

Infnote: A Decentralized Information Sharing Platform Based on Blockchain

Zhao, Cai

ID: 2025-03-10

Table of Contents

1. Introduction	2
2. Internet Censorship	3
2.1 Censorship Methods.....	3
2.1.1 DNS Manipulation or Tampering.....	3
2.1.2 IP Address Blocking	3
2.1.4 Deep Packet Inspection (DPI)	3
2.1.5 Content and Keyword Filtering	3
2.1.6 Distributed Denial of Service (DDoS).....	4
2.2 Levels of Internet Censorship.....	4
3. Blockchain	4
3.1 Advantages of Blockchain	5
3.1.1 Decentralization	5
3.1.2 Transparency.....	5
3.1.3. User-controlled Networks.....	5
3.2 Consensus Mechanisms	5
3.2.1 Proof of Work (POW)	5
3.2.2 Proof of Stake(POS)	5
3.2.3 Delegated Proof of Stake (DPOS)	6
3.2.4 Proof of Authority (POA)	6
3.2.5 Practical Byzantine Fault Tolerance (PBFT)	6
3.2.6 Delegated Byzantine Fault Tolerance (DBFT)	6
3.3 Consensus Mechanisms Comparison.....	6
4. My Work: Infnote Seeder.....	9
4.1 Problem Statement	9
4.2 System Architecture of Infnote seeder	10
4.3 Implementation of DNS server.....	11
4.4 Implementation of Crawler.....	11
4.5 Results	12
4.6 Discussion	12
References	12

1. Introduction

The Universal Declaration of Human Rights. Article 19: everyone has the right to freedom of opinion and expression; this right includes freedom to hold opinions without interference and to seek, receive and impart information and ideas through any media and regardless of frontiers. The Internet has brought with it both opportunities and challenges for freedom of opinion and expression.

On the one hand, opinion and expression can be posted or obtained on the Internet more easily. On the other hand, governments or regulators can use both technical and non-technical methods to control or suppress what can be published or viewed on the Internet. While Internet users could utilize circumvention technologies to bypass the Internet censorship to access or publish information, regulators around the world have significantly increased their efforts to control the information flow on social media [18].

The current Internet infrastructure model is heavily centralized. There are only 13 logical DNS root name servers. The IP address space is directly controlled by ICANN. These are apex players that control the Internet and are involved in delivering messages to the masses. This is being challenged by complex censorship techniques such as DNS/IP blocking and hacking attacks on content hosting websites (e.g.: blogs, social media platforms and more). Hence, there is an urgent need to solve the challenges related to this blockade by employing better circumventing approaches.

Bitcoin caught everyone's attention since it appeared as a whitepaper in 2008 [26]; various cryptocurrencies based on blockchain have emerged since then some improving bitcoin and others more innovative like Ethereum and Hyperledger. Today, the applications of blockchain and their respective Peer-to-Peer (P2P) network are designed to function much more than a decentralized currency.

Blockchain, as an append-only global ledger, has already been used in decentralized version of the DNS [24] and data storage. The append-only ledger is ideal for an information sharing platform aimed to circumvent Internet censorship, since no one would have authority to delete the content stored in the ledger. IPFS [10] and Blockstack [3] are two existing data storage platforms based on blockchain. However, IPFS currently does not support publishsubscribe pattern, and Blockstack utilizes centralized cloud servers to store data [4].

Infnote, is our answer to the world and its citizens who are looking to share information without thinking about data tampering, data loss, and anonymity. Too strong it combines all the learnings and formulas that were developed by open-source architects to provide a reliable and effective solution as part of the anti-censorship movement. The name Infnote means providing infinite power through the notes that the user publishes. Infnote will be helpful in providing a tool to content creators, social activities, journalists and others who simply want their voices to be heard.

Infnote, based on Blockchain and P2P technologies, aims at providing a platform for users to share their thoughts, insights and views through an easy-to-use medium with varying levels of Internet censorship, even in areas without Internet. Infnote is a decentralized platform that can provide the user with full anonymity (if required) and transparency, and allow this content to travel and be viewed freely across the network of users. Unlike conventional blockchain, which uses a single chain to store information, Infnote uses multiple chains, which bring an implicit reputation system among chain owners and ensure the quality of contents in Infnote.

