

# **В-Hydra: Белая книга**

**Автор: MRX**

В-hydra- это одноранговая электронная кассовая система P2P.

Ключевые особенности :

Децентрализованная архитектура : Платформа работает без центрального сервера, обеспечивает полную независимость.

Простота использования : Интуитивно понятный интерфейс и легкая связь с другими группами.

Надёжность и безопасность : Испол

Масштабируемость : Возможность расширения сети для работ с указанными объемами данных.

## **1. Введение.**

В-Hydra — это инновационная технология децентрализованной передачи данных, основанная на D2D (Device-to-Device) связи через Wi-Fi и Bluetooth. Проект стремится обеспечить автономную и безопасную систему передачи информации без необходимости в централизованных серверах. Эта система может быть использована в различных сферах, включая финансы, медицину и умные города.

## **2. Техническая архитектура.**

### **2.1 Узлы сети**

В-Hydra использует узлы (устройства), которые связываются между собой напрямую, создавая динамическую P2P-сеть. Взаимодействие между узлами строится на основе следующих компонентов:

Wi-Fi и Bluetooth для передачи данных.

Кибертронский хеш (SHA-512) для защиты информации.

Автоматическая маршрутизация на основе состояния сети.

Энергоэффективные алгоритмы для минимизации потребления ресурсов.

### **2.2 Механизм работы узлов**

Узел сканирует ближайшие устройства для установления связи.

При нахождении доступного узла устанавливается зашифрованное соединение.

Передача данных осуществляется по динамически изменяемым маршрутам.

В случае сбоя система автоматически перенаправляет пакеты данных.

### 2.3 Графическое описание работы.

Система B-Hydra состоит из следующих ключевых компонентов:

Блоки данных: информация, передаваемая между узлами.

Узлы связи: устройства, обеспечивающие связь между блоками данных.

Динамические маршруты: адаптивные пути передачи информации.

Процесс передачи данных:

Исходный узел → генерирует блок данных и ищет ближайший узел для передачи.

Передача данных → зашифрованный пакет передается ближайшему доступному узлу.

Роутинг → узел определяет оптимальный маршрут и передает данные дальше.

Получение данных → конечный узел расшифровывает данные и подтверждает получение.

Диаграмма процесса работы:

Исходный узел → Узел А → Узел В → Узел С → Конечный узел  
(шифрование) (маршрутизация) (перенаправление) (дешифровка)

**2.4 Proof of Contribution (PoC)** — это механизм консенсуса, основанный на вкладе участника в развитие сети или экосистемы. В отличие от Proof of Work (PoW) или Proof of Stake (PoS), где основными факторами являются вычислительная мощность или объем стейка, в PoC учитываются реальные действия, приносящие пользу системе, например:

- разработка и улучшение кода;
- участие в тестировании;
- создание контента и распространение информации о проекте;
- предложения по улучшению (governance);
- активность в сети (например, поддержка узлов, валидаторов и т.д.).

Система баллов или токенов начисляется на основе определённых метрик вклада. Эти метрики могут быть как автоматическими (например, количество пул-реквестов в репозитории), так и полуавтоматическими (одобренными децентрализованным сообществом или DAO).

Преимущества PoC:

Мотивация к реальному участию в развитии проекта.

Снижение энергозатрат по сравнению с PoW.

Повышение прозрачности и справедливости распределения вознаграждений.

Более активное и вовлечённое сообщество.

Применение в нашем проекте:

PoC используется в сочетании с [вставьте другие алгоритмы, если применимо], чтобы обеспечить устойчивое и справедливое распределение ресурсов, поощряя как технический, так и социальный вклад в экосистему.

**2.5 Proof of Contribution**, которое можно добавить в технический раздел белой книги:

Механизм работы Proof of Contribution (PoC)

Proof of Contribution реализуется через систему оценки вклада каждого участника. Этот вклад фиксируется и проверяется сетью на основе заранее определённых метрик. Алгоритм состоит из следующих этапов:

1. Сбор данных о вкладе

Каждый участник выполняет действия, которые считаются полезными для проекта. Это может включать:

Коммиты в репозиториях (GitHub/GitLab);

Написание или перевод технической документации;

Проведение тестирования или нахождение багов;

Голосование за предложения развития (если есть DAO);

Участие в обучающих или промо-активностях.

