Ethereum Gateway: Domain Name Service on Ethereum

Along with a local dns server that communicates with the blockchain

Sung-Shine Lee
University of Waterloo
martinetlee@gmail.com

ABSTRACT

Domain Name Service helps translating IP and domain name. Despite of its wide adoption, the original design has several security vulnerabilities and problem of centralization. Recently, several blockchain based DNS was designed to solve centralization. However, not much is discussed of its efficiency and burden laid on the network. In this paper, a Domain Name Service was built on top of the Ethereum blockchain using smart contracts and deployed on Ropsten testnet. A prototype of local dns server were also provided to identify specific blockchain related DNS requests and resolve it locally by communicate with the blockchain. Lastly, We perform some tests and compare blockchain-based DNS with traditional DNS.

1 INTRODUCTION

Domain Name Service(DNS) provides a one-to-one binding between an IP and a domain name. This makes it easier for Human to address resources on the Internet and became one of the fundamental blocks of the Internet that boosted its popularity.

However, the extremely popular protocol that is embedded in almost every device wasn't designed to be secure. The content of the dns request is sent in clear text without authentication allowing malicious adversaries to eavesdrop or perform man-in-the-middle attacks. Its hierarchical and centralized administrative has the problem of a single point of failure. Moreover, it is subject to Distributed Denial of Service(DDoS) attacks.

To make DNS secure, protocols like DNSSEC[11] was proposed on top of DNS to address problems such as authenticity of a DNS request and response. On the other hand, decentralizing DNS has also always been an important direction of research. Cox et al. proposed a distributed DNS using Chord [15] to store DNS. However, they've faced severe problems in security and couldn't devise a solution of the proof-of-ownership.

Lately, the developing blockchain technology started from bitcoin[14] provides essentially a trusted decentralized distributed ledger to solve the problem above. Naturally, decentralization of DNS using blockchain has also been proposed. Namecoin, being the first blockchain-based DNS, is a fork from bitcoin and is created in hope to replace the DNS root servers. Blockstack, on the other hand, is made on top of bitcoin and treats bitcoin as data storage layer.

In this paper, we implemented a Domain Name Service on top of the Ethereum blockchain along with a local DNS server that allows client to access the blockchain data locally. This implementation retains privacy since DNS queries will not generate out-going packet. DNS response is also much faster than traditional DNS response. Using this, we perform empirical tests to compare the blockchain based DNS with the traditional DNS.

2 BACKGROUND

2.1 Traditional DNS

Originally developed by Mockapetris and others[13], the hierarchical and distributed design of DNS has served well in terms of scalability. However, it suffers from several security vulnerabilities [9]: (1) Packet Interception (2) Cache poisoning (3) Betrayal of trusted server (4) Distributed Denial of Service(DDoS). Also, the centralized administrative nature also enabled authorities to perform censorship which is often abused.

Different improvements on DNS protocol has been proposed to address some of these problems. DNSSEC[11] is a backward compatible protocol extended from the original DNS that provides authentication and integrity. These could solve problems such as cache poisoning and spoofing[8], but some problems still remain. DNSSEC does not guarantee confidentiality of the data, so DNS packets are sent in clear text that allows malicious node to eavesdrop. The structure also remains to be centralized, often making DNS a single point of failure. Moreover, The added authentication in the DNS protocol increases the size of DNS packet significantly and make it more vulnerable to DDoS attack[17].

2.2 Zooko's Triangle

According to Wilcox-O'Hearn[18], there are 3 properties that a naming system will want to achieve: human-readable, decentralization, and security. In the same article, he wrote that a naming system cannot have the three desired properties at the same time. This is widely known as the Zooko's Triangle.

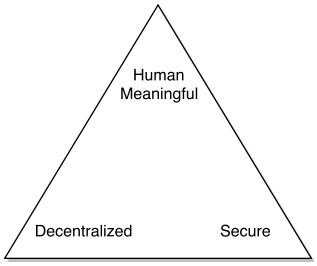

Figure 1: Zooko's Triangle

The traditional DNS only hold the human-readable property, but not the others. Where as DNSSEC achieves human-readable and security, but still remains to be centralized.

Fortunately, Namecoin implemented all three properties under the blockchain technology. This is in fact a realization of Szaba's article in 1998, Szaba[16] claimed that all three could be achieved under Byzantine Fault Tolerant protocol[12].