In this report, firstly, the current situation of Internet censorship around the world is introduced in Section 2. Blockchain is discussed in Section 3. My work about Infnote Seeder is described in Section 4.

2. Internet Censorship

The Internet is supposed to provide an open platform allows anyone to share information, access opportunities and collaborate across geographical boundaries [28]. However, Internet is being challenged today by the political systems around the world. Information flow is being manipulated to show propaganda information, source of real information is either blocked or redacted, and in many cases citizens are kept unaware of the happenings outside their borders. On one hand, we have the concept of 'open', 'decentralized', 'democratized' Internet and on the other hand we also have 'The Great Firewall of China', 'Halal Internet' from Iran, 'Kwangmyong' intranet from North Korea. This brings us to the topic of censorship, which this paper hopes to address. Before presenting out solution, we first briefly review some of the censorship methods commonly deployed by various countries.

2.1 Censorship Methods

2.1.1 DNS Manipulation or Tampering

In oppressive countries, if the regulator wants to censor websites, they can employ a technique called DNS manipulation/poisoning/tampering, i.e., when a client requests for an IP address, the DNS server sends back a false IP address. This means that the client is actually visiting an incorrect website.

2.1.2 IP Address Blocking

One of the methods to block a user from visiting a website is to block its domain and IP. This completely deprives the user from visiting the website.

2.1.3 Throttling

an ISP can control the traffic and speed, which is known as bandwidth throttling. In some countries, this technique is being used during political events so that the word does not spread out to foreign sources [6][7]. From a technical perspective, throttling is achieved by slowing down TCP either by dropping packets [16] or by controlling a bandwidth provided to a specific protocol.

2.1.4 Deep Packet Inspection (DPI)

Deep packet inspection is another form of packet filtering that is being used heavily in certain countries for purposes of monitoring, blocking and sometimes throttling data flow through the Internet gateway systems. DPI filtering is used by Internet service providers to scan the payload of the Internet packets along with a normal scan of the headers to determine where to move the packet, how to classify and control it, and whether to drop it or not. This is possible in real-time with the equipment that is available today

2.1.5 Content and Keyword Filtering

Politically repressive countries pro-actively block foreign news websites, pornography, propaganda websites and content that do not match their political principles and philosophies. One easy way of censoring websites is based on their content, domain name, and specific keywords. Any website that matches specific criteria / filters, are automatically censored for violation of the government policies.

2.1.6 Distributed Denial of Service (DDoS)

This type of censorship method has been used in the past to take down several websites that stand against the regime [25]. From a technical perspective, multiple computers on the network are controlled either deliberately or unwittingly and a coordinated series of traffic is sent to a target server or cluster of servers in the cloud. The traffic could be in the form of either ICMP, UDP packets, SYN flooding or a combination of this type of traffic that will exhaust and probably turn off the computers resources of the target.

2.2 Levels of Internet Censorship

The degrees of censorship can be divided into five broad levels from a technical point of view:

Little or No Censorship. (Level 1) Little or no censorship is enforced in these countries. There is no need to use any circumvention technology since majority of the content is open to browse and access. Note: There are several countries who have placed censorship rules against illegal content like child pornography, in a genuine attempt to protect its citizens.

Selective Censorship. (Level 2) A small number of websites are blocked. Simple censorship methods, like IP address blocking or DNS filtering and redirection are likely to be used. Most democratic countries fall under this category, where websites dealing with illegal or illicit activity may be blocked and freedom of speech is well protected by law systems. Citizens can easily use any circumvention technology to bypass the censorship.

Substantial Censorship. (Level 3) A large portion of content is being blocked. Several censorship methods are implemented simultaneously. A blacklist of IP addresses and domains is likely to be enforced by the firewall, filtering Internet traffic that goes through the border Internet gateway systems. Anti-censorship circumvention tools may also be targets of censorship, making it is extremely difficult for citizens to bypass the censorship.

