем симетричного шифоування:

- Головна проблема - шифрування і дешифрування виконується за допомогою одного ключа. Якщо хтось перехопить ключ, всі дані будуть скомпрометовані
- Немає масштабованості, тобто, припустимо, що інформаційний центр відправляє дані за допомогою симетричної кріптографії. Все нормально, поки клієнтів всього 3-4. Але чим більше їх, тим більше незручно обробляти всі ключі.

Дані вразливості симетричного ключа сприяли появі нового рішення в 1970-х роках.

3.2. Асиметричне шифрування

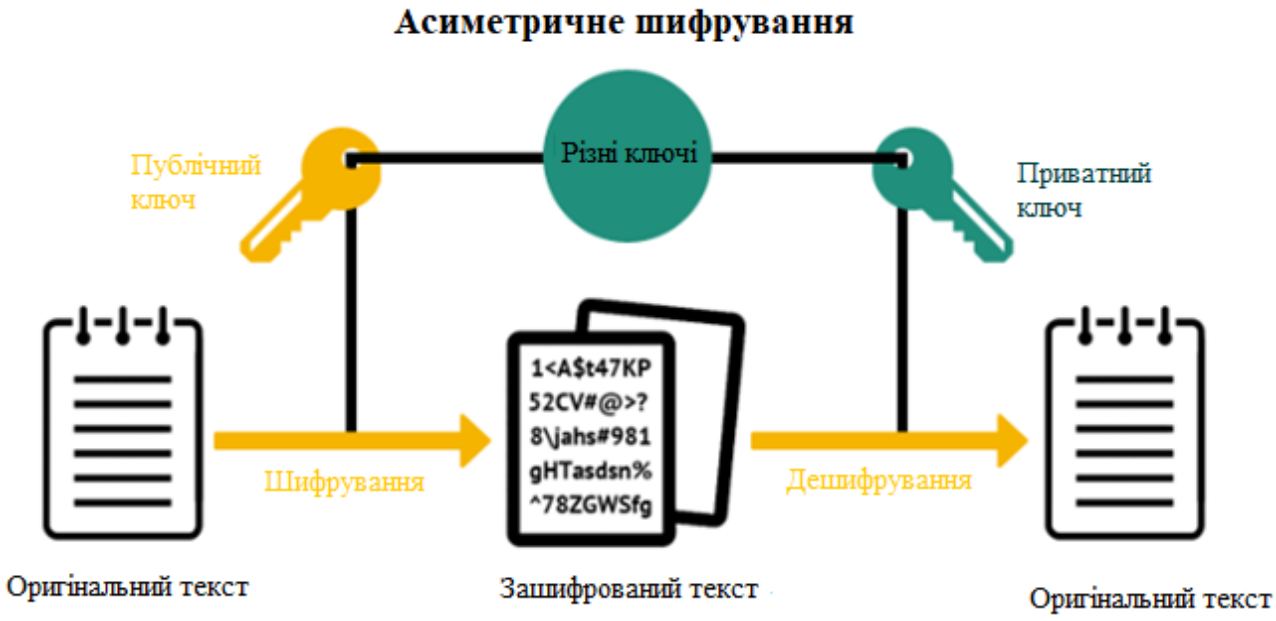
У 1970 році британський математик і інженер Джеймс Елліс запропонував ідею, заснована на прості концепці. Що, якщо шифрування і дешифрування - зворотні операції на основі двох різних ключів? У традиційній, тобто симетричній криптографії, повідомлення повинно бути надіслано разом з ключем, щоб інша сторона розшифрувала повідомлення.

Елліс припустив, що одержувач повідомлення не може бути пасивною стороною, і їм потрібно було мати «замок» і «ключ» для себе. Замок можна було відправити кому завгодно в світі, але ключ повинен залишатися приватним.

Була теорія, а практичне застосування цієї теорії складається з двох блискучих принципів:

- Одностороння функція з постійним входом: легко перейти від одного стану до іншого, але повернутися у вихідну позицію без ключа неможливо; K - відкритий ключ, k - приватний. Ключі математично пов'язані один з одним через функцію, K = f (k). Простий приклад - множення великих чисел. Наприклад, у вас є числа 1847 і 19837. Їх добуток 36638939. Однак, якщо ви просто знаєте це число(добуток), ви не знайдете ні один з множників. Потрібно знати хоча б один, щоб дізнатись другий [13].
- Обмін ключами Діффі-Хеллмана, Протокол Діффі-Хеллмана. Це обмін ключами по незахищених каналах зв'язку. Наведемо приклад, для більш зрозумілості: Аліса і Боб живуть в країні, де немає таємного листування. Але наші друзів дуже хочуть зберегти свою таємницю. Тоді Аліса відправляє замкнений замком ящик з листом Бобу. Боб його отримує, вішає свій замок, відправляє Алісі посилку з двома замками: старим і новим. Аліса отримує, знімає свій замок, відправляє Бобу. Він отримує, відкриває замок своїм ключем, відкриває ящик і читає лист. Так ні до Аліси, ні до Бобу не потрапляли чужі ключі, і також ніхто не вкрав би їх по дорозі.

Асиметрична криптографія використовує два ключа, відкритий ключ та приватний, для шифрування і дешифрування конкретних даних. Використання одного ключа скасовує використання іншого [9].



3.3. Криптографія з відкритим ключем та криптовалюта

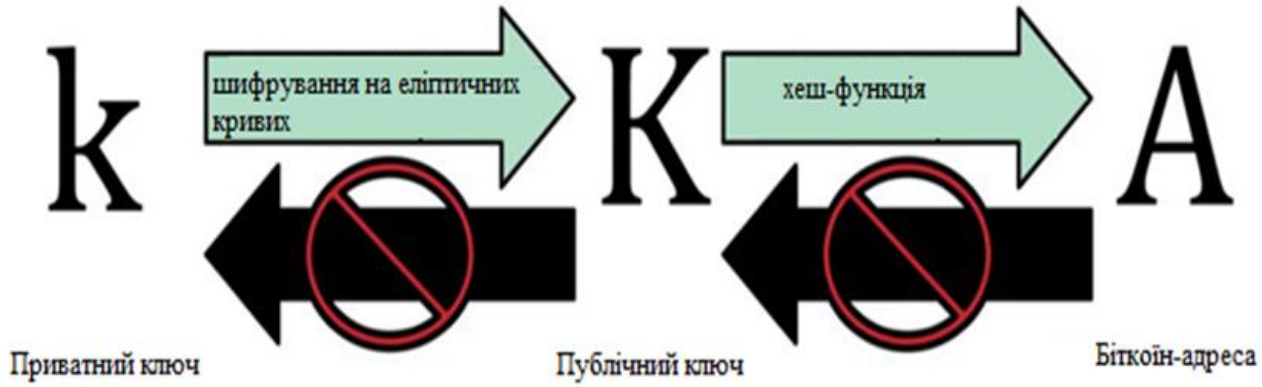
Криптографія з відкритим ключем була винайдена в 1970-х роках і є математичною основою для комп'ютерної та інформаційної безпеки [12].