2. Оценка и верификация вклада

Все действия участников проверяются системой на подлинность:

Автоматически: через API-интеграции (например, с GitHub);

Полуавтоматически: с помощью валидаторов сообщества;

Через DAO-голосование: когда вклад требует оценки (например, качество предложенной идеи).

3. Присвоение рейтинга/баллов

Каждому действию соответствует определённое количество баллов, формируя индивидуальный рейтинг вклада (Contribution Score).

Примеры:

1 пул-реквест = 10 баллов;

Успешное исправление ошибки = 20 баллов;

Принятая статья в блог = 15 баллов.

#### 4. Вознаграждение

На основе накопленных баллов участники получают:

Токены проекта;

Право голоса в управлении;

Повышенный приоритет доступа к ресурсам сети (например, ноды, API и т.д.).

#### 5. Обнуление и перезапуск (по желанию)

Система может обнуляться по сезонам (месяц, квартал), чтобы стимулировать постоянную активность и снизить пассивное накопление рейтинга.

## 2.6 Механизм обработки транзакций.

Транзакция в системе фиксирует перемещение ценности или действий между участниками сети. Это может быть передача токенов, голосование, публикация контента или иной вклад.

Процесс проходит следующие этапы:

##### 1. Формирование транзакции

Пользователь создает транзакцию, например:

Перевод токенов другому участнику;

Подтверждение выполненной задачи;

Отправка голосов в DAO.

Каждая транзакция содержит:

Отправитель и получатель (публичные адреса);

Тип действия (токен, голос, контент и т.д.);

Цифровую подпись (для подтверждения подлинности);

Метку времени.

##### 2. Подпись и шифрование

Транзакция подписывается закрытым ключом отправителя. Это:

Гарантирует, что действие совершил именно он;

Исключает подделку.

##### 3. Отправка в сеть

Транзакция передаётся в узлы (ноды), где:

Проверяется цифровая подпись;  
Проверяется баланс или разрешения;  
Устанавливается очередь на включение в блок.  
4. Подтверждение и включение в блок  
Узлы, работающие по PoS, отбирают транзакции:  
На основе важности и приоритета вклада;  
С учётом рейтинга участника;  
Могут объединять транзакции в блоки по времени или объёму.  
5. Финализация  
После добавления в блок:  
Транзакция считается подтверждённой;  
Изменения становятся частью неизменяемой истории (блокчейна);  
Пользователи видят обновлённый статус (баланс, рейтинг, право голоса и т.д.).

## 2.7 Физико-математическое выражение.

Для V-Hydra можно добавить несколько математических выражений, чтобы показать:

1. Алгоритм маршрутизации (пути передачи):  
Если узел выбирать путь с минимальной стоимостью передачи, можно использовать:

$$C_{total} = \sum_{l=1}^n C_{link}(l) + a * D(l)$$

Где:

$C_{total}$  - общая стоимость маршрута.

$C_{link}(l)$  - стоимость передачи между узлами  $l$  и  $l + 1$ .

$D(l)$  - задержка или загруженность узла  $l$ .

$a$  - коэффициент, регулирующий важность задержки.

2. Шифрование данных (упрощенно):

Если используется модифицированный SHA-512:

$$H = \text{SHA512}(K || M)$$

Где:

$H$  - хеш результата.

$K$  - секретный ключ (уникален для каждого узла).

M - передаваемое сообщение.

|| - операция конкатенации (объединения).

3. Вычисление энергии связи:

$$E_{total} = E( l = 1 \text{ to } n ) P_{tx}( l ) * t_{tx}( l )$$

Где:

$E_{total}$  - общее потребление энергии.

$P_{tx}(l)$  - мощность передачи на этапе l.

$t_{tx}(l)$  - время передаче между узлами l и l + 1.

4. Оценка устойчивости сети:

Модель на основе вероятности связи между узлами:

$$P_{connection} = 1 - e^{( -л * r^2 )}$$

Где:

$P_{connection}$  - вероятность связи между узлами.

л - плотность узлов в пространстве л-это ляба.

r - радиус связи.