Pervasive Censorship. (Level 4) In this category, a whitelist is enforced by the firewall, implying that only approved Internet traffic will be allowed to pass the firewall. This makes it theoretically impossible to use any proxy or VPN to bypass the censorship because the proxy server would not be in the whitelist.

No Internet. (Level 5) In extreme situations, the Internet service may be completely cut off. Any circumvention technology that relies on the Internet will not work. It is very hard for citizens to access or distribute digital information. Currently, most citizens in North Korea cannot access the Internet. During the Arab Spring, the Egyptian government shut down the Internet in Egypt temporarily

3. Blockchain

Blockchain is a technology that is set to change how we currently conduct business in any given sector, be it health care, finance, banking, retail, logistics and so on. Blockchain is an innovative and impenetrable stand-alone framework that was popularized by Satoshi Nakamoto's work [27] in 2008. Blockchain, in conventional terms, is a public ledger that records all events, transactions and exchanges that happen between parties or nodes in the network [30]. Bitcoin popularized the concept of Blockchain, but Blockchain as a baseline platform has far greater implications than Bitcoin itself.

3.1 Advantages of Blockchain

3.1.1 Decentralization

Blockchain do not need a central data hub. Instead of running a massive data center and verifying transactions through that hub, blockchain actually allows individual transactions to have their own proof of validity and the authorization to enforce those constraints. With information on a particular blockchain piecemealed throughout the world on individual servers, it ensures that if this information fell into unwanted hands (e.g., a cyber-criminal), only a small amount of data, and not the entire network, would be compromised.

3.1.2 Transparency

One of the prime reasons blockchain is intriguing to businesses is that this technology is almost always open source. That means other users or developers have the opportunity to modify it as they see fit. But what's most important about it being open source is that it makes altering logged data within a blockchain incredibly difficult. After all, if there are countless eyes on the network, someone is probably going to see that logged data has been altered. This makes blockchain a particularly secure technology.

3.1.3. User-controlled Networks

Cryptocurrency investors are tend to be really encouraged by the control aspect of blockchain. Rather than having a third party run the show, users and developers are the ones who get to call the shots. For instance, an inability to reach an 80% consensus on an upgrade tied to bitcoin's blockchain is what necessitated a fork into two separate currencies (bitcoin and bitcoin cash).

3.2 Consensus Mechanisms

In a blockchain system, the underlying assumption is that there is no centralized node and nodes generally do not trust each other. A consensus mechanism is a fault-tolerant mechanism to achieve necessary agreement on a single state over the network. In this section, we provide an introduction to some popular consensus mechanisms.

3.2.1 Proof of Work (POW)

Proof of work is to solve mathematical puzzles and the answers can easily be verified. Bitcoin uses proof of work [27] to achieve consensus. The node that wishes to insert a block to the chain, is called a miner. The mining process is where a miner needs to scan a value that when hashed, the hash begins with enough number of zero bits. Other nodes can easily verify it by hashing a single value. After a miner produces a satisfying hash value, they have the permission to insert a block (with transactions) into the chain. The bitcoin mining process currently needs huge amount of computational resources as well as electricity to power these computers. The use of application-specific integrated circuit or ASIC can solve the mathematical puzzles much faster than CPU and GPU, in both speed and efficiency, making it is almost impossible for personal computers to join the mining process. In order to resist ASIC, many requirements will not only rely on computational power, but also other computational resources, such as memory and disk space. In POW, nodes having sufficient computational resources are more likely to solve the mathematical puzzles and therefore have a higher chance to insert blocks into the blockchain.

3.2.2 Proof of Stake(POS)

Proof of stake states that a node needs to stake an amount of its token so it has chance to insert blocks into the chain. The more tokens a node stakes, the higher the chance of inserting blocks into the chain, because it is believed that more token a user has, the less likely he would attack the

network [11]. Instead of competing using computational resources, in proof of stake, nodes compete based on the number of tokens they stake, therefore reducing the energy requirements. Similar to POW, the node with tokens (rather than computational resources) has a higher chance of inserting blocks into the chain.