З моменту винаходу криптографії з відкритим ключем, було відкрито декілька математичних функцій, таких як піднесення до степеня простого числа і множення еліптичних кривих. Ці математичні функції практично незворотні, це означає, що результат цих функцій легко отримати в одному напрямку і неможливо в зворотному. На підставі цих математичних функцій, криптографія дозволяє створення цифрових шифрів і непідробних цифрових підписів. У біткоїнах використовується множення еліптичних кривих.

У біткоїнах для створення пари ключів, які контролюють доступ до засобів, використовується криптографія з відкритим ключем. Пара ключів складається з приватного ключа і, похідного від нього, унікального, публічного ключа. Публічний ключ використовується для отримання біткоїнів, а закритий ключ використовується для підпису транзакцій. Між публічним і приватним ключами існує математичне співвідношення, яке дозволяє підписувати повідомлення приватним ключем, а цей підпис потім можна перевірити за допомогою публічного ключа, не розкриваючи при цьому приватну частину. Коли хтось хоче витратити свої біткоїни, він надає свій відкритий ключ і підпис (кожен раз різний, але створений на основі все того ж приватного ключа) в блоці транзакцій. Шляхом відкритого публічного ключа і підпису, всі вузли мережі можуть перевірити і прийняти транзакцію за дійсну, підтверджуючи, що особа, яка передає права на кошти, є їх власником.

3.3.1. Приватні та публічні ключі

Біткоїн-гаманець містить колекцію з пар ключів, кожна з яких складається з приватного і публічного ключа. Приватний ключ (k) являє собою число, як правило, взяте навмання. Використовуючи приватний ключ, за допомогою односторонньої функції множення еліптичних кривих, ми отримуємо публічний ключ (K). З публічного ключа (K), використовуючи односторонню функцію криптографічного хеша ми отримуємо біткоіни-адреси (A). У цьому розділі ми почнемо з створення приватного ключа, поглянемо на математику еліптичної кривої, яка використовується для перетворення його в публічний ключ і, нарешті, створимо біткоіни-адреса з публічного ключа. Відношення між приватним ключем, публічним і Біткоїн-адресою показано нижче:



Приватні ключі

Приватний ключ - це просто число, яке взяте навмання. Володіння приватним ключем необхідно для контролю користувача над засобами, пов'язаними з відповідною біткоїн-адресою. Приватний ключ використовується для створення підпису, яка необхідна як доказ володіння операціями в транзакції. Приватний ключ зберігатися в повному секреті, тому що його витік в інші руки еквівалентно передачі контролю над засобами, «замкненими» цим ключем. Приватний ключ також повинен мати резервну копію і захищений від випадкової втрати, тому що якщо він буде втрачений, то і кошти так само будуть втрачені назавжди.

Ще раз, приватний ключ - це просто число. Можете взяти випадковий приватний ключ, використовуючи тільки монету, олівець і папір: кинути монету 256 раз, і у вас є двійкове число випадкового приватного ключа, який можна використовувати в біткоїн-гаманці. Публічний ключ може бути згенерований з приватного ключа.

Створення приватного ключа із випадкового числа

Перший і найважливіший крок при генерації ключів - це знайти надійне джерело ентропії, або випадковості. Створення ключа Bitcoin, по суті, те ж саме, що «взяти число між 1 і 2^{256}». Точний спосіб вибору числа не має значення до тих пір, поки не передбачуваний або повторимо. Програмне забезпечення біткоіни використовує генератори випадкових чисел операційної системи. Як правило, генератор випадкових чисел ОС инициализируется джерелом випадковості користувача, тому вам може бути запропоновано випадково поворухнути мишею протягом декількох секунд. Для справжніх параноїків, ніщо не зрівняється з кістками, олівцем, і папером.

Більш точно, приватний ключ може бути будь-яким числом між 1 і n - 1, де n константа (n = 1.158 * 1077, трохи менше, ніж 2256) визначається як порядок еліптичної кривої використовуваної в Bitcoin. Щоб створити подібний ключ, ми випадково вибираємо 256-бітове число, і переконуємося, що воно менше, ніж n - 1. В термінах програмування, це зазвичай досягається шляхом подачі великий рядка випадкових бітів, взятих з криптографически стійкого джерела випадковості, на вхід хеш алгоритму SHA256, який видасть зручне 256-бітове число. Якщо результат менше, ніж n -1 [2], ми отримуємо відповідний приватний ключ. В іншому випадку, ми просто пробуємо ще раз з іншим випадковим числом.

Нижче наведений випадковим чином згенерований секретний ключ (k) в 16-тирічному форматі:

```
1E99423A4ED27608A15A2616A2B0E9E52CED330AC530EDCC32C8FFC6A526AEDD
```

Публічні ключі

Публічний ключ обчислюється з приватного ключа за допомогою незворотного множення на еліптичних кривих: (K = k * G) [3], де k - це приватний ключ, G - константна точка, яка називається *генераторної точкою*, а K - результуючий публічний ключ. Зворотна операція, відома як "знаходження дискретного логарифма", обчислення k при відомому K можлива тільки за допомогою повного перебору k. Перш, ніж ми продемонструємо, як створити публічний ключ з приватного, давайте поглянемо на криптографію на еліптичних кривих трохи більш докладно.

