DARKNESS

I am a flyfire drift in the dark ----flyfire

Statement 1:The technology in this document is just for justice and legal purposes.

Statement 2: This document is built on the Bismuth cryptocurrency blockchain, more detail: www.bismuth.cz

1. Creed

a. Freedom

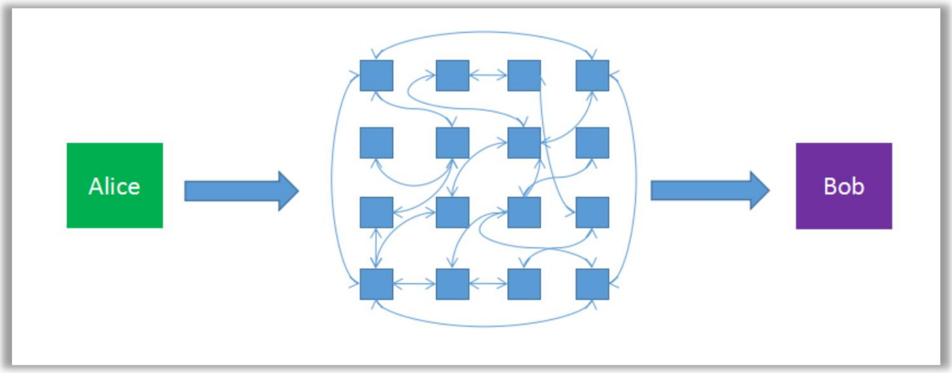
b. Equality

c. Simple is beautiful

d. Not violating laws and regulations

2. The structure of the DARKNESS

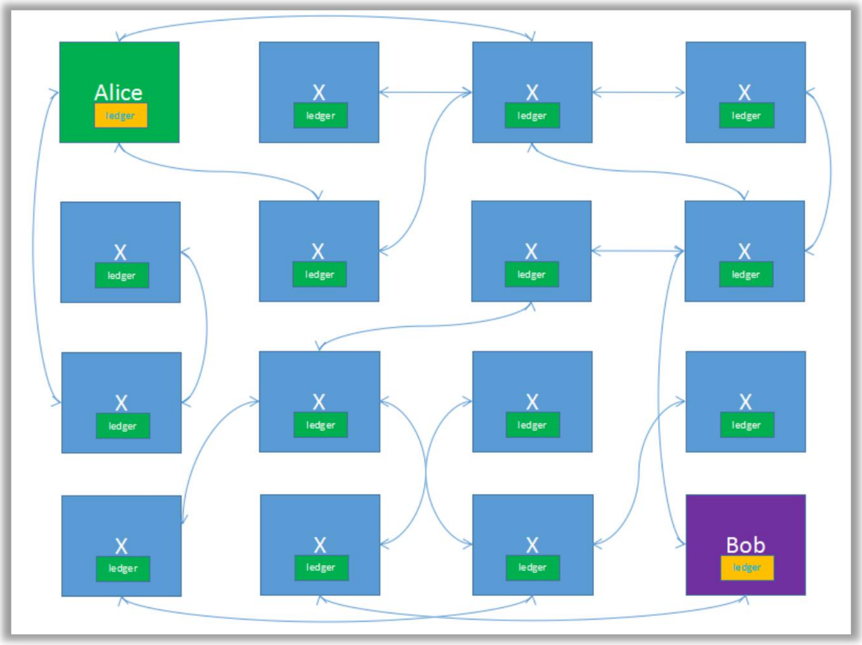
2.1 Traditional network



P1: Traditional network

In the traditional network, data is transmitted in the network space with the aid of routing mechanism.