3.2.3 Delegated Proof of Stake (DPOS)

Unlike proof of stake, in which every node has chance to insert blocks to the chain, Delegated Proof of Stake only allows delegated nodes to insert blocks. Delegated nodes are chosen by voting processes. Votes are weighted according to the number of tokens each voter stakes. The first tier of nodes (usually less than 100 nodes) who receive most of the votes will earn the right to insert blocks into the chain. Voters can acquire the tokens through Initial Coin Offering (ICO) or trade platforms. The price of the tokens is determined by the market. The value of these tokens comes with the power to vote for delegated nodes and deploy smart contracts on the chain. The more tokens a voter has in their possession, the more power they have to elect the delegated nodes and therefore have more indirect control over the chain. To attract votes, potential delegated nodes need to demonstrate their abilities and identities.

3.2.4 Proof of Authority (POA)

In Proof of Authority (POA) network, only approved nodes can validate blocks and insert them into the chain. Unlike delegated nodes in DPOS mechanism, approved nodes are not chosen by voting. Currently, POA is mainly used in private network, where every node knows each other and therefore trust approved nodes to maintain the chain. However, approved nodes have to maintain their computer uncompromised state given the power vested in them. Approved nodes need to gain reputation through their work on the network. However, any negative activity recorded, can destroy the reputation of the approved node as well.

3.2.5 Practical Byzantine Fault Tolerance (PBFT)

Numerous protocols have been proposed to solve the problem of Byzantine Fault Tolerance (BFT) [23]. Practical Byzantine Fault Tolerance (PBFT) is one of its solutions, which can handle up to $\frac{1}{3}$ of the malicious nodes. A block will be generated in a round. Each round can be divided into three phases: pre-prepared, prepared and commit. Each node has to receive $\frac{2}{3}$ nodes from other nodes in order to enter the next phase [12]. Therefore, PBFT requires every node to be known to the network.

3.2.6 Delegated Byzantine Fault Tolerance (DBFT)

Delegated Byzantine Fault Tolerance (DBFT) is another solution for the BFT problem. The whole process is similar to PBFT, except only a small number of delegated nodes are voted to insert blocks into the chain.

3.3 Consensus Mechanisms Comparison

Different consensus mechanisms have different advantages and disadvantages. Table 1 gives a comparison between them basing on the example listed, including some properties given by [42].

Criteria	POW	POS	DPOS	POA	PBFT	DBFT
Example	Bitcoin [36]	Nxt [37]	EOS [38]	POA.Network [39]	Hyperledger Fabric [40]	Neo [41]
Node identity management [42]	open, entirely decentralized	open, entirely decentralized	open, entirely decentralized	permissioned, nodes need to know IDs of all other nodes	permissioned, nodes need to know IDs of all other nodes	permissioned, nodes need to know IDs of all other nodes
Consensus finality [42]	No	Yes	Yes	Yes	Yes	Yes
Scalability [42]	excellent (thousands of clients)	excellent (thousands of clients)	excellent (thousands of clients)	excellent (thousands of clients)	excellent (thousands of clients)	excellent (thousands of clients)
Performance (throughput) [42]	limited (due to possible of chain forks) [42]	Good (more than nine times Bitcoin's current peak value [43])	Good (better than Bitcoin) [38]	Good (better than Bitcoin) [39]	excellent (tens of thousands transactions per second) [42]	excellent (tens of thousands transactions per second) [42]
Energy Consumption [42]	high	low	low	low	low	low
Tolerated power of an adversary [42]	$\leq 25\%$ computing power [42]	$< 51\%$ stake [46]	$< 51\%$ validators [46]	Ensure that validators do not want to attack the network	$\leq 33\%$ voting power [42]	$\leq 33\%$ voting power [42]
Forking [47]	When two nodes find suitable nonce at the same time	very difficult	very difficult	very difficult	very difficult	very difficult
Block creating speed [47]	low, depends on variant	Fast [37]	Fast [38]	Fast [39]	Fast [40]	Fast [41]
Type[47]	Proof-Based Consensus	Voting Based Consensus	Voting Based Consensus	Voting Based Consensus)	Voting Based Consensus	Voting Based Consensus
Decentralization [47]	High	Low	Low	Low	Low	Low

Table 1: Typical Consensus Mechanisms Comparison

1. *Example.* Bitcoin is based on Proof of Work (POW) [36] while Nxt is a Cryptocurrency with Proof-of-Stake (POS) forging [37]. Further, EOS is an open source smart contract platform utilizing Delegated Proof of Stake (DPOS) [38]. POA.Network, a sidechain to Ethereum, adopts Proof of Authority (POA) as its consensus mechanism [39]. Hyperledger Fabric utilizes Practical Byzantine Fault Tolerance (PBFT) to reach consensus [40]. NEO implements DBFT to guarantee fault tolerance [41].