2.3 Blockchain

Blockchain serves as an open, decentralized and append-only ledger. It is built by cryptographic primitives and construct a byzantine tolerant protocol that allows the system to reach decentralized consensus under the presence of malicious node. This maintains a jointly decided authority that cannot be controlled by any specific node by randomly elect a node and give it the right to append blocks of records to the ledger. Other nodes will then add the block to its local blockchain after it checked the proposed block to be valid. There are several election processes: proof-of-work is used by bitcoin [14] and the most widely used method. However, there are also other election processes such as proof-of-stake[10].

The security of blockchain requires a cryptographic puzzle that could ensure the total power of malicious nodes is under certain percentage. More specifically, in the bitcoin-like systems, a cryptographic puzzle is based on calculating hash function and all nodes that competes with each other to be elected by the network by solving the cryptographic puzzle. The majority of hash power of the whole network has to act honestly to ensure security. If malicious nodes control more than half of the hash power, this implies that it is possible for them to write to the ledger arbitrarily. This is called the 51% attack.

2.4 Ethereum

Ethereum[19] is a blockchain project that constructs a Turing-complete computer over the network that records global states. This is done by requiring miner to perform actions described by programs (smart contracts) in the transactions aside from just solving the cryptographic puzzle. In order for the miner to perform the computation, the transaction sender must pay gas (fees) to the miner. If the gas is not enough, then the transaction is not mined and the gas is still paid to the miner. As a smart contract platform, Ethereum facilitates the manipulation of data on the blockchain.

2.5 Related Works

Being the first blockchain based DNS, Namecoin[1] is one of the first projects which forked from the Bitcoin[14]. Namecoin has forked with minimum modifications that allows it to store name-value data. On the other hand, BlockStack[7] constructs another application layer on top of the bitcoin blockchain and uses "colored coin" technique to store state changes on the blockchain. This separation of trust-layer blockchain and application makes it much more flexible and grants the ability to migrate between different blockchains if needed.(This has already been done before when there was a 51% security problem on the Namecoin, the authors of BlockStack migrated the application from Namecoin to Bitcoin.) There are also several other projects that are doing domain name service, such as Emercoin[2] or EtherId[3]. All of which adds additional domain aside from the existing ones and serve these additional domain in a decentralized fashion.

On the smart contract platform Ethereum, Ethereum Name Service (ENS)[5] was launched mainly to map identifiable names into Ethereum addresses. It also comes with an experimental DNS feature that hosts traditional DNS on top of the blockchain. However, the goal of the experimental DNS feature is quite different than ones described above: it is mainly used to host the Zone files of existing domains on the blockchain.

3 IMPLEMENTING DNS ON ETHEREUM

Ethereum is the second largest cryptocurrency and provides a secure and stable network for blockchain data. As opposed to namecoin, being a smaller network that has been subjected to 51% attack, Ethereum is more secure.

Ethereum has set the block time to be between 14s to 15s compared to bitcoin's 10 mins. This enables faster transactions. While the traditional DNS update takes up to 24 hours and the DNS update based on namecoin takes up to 40 mins, Ethereum transaction can typically be verified within 1 min. This reduces the downtime further compared to Namecoin.

There exist two projects in Ethereum that are related to the idea of name service. EtherID does the name translation via javascript, this means that the name is not translated outside the browser. The other is ENS, which mainly translates names into addresses. The experimental feature related to DNS is that it uploads the zone file into the blockchain and provides a gateway for the traditional DNS system to access it. This makes no changes to the client side and provide no benefits to the network.

As seen in Figure.2, there are two components in the project: (1)Smart contract that implements the database of the DNS on the block chain. (2) Local DNS Server that intercepts DNS requests with specified condition, and query the blockchain for these special request.

3.1 Environment

Ethereum Smart Contract is implemented on Solidity 0.4.15, deployed with Truffle 3.4.11, Using Geth 1.7.3 as the Ethereum Client.

Local DNS Server is developed under Java 1.8.0_131, using library web3j to communicate with Ethereum blockchain and dnsjava to process DNS request.

3.2 Smart Contract on Ethereum

Currently, The smart contract allows several actions: (1) Registering pairs of name and value: this writes the pair into a mapping of records. This also sets the message sender to be the owner of that particular name. (2) Querying a specified name: the contract will return the value stored in the blockchain. DNS Records are made public and everyone can query it. (3) Modify value: provided that the message sender is the owner of the specific name, he will be able to change the value. (4) Transfer ownership: the owner of the name is able to transfer ownership to another address.

Registration, value modification and ownership transfer perform changes on the blockchain and therefore require miners to perform the operation for us. This implies that not only the result is made public through the blockchain, but also the calculations that reached the result itself is performed by untrusted parties. On the other hand, querying does not perform any change on the blockchain and therefore is performed locally.