Криптографія на еліптичних кривих

Останнім часом в криптографії інтенсивно почали використовувати еліптичні криві. ЕК– це розділ криптографії, який вивчає асиметричні криптосистеми, заснованих на еліптичних кривих над кінцевими полями [3]. Асиметрична криптографія заснована на складності рішення деяких математичних задач. Напротиріч раннім криптосистеми з відкритим ключем, такі як алгоритм RSA, наприклад, безпечні завдяки тому, що досить важко розкласти складене число на прості множники. У випадку застосування алгоритмів на еліптичних кривих вважається, що не існує субекспоненціальних алгоритмів щодо розв'язку задачі «дискретного логарифмування в групах їх точок» [4]**.**

У сучасній криптографії на основі еліптичних кривих бінарної розмірності в діапазоні від (150 до 350) забезпечується рівень криптографічної стійкості, який потрібно використовувати у відомих криптографічних системах бінарної

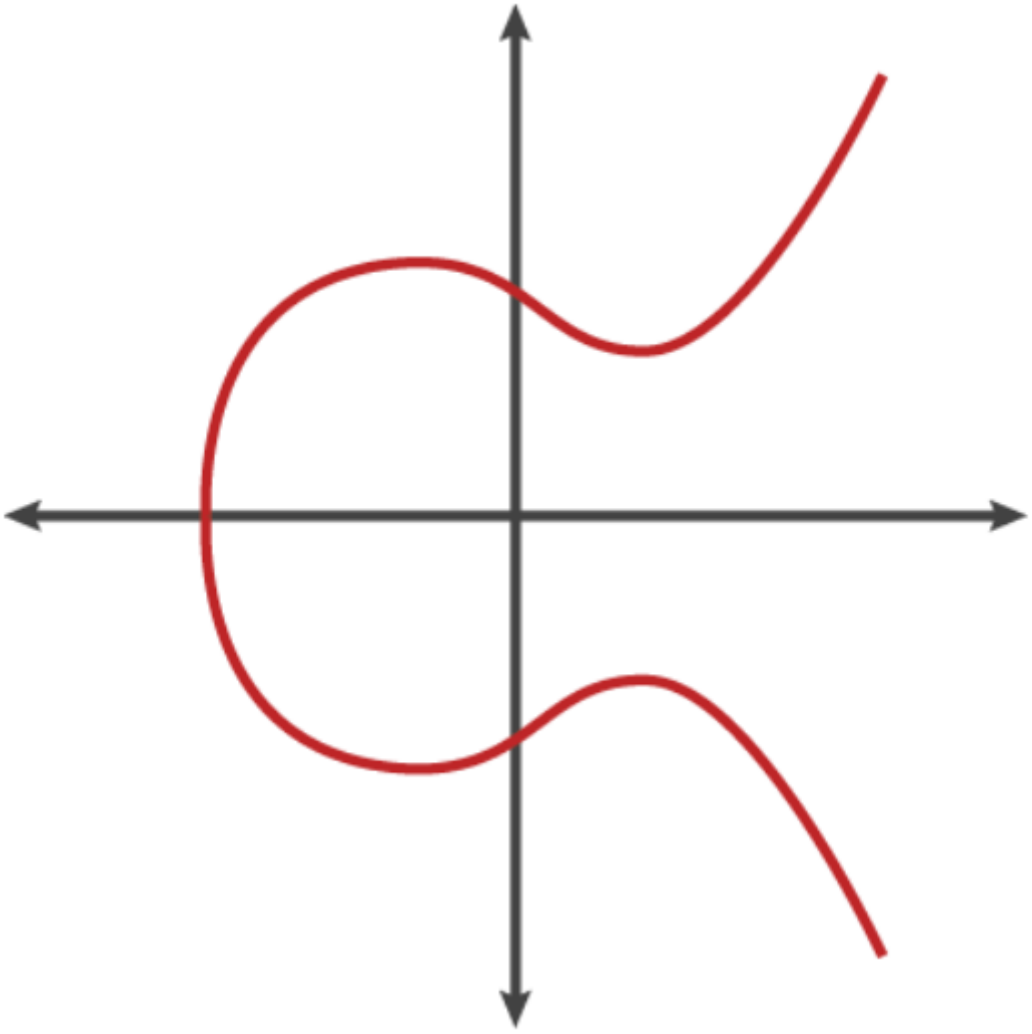
розмірності від (600 до 1400) і більше.

Розглянемо принципи побудови і використання еліптичних кривих в криптографії.

Кубічна крива на декартовій системі координат *XY*, задана рівнянням

$$y^2 = x^3 + ax + b$$

що не має особливих точок, називається еліптичною кривою [9].



В реальних криптосистемах використовується рівняння

$$y^2 = x^3 + ax + b \pmod{p}, \qquad (1)$$

де a, b належать полю GF(p), $4a^3 + 27b^2 \pmod{p} \neq 0$, p – просте число. Множині $E_p(a, b)$ належать усі точки (x, y), $x \geq 0$, $p > y$, що задовольняють рівняння (1), і точку в нескінченності O, яка є нулем для площини еліптичних кривих. Кількість точок еліптичної кривої позначається $E_p(a, b)$ [14].

Множина точок $E_p(a, b)$ має такі властивості:

- якщо P = (x, y) є точкою ЕК, то $(x, y) + (x, -y) = O$; точка $(x, -y)$ позначається як –P;
- якщо P=(x_1, y_1) і Q = (x_2, y_2), де P≠Q, то P + Q = (x_3, y_3) обчислюється так:

$$x_3 = (y_2–y_1)/(x_2–x_1)–x_1–x_2; \quad y_3 = –y_1–(y_2–y_1)(x_2–x_1)/(x_2–x_1) \qquad (2)$$

- якщо P = Q, то подвоєння точки $P = (x_3, y_3)$ обчислюється за формулою [14]:

$$x_3 = ((3x_1^2+a)/2/y_1)^2–2x_1; \quad y_3 = –y_1–(3x_1^2+a)/2/y_1 \qquad (3)$$

Одним із найрозповсюджинішим застосуванням криптографічних еліптичних кривих є розподіл ключів. Даний криптографічний протокол обміну ключами з використанням еліптичної кривої має представлення:

- користувач повинен вибрати та оповістити всім форму еліптичної кривої та точку G на даній кривій, яка власне і є генеруючою точкою;
- користувач 1 повинен вибрати ціле число k і знайти точку $P_A = k*G$ (або ж простіше, додати точку G до самої себе k разів);
- користувач 2 повинен вибрати число m та обчислити точку $P_B = m*G$. Після даної операції користувачі обмінюються власними результатами та спільним їх секретним ключем стає точка $k*m*G$.

Для того щоб розкрити даний протокол потрібно за відомими $k*G$ та G обчислити $k<p$, що є аналогічною важкою задачею дискретного логарифмування, як от у випадку алгоритму RSA: де легко обчислити P за відомими k і G, проте важко обчислити k за відомими P і G.

Для криптографічного застосування еліптичних кривих однією з проблем є вибір надійної випадкової кривої. Саме стійкість результуючої криптосистеми визначає вирішення цієї проблеми .

Алгоритм формування надійної випадкової кривої .