2. *Node identity management.* How node identities are managed in consensus mechanisms is possibly their most fundamental difference [42]. In Bitcoin (POW), Nex (POS) and EOS (DPOS), everyone can participate in the network without being permissioned. In contrast, in POA.Network (POA), Hyperledger Fabric (PBFT) and Neo (DBFT), require blockchain participants identity to be known.

3. *Consensus finality.* Roughly speaking, what is often informally referred to as “consensus finality” is a property that mandates that a valid block, appended to the blockchain at some point in time, be never removed from the blockchain [42]. Consensus finality is not satisfied by POW-based blockchains [42]. In contrast, consensus finality is satisfied by POS, DPOS, POA, PBFT and DBFT.

4. *Scalability.* Although decoupling the issue of blockchain scalability (with the number of nodes and clients in the system) from that of blockchain performance (latency and throughput) is not entirely possible, we nevertheless first focus on the number of nodes and clients for which POW and BFT technologies have been proven to work in practice [42]. when it comes to scalability with the number of clients, both POW and BFT protocols (including PBFT and TBFT) support thousands of clients and scale well [42]. Also, POS, DPOS and POA can support thousands of clients and scale well.

5. *Performance.* Beyond the very limited performance of Bitcoin of up to 7 transactions per second (with the current block size) and 1-hour latency with 6-block confirmation, POW-based blockchains face inherent performance challenges [42]. In contrast, modern BFT protocols (PBFT and DBFT) have been confirmed to sustain tens of thousands of transactions with practically network-speed latencies [42]. In terms of POS, DPOS and POA adopted by the example in table 1, their performance is better than POS which is utilized by Bitcoin according to their official document.

6. *Energy Consumption.* Energy here refer to electricity, processing power etc. Among all the consensus mechanisms, only POW needs huge amount of energy. Estimated annual electricity consumption for entire Bitcoin network currently is 73.12 TWh, about 30% annual consumption of Australia, as Oct 2018 [49].

7. *Adversary.* POW and BFT (PBFT, DBFT) consider different adversaries [42]. In POW blockchain like Bitcoin, what matters is the total computational (hashing) power controlled by the adversary. Initially, Bitcoin was thought to be invulnerable so long as the adversary controls less than 50% of hashing power. Years later, it was shown that Bitcoin mining is actually vulnerable even if only 25% of the computing power is controlled by an adversary [44]. In POS and DPOS, an attacker would need to acquire 50%+ of the stake or the validators within that system. However, POA, as envisioned by the POA Network, comes with a security approach that is more human: validators are

people who have a name, a family, and a home. In their home one could find hardware and potentially private keys important to the network. In fact, validators of the POA Network have their name and address disclosed to the public in a registry maintained by the state where they live, and guaranteed to be up to date within 30 days from a change in their record [39]. So the point is to ensure the validators do not want to attack the network. In contrast, BFT voting schemes are known to tolerate at most $n/3$ corrupted nodes [45].

8. *Forking* [47]. In POW, forking can happen if two miners find a suitable nonce at the same time. Meanwhile with POS, it is very difficult, happening only when a miner can own up to 51% of all stake in the whole verifying network [50]. In DPOS, POA, PBFT and DBFT, normally it is very difficult to make a fork.

9. *Block creating speed* [47]. Considering the speed of creating a new block in the chain, POW require much time to append a new block to the chain [47], while with POS, DPOS, POA, PBFT and DBFT, the block creating speed is faster, because no node has to solve any puzzle.

10. *Type* [47]. Consensus Mechanisms can be divided into two main subsections: proof-based consensus Mechanism and vote-based consensus Mechanism [47]. POW is a typical proof-based consensus Mechanism, while to some degree, POS, DPOS, POA, PBFT and DBFT are vote-based consensus Mechanisms.