2.5 Временной Халвинг до 3000 года

Математическое формула:

$$R(t) = R_0 * (1/2)^{(t - t_0) / T}$$

Где:

$R(t)$  - награда за блок в момент времени t,

$t_0$  - год запуска сети,

T - период в годах, через которые происходит халвинг (например, каждые 4 года), t - текущий год.

```

C++
#include <iostream>
#include <boost/asio.hpp>
#include <d2d1effecthelpers.h>
#include <ded1.h>
#include <dwrite.h>
#include <dwmapi.h>
#include <fstream>
#include <comdef.h>
#include <ctime>
using boost::asio::ip::udp;

int main(){
    try{
        boost::asio::io_context(io_contxt, udp::endpoint(udp::v4(),0));
        std::string message = "D2D!";
        udp::endpoint target(boost::asio::ip::address::from_string("..")8080);

        socket.send_to(boost::asio::buffer(message),target);

        char reply[1024];
        udp::endpoint sender_endpoint;
        size_t reply_length =
socket.receive_from(boost::asio::buffer(reply),sender_endpoint);
        std::cout<<"Reply:"<<std::strng(reply,reply_length)<<std::endl;
    }
    catch(std::exception& e)
    {
        std::cerr<<"Error:"<<e.what()<<std::endl;
    }
    return 0;

    std::wstring fontname = L"Courler";

```

## PYTHON

```

from b_hydra import Blockchain
blockchain = Blockchain() blockchain.add_block(data="Пример
транзакции") print(blockchain.chain)

```

### **3. Экономическая модель.**

В-Hydra может использовать несколько вариантов монетизации:

Токенизированная экономика: создание крипто-токенов для стимулирования участников сети.

Коммерческое лицензирование: предоставление технологий для частных компаний.

Децентрализованные сервисы: платежные и идентификационные решения на основе D2D.

### **4. Алгоритмы и безопасность.**

Для защиты данных используется Кибертронский хеш, построенный на базе SHA-512 с модификациями, повышающими стойкость к атакам. Основные принципы безопасности:

Полное шифрование передаваемых данных.

Механизмы аутентификации для предотвращения подмены узлов.

Защита от перехвата с использованием уникальных ключей шифрования.

### **5. Применение и перспективы.**

В-Hydra может быть использована в:

Финансовой сфере: мгновенные транзакции без посредников.

Медицине: безопасный обмен данными между устройствами.

IoT и умных городах: оптимизация связи между устройствами без интернета.

### **6. Заключение.**

В-Hydra — это новый шаг в развитии децентрализованных систем, обеспечивающий автономность, безопасность и эффективность передачи данных. Внедрение этой технологии позволит создать независимые сети, способные работать без центральных серверов, что делает ее перспективным решением для будущего интернета.

## 7. Литература.

- [1]. Накамото, С. (2008-гг). Bitcoin: Децентрализованная электронная платежная система. <https://bitcoin.org/bitcoin.pdf>
- [2]. Вуд, Г. (2014-гг). Ethereum: Безопасный децентрализованный универсальный реестр транзакций. <https://ethereum.org/en/whitepaper/>
- [3]. Таненбаум, Э. С., & Везеролл, Д. (2010-гг). Компьютерные сети. Pearson Education.  
Столлингс, У. (2017-гг). Криптография и безопасность сетей: Принципы и практика. Pearson.
- [4]. Куроце, Дж. Ф., & Росс, К. У. (2020-гг). Компьютерные сети: Подход сверху вниз. Pearson.
- [5]. Шнайер, Б. (1996-гг). Прикладная криптография: Протоколы, алгоритмы и исходный код на С. John Wiley & Sons.
- [6]. А. Бэк, (2002-гг) "Hashcash - мера отказа в обслуживании ," <http://www.hashcash.org/papers/hashcash.pdf>,  
.
- [7]. Глушков В.М. (1960-1980-гг). Проекта создания автоматизированной системы управления советской СССР экономикой (ОГАС ).
- [8]. Китов А.В. (1958-гг). "Вычислительные центры должны быть связаны в Единую систему автоматической информационной и вычислительной службы".
- [9]. Бондаренко Н.Н. (2018-гг). " Именно это мы говорим им в глаза от твоего имени! Именно это ты и увидишь на канале дневник депутата!".
- [10]. Делягин М.Г. (2022-гг). Экономика и депутаты Госдумы от фракций " Справедливая Россия-за Правду".