1. Обирається випадкове просте число p.
2. Далі вибираються випадкові числа a і b, такі, що $a, b \neq 0$ і $(4a^3 + 27b^2) \pmod p \neq 0$ [14]. Задля ефективності розрахунків можна рандомно обирати тільки b, а a приймати рівним невеликому цілому числу.
3. Визначається кількість точок на кривій $n = E_p(a, b)$. Слід зазначити, що є важливим щоб n мало великий простий дільник q (це розмір підмножини точок кривої).
4. Виконується процедура перевірки виконання умови $(p^k – 1) \bmod q \neq 0$ для всіх k, $0 < k < 32$ [14]. У випадко, коли умова не виконується, то повертаэмось до пункту 2.
5. Виконується перевірка виконання умови $q \neq p$. Якщо ні, то повертаються до кроку 2.
6. Знаходимо точку G – генератор підмножини точок q. Коли $q = n$, то будь-які точки (крім O) є генераторними. Коли $q < n$, то вибираються випадкові точки G', до тих пір, поки не виконається умова $G = [n/q]G' \neq 0$ [15].

3.4. Електронний цифровий підпис

Певний час у криптографії використовувався лише алгоритм симетричного шифрування, де відправник повинен передати отримувачу разом із зашифрованим текстом і свій секретний ключ, яким було зашифроване це повідомлення, що створювало певні ннезручності, адже потрібно було забезпечити наявность закритого каналу для передачі секретного ключа, що збільшувало ризик розкриття інформації. Асиметричні алгоритми шифрування (в протиріч до симетричних) використовують пару споріднених ключів – публічний та приватний. При цьому, незважаючи на пов'язаність ключів у парі, обчислення секретного ключа на основі відкритого вважається технічно неможливим. В асиметричній криптосистемі публічний ключ може спокійно у відкритому доступі, в той час як приватний ключ має зберігатись лише у строгій таємниці.

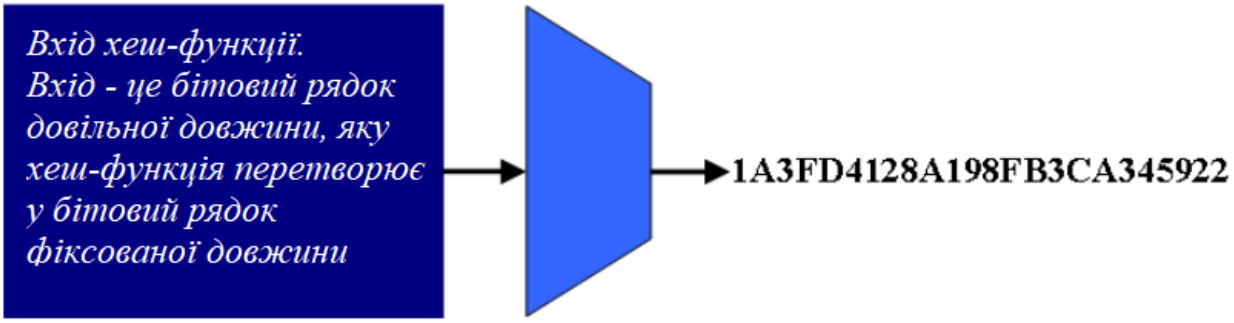
Історія, дослідження в напрямку криптографії з відкритим ключем була розпочата в 1975 році шляхом об'єднання зусиль двох незалежних груп учених У. Діффі – М. Хеллмана та Р. Меркла в Стенфордському університеті, ця праця призвела до відкриття відомого алгоритму, який має назву Діффі–Хеллмана–Меркла (протокол обміну ключами), він став основою для створення міжнародної інфраструктури відкритих ключів (та сама схема була розроблена М. Вільямсоном у 1970-х, але трималася в секреті до 1997 року). Роком пізніше групою вчених із Массачусетського технологічного інституту Р. Рівестом, Л. Адлеманом та А. Шаміром було винайдено перший, алгоритм асиметричного шифрування RSA, що дозволив вирішити проблему спілкування через відкриті, незасекречені канали, він і став основою для створення Електронно Цифрового Підпису – складової інфраструктури відкритих ключів. Перші реалізації ЕЦП були свторені для аутентифікації текстів, які відправляються по телекомунікаційних каналах, зі збереженням основних властивостей звичайного рукописного. За реалізацією ЕЦП є невеликою кількістю додаткової інформації, що передається разом із підписаним текстом. При створенні ЕЦП використовується закритий ключ, а при перевірці – відкритий. В алгоритм генерації цифрового підпису було закладено, що неможливо створити ЕЦП без секретного ключа, який вже потім, при перевірці буде визнаний коректним. ЕЦП застосовується для того, аби підтвердити, що дане повідомлення надійшло дійсно від відправника (за припущення, що лише відправник володіє закритим ключем, відповідним його відкритому ключу) [14].

Одинм із використанньь ЕЦП є проставлення штампа часу (timestamp) на різних документах: приклад, сторона, до якої у вас є довіра, підписує документ із штампом часу за допомогою свого закритого ключа що тим самим значить, що сторона підтверджує про існування документа, на момент оголошення у штампі часу.

Основні властивості систем криптографічного захисту, такі як цілісність, справжність і неспростовність відправника, можуть бути забезпечені за умови обов'язкового використання ЕЦП. Обов'язковим елементом, що використовується в ЕЦП, є хеш-функція, за допомогою якої обчислюється хеш-значення від електронних даних і взагалі інформації, що підписується.

3.5. Криптографічні хеш-функції

Криптографічна хеш-функція, або хешування — це процес перетворення вхідного масиву даних довільної довжини у вихідний бітовий рядок фіксованої довжини [15].



Хеш-функція – це певна функція h(K), котра бере ключ K і в результаті повертає адресу, по якому відбувається пошук в хеш-таблиці, для того щоб отримати інформацію, пов'язану з K.

Колізія — це ситуація, коли h(K1) = h(K2) [15]. У випадку з колізією, необхідно знайти вже нове місце для зберігання даних. Певна річ, що кількість колізій повинна бути максимально наближена до 0. Хеш-функція повинна задовольняти таким вимогам:

- її обчислення повинно виконуватися дуже швидко;
- вона повинна мінімізувати число колізій

Перша властивість хешування залежить прямою мірою від потужностей комп'ютера, друга — від даних.