11. *Decentralization* [47]. Vote-based consensus is often conducted in private and consortium blockchain, in which the decentralization degree is lower than in public blockchain with proof-based consensus [47]. Consequently, in POW the decentralization degree is higher than POS, DPOS, POA, PBFT and DBFT.

12. *Trust* [47]. Since in POW the decentralization degree is higher than POS, DPOS, POA, PBFT and DBFT, the trust in POW would be higher than POS, DPOS, POA, PBFT and DBFT.

4. My Work: Infnote Seeder

4.1 Problem Statement

Infnote stores information based on blockchain technology. And it supports a multi-chain structure, which means there are several independent parallel chains. Each chain is controlled by its chain owner and every one can become a chain owner simply created by creating a new chain. Infnote uses POA as its consensus mechanism. There are two kinds of nodes, full node and light node. Same as bitcoin, full nodes are for devices that have sufficient bandwidth and computational resources to support all the functions of Infnote. Functions include: store all data in blockchain, provide logic to view and publish content and act as a server by listening for connections and providing services to clients. Light node are devices, such as smart-phones or web browser, which has limited resources and processing power. Also, Infnote has its own P2P network, namely Infnote P2P network. Peer discovery in Infnote P2P network is extremely crucial.

As to Bitcoin P2P network, when started for the first time, programs don't know the IP addresses of any active full nodes. In order to discover some IP addresses, they query one or more DNS names (called DNS seeds) hardcoded into Bitcoin Core and BitcoinJ. The response to the lookup should

include one or more DNS A records with the IP addresses of full nodes that may accept new incoming connections.

The DNS seeds are maintained by Bitcoin community members: some of them provide dynamic DNS seed servers which automatically get IP addresses of active nodes by scanning the network; others provide static DNS seeds that are updated manually and are more likely to provide IP addresses for inactive nodes. In either case, nodes are added to the DNS seed if they run on the default Bitcoin ports of 8333 for mainnet or 18333 for testnet.

Similarly, in Infnote P2P network, a DNS seeder is needed as well. However, as Infnote P2P protocol is quite different from Bitcoin's P2P protocol, it is impossible to reuse Bitcoin's DNS seeder, although it is open-source. And building a DNS seeder for Infnote, namely Infnote seeder is important and meaningful.

4.2 System Architecture of Infnote seeder

Infnote seeder is an important part of Infnote, especially when there are lots of full nodes in Infnote P2P network. Figure 1 shows the system architecture of Infnote seeder. Firstly, the crawler sends messages to the full node in Infnote P2P network; if the crawler gets response message, including chain information or peers information, from a full node, this full node is labelled as a good node; if no messages are received by the crawler, this full node is labelled as a not good full node. When all the nodes in P2P network are crawled or visited, the crawler will update the infnote DNS database and generate a nodes report, which tells us which full node is good, and which is not good. When the crawler crawls Infnote P2P network, it can send want peers message to the full node to get other full nodes' ip it records. By this way, the crawler can discover all the good full node in Infnote P2P network. When the build-in DNS server is started, it can receive request messages from a new node, which wants to join to the Infnote P2P network; and then it response a message to the new node, which includes the IPs of good full nodes in Infnote P2P network. And then this new node can connect to Infnote P2P network. Infnote seeder is open-source and free. The whole source code can be obtained from the Github, and the link is https://github.com/Infnote/infnote_seeder.

