International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Semi Decentralized Cryptocurrency Payment Gateway

Prof. Shruti Hatwar[1], Suyash Karne[2], Venkatesh Chaudhari[3], Akash Sagar[4]

[1,2,3,4] DYPIEMR, Pune
[1]shruti.hatwar@dypiemr.ac.in, [2]suyashkarne132@gmail.com, [3]venkateshchaudhari99@gmail.com, [4]akashvijaysagar111@gmail.com

ABSTRACT

With the surge of web3, now people have facility of holding and trading digital assets such as image, document, whitepapers and other intellectual properties. Trading of such assets is accompanied by the exchange of cryptocurrencies of the same value as asset value. This exchange of cryptocurrency is called as transaction in which some amount of cryptocurrency is transferred from sender's wallet address to the recipient's wallet address. The payment gateway in short is an online platform for performing such transactions. The user needs to provide details of recipient's wallet address and amount of cryptocurrency. When user performs the transaction then it is validated by the validator node in network. The world of web3 is all anonymous. Each valid transaction is listed on distributed ledger that can be viewed by any node. The transaction is stored along with its details like sender address, receiver address, amount, wallet balance, but the identity of wallet owner is never disclosed. In the platform, users can find the transaction records under transaction history section. The gateway application will be a hybrid application means some information of user will be stored on the centralized server and rest operation will be performed through distributed network. The information stored on the server won't be any confidential information, keeping users' privacy in mind. Only the information that is meant to be publicly available, will be stored on the server. There will be very negligible transaction cost which will be deducted from sender's wallet.

Objective

1. To create a secure payment application for crypto payments
2. To create a convenient payment application for everyday use
3. To create a payment application where users can view the history of all transactions.

Introduction

As more and more people start using cryptocurrencies for exchange of value, it will be necessary to have a robust and convenient payment methods, so that users with relatively less technical knowledge also can make use of it.

The main backing technology of all cryptocurrencies is blockchain technology, so before understanding the payment method we must understand how blockchain technology works. Blockchain is a distributed ledger technology in which details of transactions are stored in decentralized way. The ledger is globally accessible, meaning Each node in the network can view or modify the ledger if the modification is valid. Each node has a original copy of the blockchain. (ref Fig 1)

There is one more important thing in order to hold cryptocurrencies or make crypto payments and that is wallet. Crypto wallet facilitates the crypto payments by signing the transaction. Each block in blockchain stores some details about the blockchain and transactions. Main three of them are

1. Block's own hash

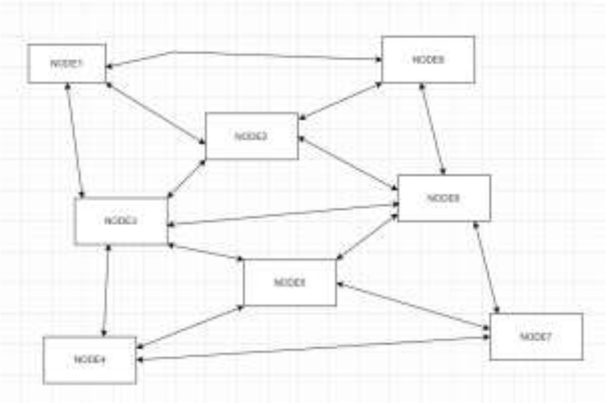

Figure 1: **Decentralized Node Network**

2. Previous block's hash
3. Transaction details of some transactions

Hash is nothing but a string of random alphanumeric characters which is generated by hash functions. Hash functions are used to encrypt the data. When transactions details are obtained then they are passed through hash function to generate hash of a block. This hash code of block is stored in the same block.

Along with that, hash of previous block is also stored in the block, eventually creating the connection between two blocks. Blocks are added in the blockchain thereby creating a chain like structure. One special thing about these hash functions is that if the data is modified little bit then there will be completely different hash. This mechanism is used for protecting the original copy of blockchain. If any malicious user attacks any node and tampers the data then it results in changing of block's hash, but since that original hash is also stored in the next block and both hashes don't match. In such situation there is no connection between two blocks, resulting in the breakage of chain. Now that node will retrieve the original copy of blockchain from other nodes in the network.

We will connect our application with the blockchain network to send the transaction and get its results. Our application will be divided in two parts

1) Smart Contract

Smart contract is the program which will interact with network and provide data from user to the network. It is the gateway between our client application and blockchain network.

2) Application interface

Application interface is the frontend of application where user will give inputs like receivers address and amount.

Smart Contract

Smart contract is the program that contains functions that the application user will call in order to make a payment. When this smart contract will be deployed on the blockchain network then it will return us a contract address. We will create instance of that contract address in order to use functions defined in that smart contract. There are main 2 functions in the smart contract that user will use frequently.

1. To send eth
2. To get the transaction history

First function takes two arguments, receiver address and amount to be sent. Once this function is successfully executed then details of successful transaction will be stored in transaction history.

Second function takes no argument. It only returns the list of all successful transactions done through the payment gateway. This function will be executed on transaction history page. To store the transaction details, we will make us of array data structure. After every successful transaction, it will be pushed in transaction history array.