Застосування хешування

Одна із причин застосувань хешування полягає в тому, що вона створює певного роду копію, або ж, ще називають "відбиток пальця" для повідомлення, текстового рядка і т. п. Цей "Відбиток пальця" може прагнути як і до "унікальності", так і до "схожості". При створенні хеш-функціій односпрямованого характеру часто використовують функцію стиснення (що видає значення довжини n при вхідних даних більше довжини m і працює з кількома вхідними блоками). При хеш-функціях враховується довжина повідомлення, щоб виключити проблему появи однакових хеш-адрес для повідомлень різної довжини. Найбільшу популярність мають такі хеш-функції [2]: MD4, MD5, RIPEMD-128 (128 біт), RIPEMD-160, SHA (160 біт). У українському стандарті цифрового підпису використовується розроблена вітчизняними криптографами хеш-функція (256 біт) стандарту ГОСТ 34.311-95.

Розділ 4. Bicoin-адреса

Біткоїн-адреса являє собою рядок, що містить у собі цифри та символи латинського алфавіту. Адреси, отримані з публічних ключів починаються переважно з цифри "1". Нижче наведений приклад, існуючої біткоїн-адреси:

```
1J7mdg5rbQyUHENYdx39WVWK7fsLpEoXZy
```

Біткоїн-адреса використовується в системі задля отримання коштів в транзакціях. Якщо порівняти біткоїн-транзакцію зі звичним нам чеком, то адреса одержувача це те, що ми пишемо в рядку «Вкажіть контакти одержувача». «Одержувачем» на паперовому чеку іноді може виступати ім'я власника банківського рахунку, так ще й може бути і назва фірми, установи. Так як на паперових чеках можна не вказувати номер рахунку, а використовувати абстрактне ім'я в якості отримувача коштів, то ці чеки стають дуже гнучкими платіжними інструментами. Біткоїн-транзакції використовують дещо подібну абстрактність.

Біткоїн-адреса генерується з публічного ключа шляхом використання одностороннього криптографічного алгоритму хешування. А "алгоритм хешування" або просто "алгоритм хешу" - це одностороння функція, яка здійснює відбиток або "хеш" довільної кількості даних на вході.

Криптографічні хеш-функції широко використовуються в системі біткоїн, а саме: в біткоїн-адресах, в адресах сценаріїв, в процесі майнінгу.

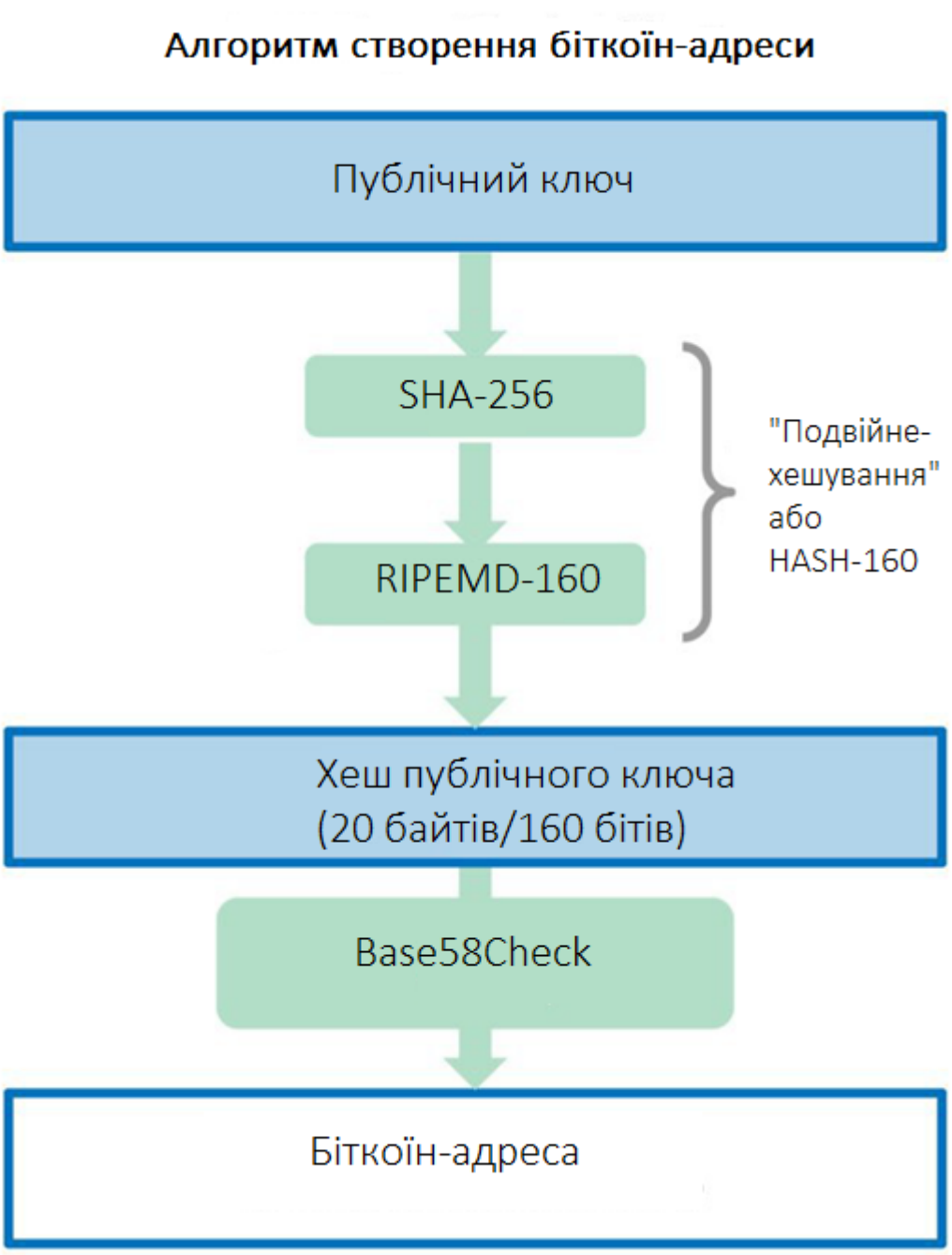
Адреса з публічного ключа отримується в результаті застосування алгоритму під назвою Secure Hash Algorithm (SHA) і RACE Integrity Primitives Evaluation Message Digest (RIPEMD), конкретно SHA256 and RIPEMD160[4].

Починаючи з відкритого ключа K, ми обчислюємо SHA256-хеш і результат «пропускаємо» через хеш алгоритму RIPEMD-160, отримавши на виході 160-ти бітове (20-байт) число: де K - це публічний ключ і A результуюча біткоїн-адреса.

