BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF INFORMATION TECHNOLOGY
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF INFORMATION SYSTEMS
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

PROCESSING OF THE BLOCKCHAIN EMPLOYING IPFS

VYUŽITÍ IPFS PRO ZPRACOVÁNÍ BLOCKCHAINU

MASTER'S THESIS
DIPLOMOVÁ PRÁCE

AUTHOR **Bc. MATÚŠ MÚČKA**
AUTOR PRÁCE

SUPERVISOR **Ing. VLADIMÍR VESELÝ, Ph.D.**
VEDOUCÍ PRÁCE

BRNO 2020

Abstract

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v anglickém jazyce.

Abstrakt

Do tohoto odstavce bude zapsán výtah (abstrakt) práce v českém (slovenském) jazyce.

Keywords

Sem budou zapsána jednotlivá klíčová slova v anglickém jazyce, oddělená čárkami.

Klíčová slova

Sem budou zapsána jednotlivá klíčová slova v českém (slovenském) jazyce, oddělená čárkami.

Reference

MÚČKA, Matúš. *Processing of the Blockchain Employing IPFS*. Brno, 2020. Master's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Ing. Vladimír Veselý, Ph.D.

Processing of the Blockchain Employing IPFS

Declaration

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana X... Další informace mi poskytli... Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Matúš Múčka
November 26, 2019

Acknowledgements

V této sekci je možno uvést poděkování vedoucímu práce a těm, kteří poskytli odbornou pomoc (externí zadavatel, konzultant apod.).

Contents

Chapter 1

Introduction

HTTP is „good enough“ for the most use cases of distributing files over the network (like webpages, etc.). But when we want to stream lots of data to multiple connected clients at once, we are starting to hittings its limits. When two clients are requesting the same data, there is no mechanism in HTTP that would allow sending the data only once. Sending duplicate data has become a problem in large companies because of bandwidth capacity. Blizzard [1] started to distribute video game content by distributed solution because it was cheaper for the company and faster for players [1]. Linux distributions use BitTorrent to transmit disk images [2].

The bitcoin blockchain has now 242 gigabytes [3]. When blockchain is processed (parsed address, created search indexes), the size on a disk can doubles. If there are multiple blockchains, then data can have few terabytes. When we are sharing blockchains data from the server for several clients, there is a big chance that multiple clients want the same data. They may be working on the same case and investigating the same wallets. So in standard solution with relational database and some HTTP server, for every request server have to search in all data (that can have a size of few terabytes) and transmit selected data to the client. This happens even if a different client asks for the same data in a few minutes ago. Behaviour mentioned above dramatically limits the scalability of the server.

Services that are using HTTP, often have client-server architecture, so there is also a problem with one point of failure. If the server for some reason stops working, the client can not receive data. In distributed file system such as IPFS, there is no such problem as one point of failure, because all data are duplicated on multiple clients.

[1] Game company

[2] Image of Debian downloadable by BitTorrent https://www.debian.org/CD/torrent-cd/

[3] Current size of bitcoin blockchain can be seen at https://www.statista.com/statistics/647523/worldwide-bitcoin-blockchain-size/

Chapter 2

Cryptocurrencies

2.1 Ethereum

2.2 Bitcoin

2.3 DigiByte

2.4 Decred

2.5 Monero

Chapter 3

IPFS

IPFS stands for InterPlanetary File System and is a peer-to-peer distributed filesystem designed to make the Web faster, safer, and more open. In contrast with standard filesystems, objects in IPFS are content-addressed, by the cryptographic hash of their contents. In the case of the standard Web, when user wants some file, he needs to know on which server is a file located and the full path to the file. In IPFS user needs only to know the hash of the requested file. He does not care about the location of the file.

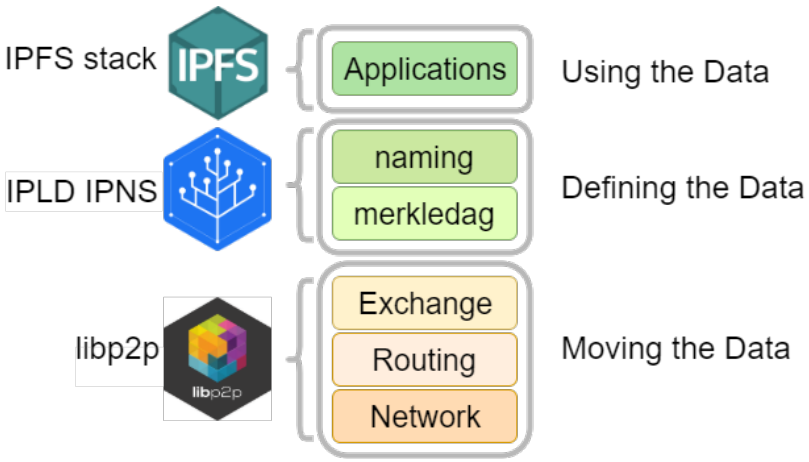

Figure 3.1: IPFS stack

3.1 CID

3.2 IPFS

3.2.1 Cluster

3.2.2 Node

3.3 IPLD

3.3.1 Formats

3.3.2 Routing

3.3.3 Exchange

3.3.4 Objects

3.3.5 Files

3.3.6 Naming (IPNM)

3.3.7 Distributed hash table

Chapter 4

Design

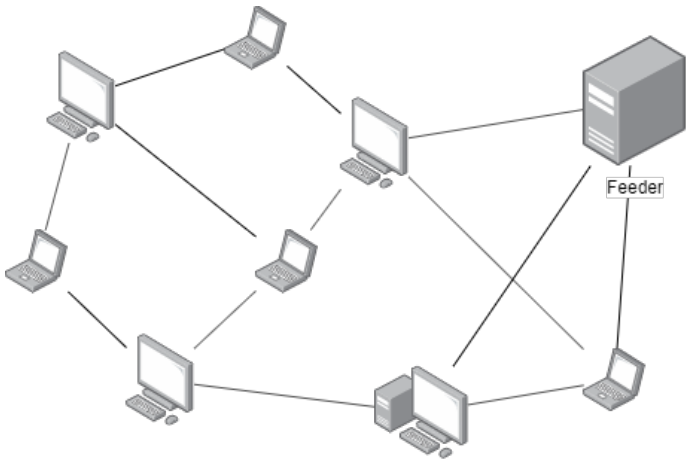

Figure 4.1: System design

4.1 Feeder

4.2 Client application

Chapter 5

Implementation

Chapter 6

Testing

6.1 Testing enviroment

6.2 Testing

Chapter 7

Conclusion

Bibliography

[1] Kalle, K. *Big data in video games*. Lappeenranta, FI, 2017. Bakalářská práce. Lappeenranta University of Technology, School of Business and Management, Computer Science. Available at: http://lutpub.lut.fi/handle/10024/147666.