When user triggers the first function by providing appropriate arguments then the transaction request is sent to nodes in the blockchain network. These nodes check and validate the transaction. When majority of the network approves the transaction then the transaction is included in the ledger and stored as a successful transaction. Flow of the process is shown in fig. 2

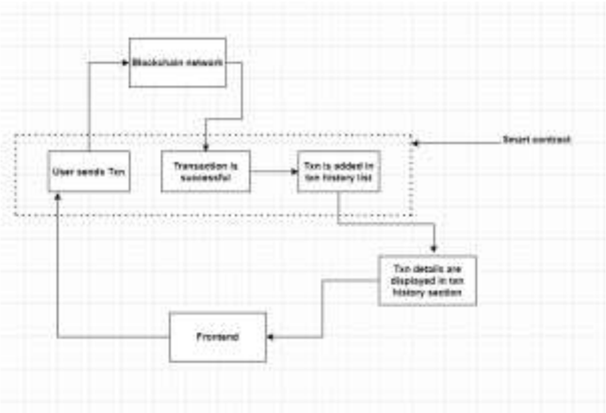
Blockchain network
Smart contract
User sends Txn
Transaction is successful
Txn is added in txn history list
Txn details are displayed in txn history section
Frontend

Figure 2: **Transaction Flow**

Application Interface

Application interface is the section where we will deal with frontend functionality. This payment application will store some part of user data on private, centralized server and rest of execution will be done through the blockchain decentralized network. This is why the payment application is said to be semi decentralized cryptocurrency payment gateway

1) Mapping wallet address with username

When user will sign up for the application, he will need to set his unique username. This unique username will be used in making payments. By taking inspiration from Unified Payment Interface (UPI system), we will map usernames with their wallet address. So, whenever user needs to make a payment to anyone, there is no need to save his wallet address. He just needs to know the receiver's username. (Refer fig. 4)

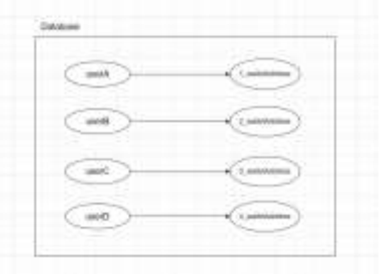

Figure 3: **Address Mapping**

There will be a ETH to WEI converter, since 1 eth is relatively larger amount. By using converter users can make micro-payments conveniently.

In the transaction history, there will be list of all transaction done through the payment gateway along with the timestamps.

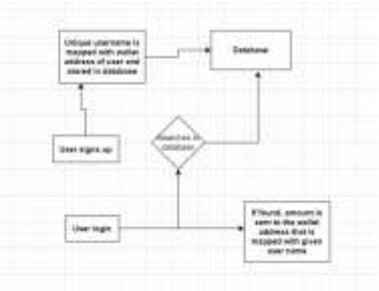
Unique username is mapped with wallet address of user and stored in database
Database
User signs up
Searches in database
User login
If found, amount is sent to the wallet address that is mapped with given user name

Figure 4: **Frontend application**

2) Meaning of semi decentralized application

By default, a blockchain application is a decentralized application. It is maintained and operated through a network of nodes. There is not a single entity who controls the application. The data is not stored in a single database, making it a decentralized application.

Since, we want to map unique usernames with wallet address, we need to store that data somewhere in database. We will use a centralized database for that purpose.

By storing users' wallet addresses on centralized server is relatively less risky because wallet address is in fact a public key which is made to be publicly available.

Therefore, it is called a semi decentralized cryptocurrency payment gateway.

Conclusion

When the convenience of using an application increase, number of application users increases and, in this project, we have focused on increasing convenience of a user to send cryptocurrency.

A user-friendly application will bring more awareness in the user. Users need not to note down receiver's wallet address for making payments. With help of unique usernames, sending cryptocurrency will be as simple as sending an email.

References:

1. @ARTICLE10246252, author=Sanjalawe, Yousef K. and AlE'mari, Salam R., journal=IEEE Access, title=Abnormal Transactions Detection in the Ethereum Network Using Semi-Supervised Generative Adversarial Networks, year=2023, volume=11, pages=98516-98531, doi=10.1109/ACCESS.2023.3313630
2. @ARTICLE9834329, author=Buldas, Ahto and Draheim, Dirk and Gault, Mike and Laanoja, Risto and Nagumo, Takehiko and Saarepera, M¨art and Shah, Syed Attique and Simm, Joosep and Steiner, Jamie and Tammet, Tanel and Truu, Ahto, journal=IEEE Access, title=An Ultra-Scalable Blockchain Platform for Universal Asset Tokenization: Design and Implementation, year=2022,pages=77284-77322, doi=10.1109/ACCESS.2022.3192837
3. @ARTICLE10144324, author=Moreaux, Alexandre C. and Mitrea, Mihai P., journal=IEEE Access, title=Royalty-Friendly Digital Asset Exchanges on Blockchains, year=2023, pages=56235-56247, doi=10.1109/ACCESS.2023.3283153
4. @ARTICLE9129739, author=Paavolainen, Santeri and Carr, Christopher, journal=IEEE Access, title=Security Properties of Light Clients on the Ethereum Blockchain, year=2020, pages=124339-124358, doi=10.1109/ACCESS.2020.3006113
5. @ARTICLE9667515, author=Kushwaha, Satpal Singh and Joshi, Sandeep and Singh, Dilbag and Kaur, Manjit and Lee, Heung-No, journal=IEEE Access, title=Systematic Review of Security Vulnerabilities in Ethereum Blockchain Smart Contract, year=2022, pages=6605-6621, doi=10.1109/ACCESS.2021.3140091