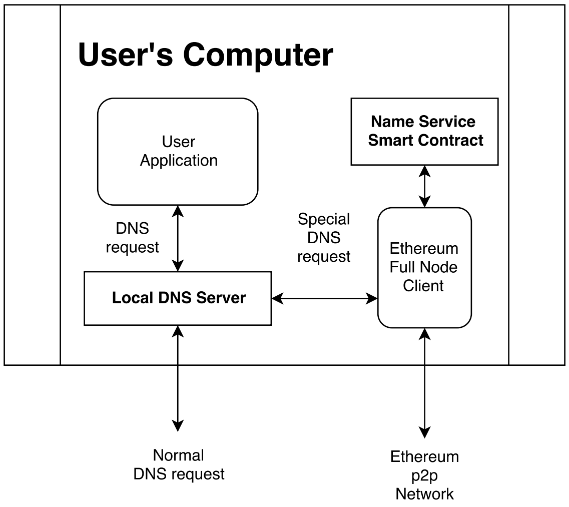

Figure 2: Components of the project

Table 1: Query Time

Domain	Non-Cached Query (ms)	Cached Query (ms)
*.meth	3.22	3.22
google.com	34.08	6.7
amazon.com	32.09	4.54
yahoo.com	49.55	4.69
netflix.com	85.66	6.22
uwaterloo.ca	7.93	4.49
nctu.edu.tw	235.37	5.27

Changes on the Ethereum blockchain is done through the Truffle console. The smart contract is deployed and tested on Ethereum Ropsten Testnet, the contract address is:

0xd5ab30a67003eec3480d28c7195df7eae91e37b6

3.3 Local DNS Server

By running a local DNS Server, we could intercept all DNS request made by the computer and process them accordingly. It checks the domain names to determine if it ended in ".myeth" and needed to query the blockchain. Currently, every domain name request that ends in ".myeth" will query the Ethereum blockchain.

This local DNS Server consists of 4 functions: (1)Receive DNS requests. (2)Interpret and filter requests (3)Query from the local blockchain data (4) Send back an answer to the DNS request.

To test the efficiency, some hosts are selected and 10,000 DNS queries are sent to each host to recorded the query time. Since DNS caching improve the speed of traditional DNS efficiency significantly, one of our tests specifically asks for the authoritative server to reply the DNS request. This would nullify the effect of DNS caching and reveal the worst case for DNS requests.

While the local DNS Server is still under development, it can successfully perform from (1) to (4). The "dig" command is a tool that will querying name service on Macintosh OS X or Linux. For testing purposes, performing "dig google.com" the local DNS Server will ignore it and let other DNS server handle the request. On the other hand, if we perform "dig test.myeth", it will notice that the domain name ends with myeth and prints the result generated from the Ethereum blockchain.

Note that the blockchain data is queried locally, therefore no DNS request is sent out to the network. To resolve DNS request locally makes it impossible for the malicious node in the network to eavesdrop or perform man-in-the-middle attack.

4 COMPARING TRADITIONAL DNS WITH BLOCKCHAIN-BASED DNS

Traditionally, the client doesn't maintain the DNS database, its DNS queries are sent to a DNS server that helps to resolve it. In the blockchain-based DNS, this depends on the kind of node that the client is running. There are full nodes that downloads the whole blockchain data and lightweight nodes such as MetaMask[6] that accesses the blockchain data through contacting a designated full node.

4.1 Improved Security and Privacy

Since Full node clients have all the blockchain data, DNS queries doesn't generate requests in the network and thus improving privacy. This also eliminates the single point failure problem since every full node must have a complete copy of the blockchain on its own. Joining the p2p network that updates the blockchain data also ensures that these data are trusted by the majority of the network, therefore avoiding cache-poisoning problem. Furthermore, since querying the blockchain is essentially accessing files on the local disk, and changing the network would require some fee, it is much harder for an attacker to perform DDoS on the blockchain network. Lastly, name registrations and accesses are not controlled by any central authorities which frees the name service from censorship.

4.2 Improved DNS query time

From the test result we could see that the query time of blockchain-based DNS is significantly shorter than the uncached traditional DNS request. This is natural since for the blockchain-based DNS, all the data are already in local storage. As for the uncached traditional DNS it will need to go through queries to reach the authoritative dns server that is potentially geometrically far. The test included "nctu.edu.tw" which is a site located in Asia so that we could see the effect of location of the authoritative dns server.