1.1. Створення Біткоїн-адреси

Слід зазначити, що біткоїн-адреса - це не те ж саме, що публічний ключ. Біткоїн-адреса отримується з публічного ключа за допомогою односторонньої функції. Біткоїн-адреси майже завжди мають вигляд представлений в кодуванні «Base58Check», в якій використовується 58 символів (формат Base58) і контрольна сума для кращого представлення, уникнення двозначностей та захисту від будь-яких помилок. Base58Check використовується і в багатьох інших випадках, коли важливе коректне представлення числа. Простішими словами, Base58Check використовується для зручного читання числа.

Нижче подана ілюстрація перетворення публічного ключа в Біткоїн-адресу:



4.1.1 Кодування Base58 та Base58Check

Для представлення довгого числа в компактному вигляді (використовуючи менше символів), багато комп'ютерних систем використовують змішані буквено-цифрові системи числення. Наприклад, в той час як традиційна десяткова система використовує 10 цифр від 0 до 9, шістнадцяткова система використовує 16, з буквами від A до F як шести додаткових символів. Число, представлене в шістнадцятковому форматі коротше, ніж еквівалентна йому в десятковому. Для ще більш компактної форми, використовують Base-64 для передачі бінарних даних в "текстових полях", на зразок системи email. Алфавіт Base-64 складається з 26 великих літер та такої ж самої кількості малих, 10 цифр, і ще двох символів: "+" і "/". Base-64 найчастіше використовується для кодування повідомленнь електронної пошти. Формат кодування Base58 розроблений для використання в біткоїнах і

використовується в багатьох інших криптовалютах. Він пропонує своєрідний баланс між компактним представленням, зручною читаністю, і запобіганням помилок. Base58 - це деяека підмножина Base64, що використовує великі та малі літери, а також цифри, але без деяких символів, які часто помилково приймають за один одного і можуть відображатися ідентично деякими шрифтами. Зокрема, Base58 - це Base64 без числа 0 (нуль), O (великої літери O), l (маленької L), I (велика і), і символів "\ +" і "/". Або, простіше кажучи, це набір великих та малих літер, цифр без чотирьох (0, O, L, I), згаданих вище.

Приклад 1. Алфавіт Base58

```
123456789ABCDEFGHJKLMNPQRSTUVWXYZabcdefghijkmnopqrstuvwxyz
```

Щоб додати додатковий захист від помилок або помилок транскрипції, Base58Check - це Base58 з вбудованим кодом перевірки помилок.

4.1.2. Формати ключів

Приватний і публічнй ключ можуть бути представлені в різних форматах. Ці формати використовується, в основному для того, щоб люди могли сприймати на слух і вимовляти текст ключів без помилок.

У нище поданій таблиці приватних ключів (формати кодування) показані три поширені формати, які використовуються в системі біткоїн.

Тип	Префікс	Опис
Шістнадцятковий	Не має	64 шістнадцятковий чисел
WIF	5	Кодування Base58Check: Base58 з префіксом версії 128 і 32-бітною контрольною сумою
Стислий WIF	K чи L	Те що й вище, але з додаванням суфіксу 0x01 перед кодуванням

Приклад: Один і той же ключ у різних форматах

Формат	Приватний ключ
Шістнадцятковий	1e99423a4ed27608a15a2616a2b0e9e52ced330ac530edcc32c8ffc6a526aedd
WIF	5J3mBbAH58CpQ3Y5RNJpUKPE62SQ5tfcvU2JpbnkeyhfsYB1Jcn
Стислий WIF	KxFC1jmwwCoACiCAWZ3eXa96mBM6tb3TYzGmf6YwgdGWZgawvrtJ

Всі ці представлення просто різні способи закодувати одне і те ж число, один і той же приватний ключ. Вони виглядають по-різному, але будь-який формат може бути легко перетворений в будь-який інший формат.

Формати публічних ключів

Публічні ключі також представлені по-різному. Як ми бачили раніше, публічний ключ - це точка на еліптичній кривій, що має координати (x, y). Як правило він представлений префіксом 04 і двома 256-бітними числами після префікса: в першому числі x координата точки, а в другому y координата..

Ось, відкритий ключ, згенерований з приватного ключа, у вигляді осі координат x та y:

```
x = F028892BAD7ED57D2FB57BF33081D5CFCF6F9ED3D3D7F159C2E2FFF579DC341A
y = 07CF33DA18BD734C600B96A72BBC4749D5141C90EC8AC328AE52DDFE2E505BDB
```

Ось той самий ключ продемонстрований у вигляді 520-бітного числа (130 шістнадцятирічних ключів) з префіксом 04 і далі x та y у вигляді 04 x y:

```
K = 04F028892BAD7ED57D2FB57BF33081D5CFCF6F9ED3D3D7F159C2E2FFF579DC341A<?pdf-
cr?>07CF33DA18BD734C600B96A72BBC4749D5141C90EC8AC328AE52DDFE2E505BDB
```

4.1. Власна реалізація біткоїн-адреси на мові програмування Java

Перед демонстрацією коду, всласне самої програми, коротко нагадаємо принцип роботи криптовалюти біткоїн та які технологічні аспекти вона в себе включає:

- Біткоїн-адреса використовується для відправки та откримання біткоїнів
- Транзакція – передача біткоїнів з однієї адреси на іншу

- Транзакції групуються в блоки. В свою чергу додаючи дані блоки до біткоїн-мережі ми тим самим створюємо процес майнінгу.
- Блоки «збираються» в так званий ланцюжок блоків

Біткоїн-адреса представляє собою випадковий рядок шістнадцятирічних символів, які використовуються в мережі Bitcoin для відправлення та прийому цифрової валюти. Це загальнодоступна частина асиметричного ключа ECDSA з публічним і приватним ключами. Відповідно, закритий ключ використовується для підписання транзакції біткойна як доказ того, що транзакція виникла саме у вас.

Технічно, біткоїн-адреса генерується з частини публічного ECDSA ключа, хешується виокристовуючи SHA-256 і RIPEMD-160, обробляє результуючі хеші, і нарешті, кодує ключ використовуючи кодування Base58Checked[16].

Що ж, проаналізуємо кожну частину коду нашої програми реалізованої на мові програмування Java окремо, зазначимо повністю код знаходиться в Додаток. Текст програми.

Як ми зазначали раніше, біткоїн використовує ECDSA для алгоритму генерування ключів. Тому, створюємо пари ключів ECDSA.