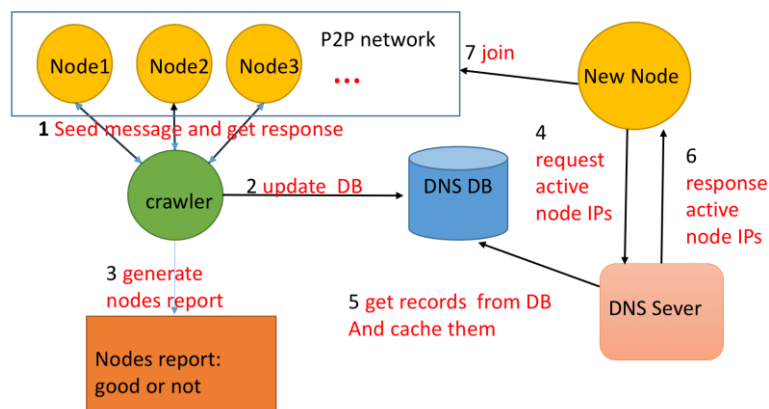


Figure 1: System architecture of Infnote seeder

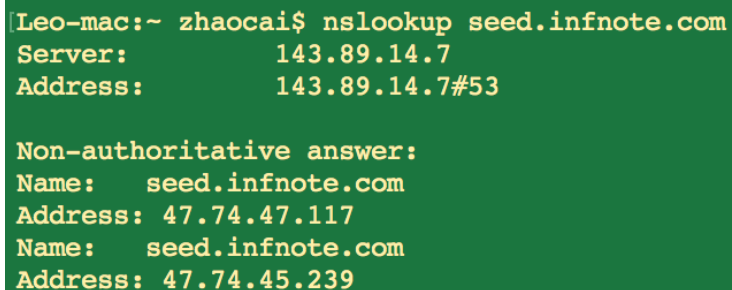
4.3 Implementation of DNS server

Although there are a lot of powerful DNS server software, they are not suitable to our special application. We just want a light DNS server, which does not cost too much memory or resources. As python language is used in the development of Infnote P2P protocol, it is used to implement a light DNS server for Infnote, namely Infnote DNS. There are several useful packages such as `gevent`, `dnslib` and `pylru` used to build Infnote DNS. `Gevent` is used to start coroutines. `Gevent.queue` is used as request message queue. `Dnslib` is used to encode or decode DNS packets. `Pylru` is used as LRU cache

The main steps of Infnote DNS are as follows:

1. start udp server
2. accept requests, and store in `deq_cache`
3. get data from `deq_cache`
4. decode DNS packets from data
5. query in `dns_cache` if dns db is not updated,
else query in dns db, and return and cache the result in `dns_cache`

For example, when we use the command: `nslookup seed.infnote.com`; Infnote DNS will receive the request message stored in `deq_cache`, which is a DNS packet; then this DNS packet will be decode and Infnote DNS knows the client requests the IPs of 'seed.infnote.com', then Infnote DNS query in `dns_cache` if dns db is not updated if find records, return the result, else query in dns db, and then return and cache the result in `dns_cache`. More Figure 2 shows the result of running the command of `nslookup seed.infnote.com`. The detailed implementation of Infnote DNS can be seen in the source code file, `infnote_dns.py`.

A terminal window with a green background and yellow text. The prompt is 'Leo-mac:~ zhaocai\$'. The command 'nslookup seed.infnote.com' has been executed. The output shows the server IP as 143.89.14.7 and the address as 143.89.14.7#53. Below this, it shows a 'Non-authoritative answer:' with two entries: 'Name: seed.infnote.com Address: 47.74.47.117' and 'Name: seed.infnote.com Address: 47.74.45.239'.

```
Leo-mac:~ zhaocai$ nslookup seed.infnote.com
Server:          143.89.14.7
Address:         143.89.14.7#53

Non-authoritative answer:
Name:   seed.infnote.com
Address: 47.74.47.117
Name:   seed.infnote.com
Address: 47.74.45.239
```

Figure 2: the result of `nslookup seed.infnote.com`.

4.4 Implementation of Crawler

Given a IP address of a full node, the crawler uses this IP address to visit the full node and get other full nodes' IP by this known full node. By this way, the crawler can visit all the full node in current Infnote P2P network. And then it generates a report of the full nodes' availability and updates the DNS server database, which only records the good nodes' IP address.

The crawler crawls Infnote P2P network every 60 seconds as default. Of course, the interval of crawling can be set as other time interval, such as 300 seconds by a command. The detailed implementation of Infnote crawler can be seen in the source code files, `crawler.py` and `run_crawler_regularly.py`.