4.3 Improved DNS update time

In traditional DNS, DNS update is typically up to 24 hours. In blockchain-based DNS, the update message is collected by miners into the a block, and mined for the next round. All participants would know the information after they downloaded the latest block. In bitcoin or namecoin, blocks are being mined at an average rate of 10 minutes. In order to confirm that it is a well-established consensus, generally it would wait for 4 rounds to complete. (i.e. 40 minutes). For Ethereum, the average block time is 14 17s, therefore a DNS update could be done within 1 minute.

Table 2: Current Blockchain Info[4]

Blockchain	Total Size	Block Size Limit	Block Time
Bitcoin	171.09 GB	1 MB	8m11s
Ethereum	124.65 GB	variable (avg 18kB)	14.3s
Namecoin	5.18 GB	1 MB	9m 10s

4.4 More Network usage

However, these don’t come without a price. To maintain the database, one must join the p2p network that shares the whole blockchain data. A full node doesn’t only store the current blockchain status, instead, the full chain is downloaded. Additionally, if the network is not dedicated to name services, other data that is irrelevant to the name services must also be downloaded to maintain trust.

A simulation was performed by running a full node and joining Ethereum Ropsten test network. In 34.4 hours, 191MB of data has been downloaded and 179MB of data has been uploaded. This figure will depend on the average block size and block time of different networks. Current data of different networks are given in Table.2. As a rough estimate by using the figures of the protocol, Namecoin and Bitcoin generates around 144MB of blockchain data every 24 hours.

4.5 Notes on blockchain-based lightweight nodes

Because lightweight node clients (such as MetaMask) access the blockchain through a designated full node client, it is very close to the original DNS request and retains all of the problem in a traditional DNS. Typically the DNS requests and responses are sent as a payload of smart contract interactions, making the performance worse than traditional DNS.

5 CONCLUSION AND FUTURE WORK

A domain name service is implemented and deployed on the Ethereum Blockchain, along with a prototype of local DNS server that has the ability to resolve DNS request locally through accessing blockchain data. Advantages of the project compared to others are: (1) DNS update within 1 minute. (2) Improved privacy and security by resolving DNS request locally. However, network usage is quite large. To address the network usage problem, is it possible just to download partial data and remain secure? While currently there are proposals of lightweight node that only download data that is relevant to the node and still ensure the integrity of data, they’re still in their infancy and the security properties are still unclear.

The code of the project can be downloaded through this link: https://goo.gl/yjMKjh

REFERENCES

[1] 2010. Namecoin. (2010). https://namecoin.org/

[2] 2015. Emercoin. (2015). https://emercoin.com/

[3] 2015. EtherID. (2015). http://www.etherid.org/

[4] 2017. Bitinfochart. (2017). https://bitinfocharts.com/

[5] 2017. Ethereum Name Service. (2017). https://ens.domains/

[6] 2017. MetaMask. (2017). https://metamask.io/

[7] Muneeb Ali, Jude C Nelson, Ryan Shea, and Michael J Freedman. 2016. Blockstack: A Global Naming and Storage System Secured by Blockchains.. In *USENIX Annual Technical Conference.* 181–194.

[8] Suranjith Ariyapperuma and Chris J Mitchell. 2007. Security vulnerabilities in DNS and DNSSEC. In *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on.* IEEE, 335–342.

[9] Derek Atkins and Rob Austein. 2004. Threat analysis of the domain name system (DNS). (2004).

[10] Vitalik Buterin. 2013. What proof of stake is and why it matters. *Bitcoin Magazine, August* 26 (2013).

[11] Donald E Eastlake et al. 1999. Domain name system security extensions. (1999).

[12] Leslie Lamport, Robert Shostak, and Marshall Pease. 1982. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4, 3 (1982), 382–401.

[13] Paul V Mockapetris. 1987. Domain names-concepts and facilities. (1987).

[14] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[15] Ion Stoica, Robert Morris, David Karger, M Frans Kaashoek, and Hari Balakrishnan. 2001. Chord: A scalable peer-to-peer lookup service for internet applications. *ACM SIGCOMM Computer Communication Review* 31, 4 (2001), 149–160.

[16] Nick Szabo. 1998. Secure property titles with owner authority. *Online at http://szabo. best. vwh. net/securetitle. html* (1998).

[17] Roland van Rijswijk-Deij, Anna Sperotto, and Aiko Pras. 2014. DNSSEC and its potential for DDoS attacks: a comprehensive measurement study. In *Proceedings of the 2014 Conference on Internet Measurement Conference.* ACM, 449–460.

[18] Zooko Wilcox-OâĂŹHearn. 2003. Names: Decentralized, secure, human-meaningful: Choose two. *Retrieved May* 4 (2003), 2011.

[19] Gavin Wood. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper* 151 (2014).