Створюємо *KeyPairGenerator* для алгоритму еліптичних кривих.

```
KeyPairGenerator keyGen = KeyPairGenerator.getInstance("EC");
```

Використовуємо еліптичну криву – *secp256k1*.

```
ECGenParameterSpec ecSpec = new ECGenParameterSpec( stdName: "secp256k1");
keyGen.initialize(ecSpec);
```

Коли у нас є *KeyPairGenerator,* ми можемо створити *KeyPair* з якого маємо можливість отримати публічний та приватний ключ.

```
KeyPair kp = keyGen.generateKeyPair();
PublicKey pub = kp.getPublic();
PrivateKey pvt = kp.getPrivate();
```

Ми можете зберігати тільки приватну частину ключа, оскільки публічний ключ може бути отриманий у будь-який момент із приватного.

```
ECPrivateKey epvt = (ECPrivateKey) pvt;
String sepvt = adjustTo64(epvt.getS().toString( radix: 16)).toUpperCase();
System.out.println("s[" + sepvt.length() + "]: " + sepvt);
```

Статичний метод *adjustTo64()* просто заповнює рядок з шістнадцятирічної системи числення починаючи з 0, тому загальна довжина складає 64 символа.

```
private static String adjustTo64(String s) {
    switch (s.length()) {
        case 62:
            return "00" + s;
        case 63:
            return "0" + s;
        case 64:
            return s;
        default:
            throw new IllegalArgumentException("not a valid key: " + s);
    }
}
```

Ось прикдал закритого ключа, згенерованого вищенаведеним кодом:

```
Tkachuk
"C:\Program Files\Java\jdk1.8.0_112\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
s[64]: AE45F0644349D91697750D01990DF2427204338ED7549B1AF650E68253F78514

Process finished with exit code 0
```

Дана послідновність числа зазвичай зберігається в крипто-гаманцях.

Публічна частина ключа, згенерованого вище, кодується в біткоїн-адресу.

Ключ ECDSA є точкою на еліптичній кривій. Координати X та Y даної точки складають публічний ключ. Вони об’єднуються разом з «04», щоб представити публічний ключ.

```
ECPublicKey epub = (ECPublicKey)pub;
ECPoint pt = epub.getW();
String sx = adjustTo64(pt.getAffineX().toString( radix: 16)).toUpperCase();
String sy = adjustTo64(pt.getAffineY().toString( radix: 16)).toUpperCase();
String bcPub = "04" + sx + sy;
System.out.println("bcPub: " + bcPub);

/*
# prints
bcPub: 04CAAA5C0BDDAA22C9D3C0DDAEC8550791891BB2C2FB0F9084D02F927537DE4F443ACED7DEB488E
9BFE60D6C68596E6C78D95E20622CC05474FD962392BDC6AF29
 */
```

Тепер нам потрібно виконати операцію з алгоритмом шифрування SHA-256, а потім, отриманий результат «пропускаємо» через алгоритм RIPEMD-160.

```
MessageDigest sha = MessageDigest.getInstance("SHA-256");
byte[] s1 = sha.digest(bcPub.getBytes( charsetName: "UTF-8"));
System.out.println("  sha: " + bytesToHex(s1).toUpperCase());

 /*
# prints
sha: 7524DC35AEB4B62A0F1C90425ADC6732A7C5DF51A72E8B90983629A7AEC656A0
 */
```

Ми використовуємо Bouncy Castle для виконання RIPEMD-160, оскільки JCE не реалізує цей алгоритм

```
MessageDigest rmd = MessageDigest.getInstance( algorithm: "RipeMD160",  provider: "BC");
byte[] r1 = rmd.digest(s1);
```

Далі ми повинні додати байт версію 0x00 на початок хешу.

```
byte[] r2 = new byte[r1.length + 1];
r2[0] = 0;
for (int i = 0 ; i < r1.length ; i++) r2[i+1] = r1[i];
System.out.println("  rmd: " + bytesToHex(r2).toUpperCase());

/*
# prints
rmd: 00C5FAE41AB21FA56CFBAFA3AE7FB5784441D11CEC
 */
```

Виконуємо «подвійне хешування».

```
byte[] s2 = sha.digest(r2);
System.out.println("  sha: " + bytesToHex(s2).toUpperCase());
byte[] s3 = sha.digest(s2);
System.out.println("  sha: " + bytesToHex(s3).toUpperCase());
```

Перші 4 байта результату другого хешування використовується як контрольна сума адреси. Вона додається до хешу RIPEMD160. Це 25-байтовий адрес біткойну.

```
byte[] a1 = new byte[25];
for (int i = 0 ; i < r2.length ; i++) a1[i] = r2[i];
for (int i = 0 ; i < 5 ; i++) a1[20 + i] = s3[i];
```

Кодуємо адресу викристовуючи Base58

Тепер ми використовуємо метод Base58.encode () з бібліотеки bitcoinj, щоб отримати остаточну адресу біткоїну.

```
System.out.println("  adr: " + org.bitcoinj.core.Base58.encode(a1));
/*
adr: 1K3pg1JFPtW7NvKNA77YCVghZRq2s1LwVF
 */
```

Висновок

В ході роботи було досліджено які саме криптографічні алгоритми використовуються на базі технології Blockchain в криптовалюті Bitcoin. Слід зазначити, що теоретична база хоч і активно розвивається і є перспективною, проте дана технологія є ще не надто добре вивчена.

На мій погляд, ті програмні продукти, які будуть базуватись на технології Blockchain є в апріорі перспективним напрямком сучасних досліджень по децентралізації сховищ даних та створенню смарт-контрактів.

Поєднання елементів Blockchain, таких як розподілена база даних, криптографічне хеш-кодування, і цифрові підписи створюють дуже ефективну, нову форму передачі даних і активів, здатну усунути потребу в посередниках, сторонніх центральних органах і дорого-масштабних процесах.

Однією з цілей даної магістерської дисертації було переконатись і запевнити, що технології блокчейн слід довіряти, адже вона має один з найцінніших, найперконливіших аргументів, - дана технологія побудована на математичних принципах, а математиці варто довіряти.

Було розкрито питання, щодо застосування технології у будь-яких сферах діяльності, починаючи від IT-сфери закінчуючи банківською сферою (фінансовою діяльністю).

В результаті роботи, було створено програму, на мові програмування Java, яка генерує біткоїн-адресу. Вдосконаливши продукт, створивши тим самим власний крипто-гаманець, можна прив’язати його до будь-якого API (*Application Programming Interface*) українського банку та використовувати його як обмінний пункт з цифрової валюти на будь-які фіатні кошти.

Щодо майбутнього, є в планах розвиватись й надалі працювати з технологією Blockchain. Так як в перспективі є можливості у створення систем для smart-контрактів або IoT.