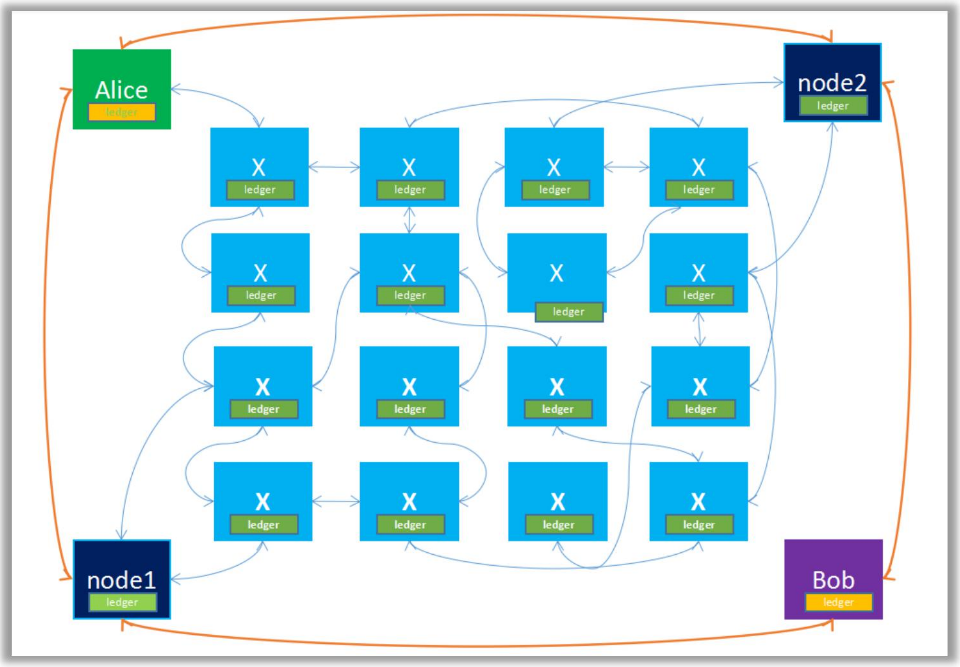
2.2 Block chain network



P2:Block chain network

The traditional network layer by constructing block chain network, data distributed storage, using encryption algorithm for data verification, to ensure data consistency and accuracy.There are two problems, one is that the distributed database usually records only the transfer information, and the occupancy of the database will increase with the increase of the number of transactions. The two is the problem of network synchronization. Alice has recorded a transfer in the local area. The synchronization time of data synchronization to Bob is uncertain, and how many layers of network are uncertain.

2.3 The network of DARKNESS



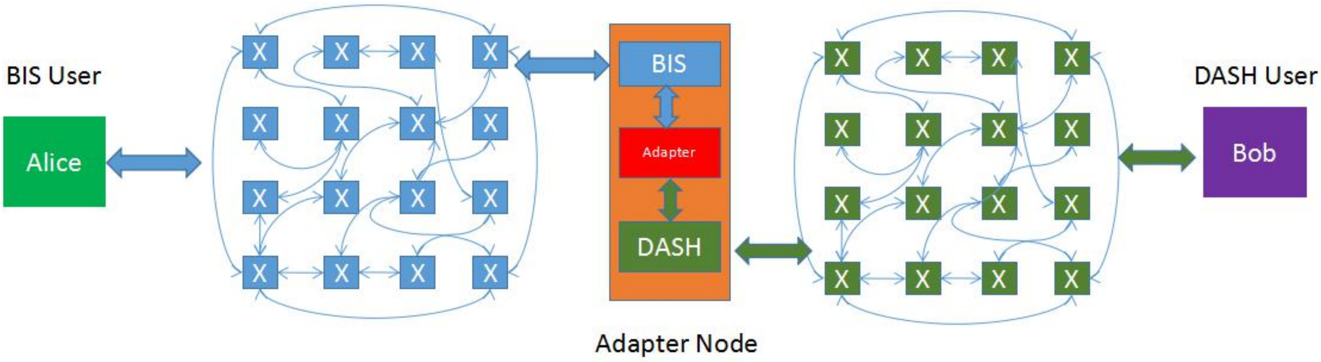
P3:The network of DARKNESS

First Alice transfer to Bob, set the openfield "get ticket" in the information (equivalent to buy ticket), Bob received the transfer， then transfer to Alice for change, which the openfield is set to "ticket:IP_list", IP address can be one or more.

Then Alice get the master node information, next registered to the master node, and then send a message to the master node , I have a message to Bob. The master node then stores the message. Bob registers and gets its own message from the master node ,so that the Bob receives the message sent by the Alice. If the message of the Alice is too large, the message splitting operation can be carried out and then sent through different main nodes.

Through the above mechanism, the size of ledger.db can be reduced, while the network layer between Alice and Bob is reduced. The messages can not be stolen with the aid of block chain technology, and the P2P encryption.

2.4 The extension of the DARKNESS



P4：The extension of the DARKNESS

When DARKNESS is formed in a cryptocurrency, the next step is the extension of the night. Different blockchain can be linked into a huge night network with Adapter Node. Adapter Node assumes the role of exchange rate conversion and information conversion.This depends on the specific implementation of cryptocurrency, as well as information encryption.Because DARKNESS is P2P encryption, so the function of the Adapter node is not complicated.

3. Interactive process

Master node: A node that plays a crucial role in the blockchain must be a public IP address and must be stable. The master node can obtain a certain amount of compensation according to the service volume.

Other nodes: nodes that are not regularly opened in the blockchain or unstable nodes, such as individual wallet or personal node, may have no public IP.

3.1 The process of Master node

① start listening, waiting for Alice or Bob registered

② Alice register

③ Master node generates a challenge code, which can be a random number

④ The master node queries Alice's public key in ledger.db, encrypts the challenge code with this public key, and sends it to Alice

⑤ The master node receives the challenge code verification information returned by Alice and decrypts the encrypted challenge code by using its own private key. If the challenge code is the same as the challenge code sent, Alice considers the registration successful and sends the result to Alice

⑥ If Alice registers successfully, listens for command sent by Alice. If Alice fails to register, she sends the failed result and disconnects, and the process ends

⑦ If Alice sends the command "sendmsg", the master node receives Alice's message, which contains the Bob’s address , the sequence information of the message, the content of the message.The sequence of the message and the content of the message are all encrypted.Sequence format is as follows:

Index of the message fragment / total length of the message / UUID of the message

If the message is stored successfully, a successful result is sent to Alice, otherwise the failed result is sent, then disconnected and the session ends.