На сьогоднішній день технологія ще не до кінця досліджена, але думаю, поєднавши її з іншими технологіями, навіть з інших галузь, то можна досягти нових, суспільно-корисних результатів.

Список використаної літератури

1. World Wide Web: Біткоїн-терміни [Електронний ресурс] – Режим доступу https://bitcoin.org/uk/vocabulary

2. World Wide Web: Украина переведет все государственные данные на блокчейн [Електронний ресурс] – Режим доступу https://hightech.fm/2017/04/14/us-ukraine-bitfury-blockchain

3. Andreas M. Antonopoulos Mastering Bitcoin: Unlocking Digital Cryptocurrencies / Andreas M. Antonopoulos – K.: NGITS, 2014.

4. Don Tapscott, Alex Tapscott Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World / Don Tapscott, Alex Tapscott Blockchain – K. : Information Systems, 2016.

5. World Wide Web: Технологія блокчейн та страхування [Електронний ресурс] – Режим доступу: http://www.insa.com.ua/uk/blog/tehnologiya-blockchain-i-strahovanie/

6. World Wide Web: Blockchain: Приклади використання технології на практиці [Електронний ресурс] – Режим доступу: https://stocx.org/blockchain-tehnologii-na-praktitsi/

7. World Wide Web: Bitcoin Wiki [Електронний ресурс] – Режим доступу: http://ru.bitcoinwiki.org - Дата доступу: 30.05.2017.

8. Melanie Swan Blockchain: Blueprint for a New Economy / Melanie Swan - K. : Economic, 2015.

9. World Wide Web: Основы блокчейн и криптовалюты [Електронний ресурс] – Режим доступу: https://www.intuit.ru

10. World Wide Web: Введение в криптовалюты и блокчейн [Електронний ресурс] – Режим доступу: https://www.intuit.ru

11. World Wide Web: Биткоин и технологии криптовалюты [Електронний ресурс] – Режим доступу: https://www.intuit.ru

12. World Wide Web: Навчальні відео-матеріали від компанії Distributedlab [Електронний ресурс] – Режим доступу: https://distributedlab.com/

13. World Wide Web: Майнінг. Процес майнінгу [Електронний ресурс] – Режим доступ https://pingblockchain.com/majning-proces-majningu/

14. Paul Vigna The Age of Cryptocurrency: How Bitcoin and the Blockchain Are Challenging the Global Economic Order K.: Economic. 2016

15. World Wide Web: Bitcoin Forum, [Електронний ресурс] – Режим доступу: https://bitcointalk.org

16. World Wide Web: Pandia: Вибір криптогрфічно стійких еліптичних кривих, [Електронний ресурс] – Режим доступу http://pandia.ru/text/80/176/51044.php

17. World Wide Web: Лекториум, Криптографические Хэш-функции [Електронний ресурс] – Режим доступу: https://www.lektorium.tv/course/22746

18. World Wide Web: BitNovosti: Математика Биткойна: Теория, [Електронний ресурс] – Режим доступу: https://bitnovosti.com/2014/10/23/bitcoin-math/

19. World Wide Web: habr: Хэш-алгоритмы: [Електронний ресурс] – Режим доступу: https://habr.com/post/93226/

20. Nathaniel Popper Digital Gold: Bitcoin and the Inside Story of the Misfits and Millionaires Trying to Reinvent Money K. : Economic, 2016

Додаток. Текст програми

```
import java.io.UnsupportedEncodingException;
import java.security.*;
import java.security.interfaces.ECPrivateKey;
import java.security.interfaces.ECPublicKey;
import java.security.spec.ECGenParameterSpec;
import java.security.spec.ECPoint;


/**
 * @author Владимир Ткачук
 */
public class Thesis {
    public static void main(String[] args) throws NoSuchAlgorithmException,
InvalidAlgorithmParameterException, UnsupportedEncodingException,
NoSuchProviderException {



        KeyPairGenerator keyGen = KeyPairGenerator.getInstance("EC");
        ECGenParameterSpec ecSpec = new ECGenParameterSpec("secp256k1");
        keyGen.initialize(ecSpec);
        KeyPair kp = keyGen.generateKeyPair();
        PublicKey pub = kp.getPublic();
        PrivateKey pvt = kp.getPrivate();


        ECPrivateKey epvt = (ECPrivateKey) pvt;
        String sepvt = adjustTo64(epvt.getS().toString(16)).toUpperCase();
        System.out.println("s[" + sepvt.length() + "]: " + sepvt);



        ECPublicKey epub = (ECPublicKey)pub;
        ECPoint pt = epub.getW();
        String sx = adjustTo64(pt.getAffineX().toString(16)).toUpperCase();
        String sy = adjustTo64(pt.getAffineY().toString(16)).toUpperCase();
        String bcPub = "04" + sx + sy;
        System.out.println("bcPub: " + bcPub);



        MessageDigest sha = MessageDigest.getInstance("SHA-256");
        byte[] s1 = sha.digest(bcPub.getBytes("UTF-8"));
        System.out.println("  sha: " + bytesToHex(s1).toUpperCase());


        MessageDigest rmd = MessageDigest.getInstance("RipeMD160", "BC");
        byte[] r1 = rmd.digest(s1);

        byte[] r2 = new byte[r1.length + 1];
        r2[0] = 0;
        for (int i = 0 ; i < r1.length ; i++) r2[i+1] = r1[i];
        System.out.println("  rmd: " + bytesToHex(r2).toUpperCase());




        byte[] s2 = sha.digest(r2);
        System.out.println("  sha: " + bytesToHex(s2).toUpperCase());
        byte[] s3 = sha.digest(s2);
        System.out.println("  sha: " + bytesToHex(s3).toUpperCase());
```

```
        byte[] a1 = new byte[25];
        for (int i = 0 ; i < r2.length ; i++) a1[i] = r2[i];
        for (int i = 0 ; i < 5 ; i++) a1[20 + i] = s3[i];

        System.out.println("  adr: " + Base58.encode(a1));

    }
    public static String bytesToHex(byte[] in) {
        final StringBuilder builder = new StringBuilder();
        for(byte b : in) {
            builder.append(String.format("%02x", b));
        }
        return builder.toString();
    }
    private static String adjustTo64(String s) {
        switch (s.length()) {
            case 62:
                return "00" + s;
            case 63:
                return "0" + s;
            case 64:
                return s;
            default:
                throw new IllegalArgumentException("not a valid key: " + s);
        }
    }


}
```