⑧ If you receive a 'getmsg' command in step 6, send all of Alice's locally stored messages to Alice, then receive the response, disconnect if successful, and close the session.

⑨ Other command functions have not yet added, follow-up optimization and supplement. Mainly contains "query Bob master node information", "query master node status information" and so on

3.2 The process of other node

The process of Alice：

① When Alice starts up, first queries Bob’s transfer record. If there is a ticket, he can directly connect to the master node and exchange messages with Bob. If there is no ticket, he needs to transfer money to Bob to purchase the ticket. The record field of the ticket is "get_ticket" in the record of the ticket, and "ticket: IP_list" is carried in the transfer (change) information returned by Bob. The IP_list here can be one or more than one, suggested for more than one, reliability considerations.

② After Alice obtains the master information used by Bob, Alice connects to the master node, sends a "register" request.

③ Then send the "sendmsg" command to the master, next send Bob's address to the master, telling the master that I want to send a message to Bob. Bob's publick_key from ledger.db to encrypt the message. Then according to the size of the encrypted message fragmentation, The format of each slice is as follows:

Index of the message fragment / total length of the message / UUID of the message

The sliced sequence also needs to be encrypted using Bob's publick_key

Because Alice connects multiple masters at the same time, different shards can be sent randomly through multiple nodes.

④ Receive the response returned by the master node, if successful, then disconnect and close the session. If it failed, try send it to other master nodes until success.

The process of Bob：

① After Bob starts, he first looks up change_ticket transfer get the information of master node, then connect to the master node.

② Bob registers

③ Then send "getmsg" command to the master node, and receive the master node response.

④ After receiving the message from each master node, the message is assembled and decoded, according to the sequence of each message.

3.3 Abnormal scenes

The master node problems：

Backlog of messages, number of connections reached limit, message sent over time, connection timed out.

The other node problems：

Failed to connect to the master node, all of Bob's master nodes failed to connect, failed to receive the message.

Abnormal scene analysis is still ongoing.

3.4 FAQ

1. The master node how to profit?

   Currently Bismuth does not yet have a master node, but there is a master node plan.
2. The message sent is safe?

   Secure, because it's encrypted P2P, no one can interpret the message content without Bob's private key.
3. Tickets price?

   The price of the ticket is set by the interviewee, If the fare is too low, you can deny access.

4. Application scenarios

4.1 The original intention of the project

Because the yellow and violent content often pops up when online , as a father, I do not want my children premature exposure to these unhealthy content, if use the DARKNESS architecture, only bought a ticket to view the contents of the Internet can be solved this problem.

First of all, if you need to browse other sites you need to buy tickets, the password must be in the hands of adults

Second, the content on the ticket buying site is manageable and there is no unhealthy content. If there is, can be local shielding, in the background of the DARKNESS will design a ban on access list.

My original intention was simple, just wanted to purify the spam on the web.

4.2 DARKNESS applicable scenes

website：

This is my original intention

With the DARKNESS architecture can built website on any node , and all the contents of the site encrypted(end-to-end encryption).In this architecture search engine is very small, if you want to build a search engine, the high ticket fees (ticket costs set by the website provider) will make the search engine can not be maintained.Second, search engines can not guarantee that access to the content is not modified, because it is end-to-end encryption, so what you see may not be the same as what I see.

chat：

Although this feature can be implemented with the openfield field in Bismuth's transfer, there is a handling fee per transfer, resulting in high cost of chatting.And this would increase the size of Bismuth's ledger.db, which would be detrimental to the crypto-currency's synchronization.Second, the cryptocurrency transfer time is uncertain because we do not know when to synchronize to local nodes.

IOT and PaaS：

IOT is a very hot topic, the DARKNESS architecture can easily organize your own public cloud, only a small investment can be, of course, the night architecture can also be used in the private cloud.Because message through the master node, and Alice and Bob are separated by only one layer, so the speed will be much faster.There are a lot of things related to network configuration in the process of cloudification. However, based on the DARKNESS architecture, IP can be changed at will. As long as the application address and private key exist, it is the previous node. You do not need to change the configuration file such as IP address, because the author is currently Engaged in cloud operation and maintenance industry, knowing that inside the bitterness, the use of this structure on the operation and maintenance is a great convenience.

Other：

DARNESS plainly provides a high-level messaging mechanism, based on this mechanism，the existing website, all kinds of cloud services can be easily ported, and security, after all, the currency encryption Algorithms have existed for nearly 10 years.

4. Roadmap

a. 2017.12 Demo -----Done
https://github.com/flyfire100/DARKNESS

a. 2018.03 Develop a Framework and API

b. 2018.06 Based on the DARKNESS to develop a chat program

c. 2018.12 Based on the DARKNESS to develop a Browser

d. 2019.12 Provide web site deployment capabilities, increasing ease of use

e. 2020.12 Adapter Node

5. Donate

BTC: 1GuoUm1mqd8pwXRee22BkATnEMtpioRiRV
ETH: 0xef56e1429b1ff7d1885e4056e7da3477d84c0b13
BIS: f69b7621d402f4ca17a679adf692b98130003019d968c643700f625e