Figure 3 shows the part running log of crawler in Infnote P2P network.

```

2018-12-11 Tuesday 16:19:38 crawler.py[line:105] INFO request_peers from 47.74.45.239
2018-12-11 Tuesday 16:19:38 crawler.py[line:118] INFO {"identifier": "0", "type": "answer", "content": {"type": "peers", "peers": [{"address": "47.74.47.117", "port": 32767}]}}
2018-12-11 Tuesday 16:19:38 crawler.py[line:137] INFO 47.74.45.239 is good
2018-12-11 Tuesday 16:19:38 crawler.py[line:105] INFO request_peers from 47.74.47.117
2018-12-11 Tuesday 16:19:38 crawler.py[line:118] INFO {"identifier": "0", "type": "answer", "content": {"type": "peers", "peers": [{"address": "1.2.3.4", "port": 32767}]}}
2018-12-11 Tuesday 16:19:38 crawler.py[line:137] INFO 47.74.47.117 is good
2018-12-11 Tuesday 16:19:38 crawler.py[line:105] INFO request_peers from 1.2.3.4
2018-12-11 Tuesday 16:21:49 crawler.py[line:127] INFO 1.2.3.4 is not available
2018-12-11 Tuesday 16:21:49 crawler.py[line:70] INFO request_info from 1.2.3.4
2018-12-11 Tuesday 16:24:00 crawler.py[line:79] INFO 1.2.3.4 is not good
2018-12-11 Tuesday 16:24:00 crawler.py[line:164] INFO crawled ips dict keys(['47.74.45.239', '47.74.47.117', '1.2.3.4'])
2018-12-11 Tuesday 16:25:00 crawler.py[line:105] INFO request_peers from 47.74.45.239
2018-12-11 Tuesday 16:25:00 crawler.py[line:118] INFO {"identifier": "0", "type": "answer", "content": {"type": "peers", "peers": [{"address": "47.74.47.117", "port": 32767}]}}
2018-12-11 Tuesday 16:25:00 crawler.py[line:137] INFO 47.74.45.239 is good
2018-12-11 Tuesday 16:25:00 crawler.py[line:105] INFO request_peers from 47.74.47.117
2018-12-11 Tuesday 16:25:00 crawler.py[line:118] INFO {"identifier": "0", "type": "answer", "content": {"type": "peers", "peers": [{"address": "1.2.3.4", "port": 32767}]}}
2018-12-11 Tuesday 16:25:00 crawler.py[line:137] INFO 47.74.47.117 is good
2018-12-11 Tuesday 16:25:00 crawler.py[line:105] INFO request_peers from 1.2.3.4
2018-12-11 Tuesday 16:27:10 crawler.py[line:127] INFO 1.2.3.4 is not available
2018-12-11 Tuesday 16:27:10 crawler.py[line:70] INFO request_info from 1.2.3.4
2018-12-11 Tuesday 16:29:21 crawler.py[line:79] INFO 1.2.3.4 is not good
2018-12-11 Tuesday 16:29:21 crawler.py[line:164] INFO crawled ips dict keys(['47.74.45.239', '47.74.47.117', '1.2.3.4'])

```

Figure 3: running log of crawler

4.5 Results

Figure 4 shows the results of using the dig command to find good full nodes in Infnote P2P network. The 'ANSWER SECTION' tells us the available full nodes' IP addresses, which are 47.74.47.117 and 47.74.45.239. This means there only two good full nodes in Infnote P2P network so far. And this IP address list are maintained by the crawler in real time. From the 'ADDITIONAL SECTION', we can know the infnote DNS sever's IP address is 47.74.45.239, which is also a full node in Infnote P2P network. Of course, Infnote DNS sever can run independently.

```

Leo-mac:~ zhaocai$ dig seed.infnote.com

; <<>> DiG 9.8.3-P1 <<>> seed.infnote.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 57544
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;seed.infnote.com.      IN      A

;; ANSWER SECTION:
seed.infnote.com.      0       IN      A      47.74.47.117
seed.infnote.com.      0       IN      A      47.74.45.239

;; AUTHORITY SECTION:
seed.infnote.com.      1799    IN      NS      ns.infnote.com.

;; ADDITIONAL SECTION:
ns.infnote.com.        1799    IN      A      47.74.45.239

;; Query time: 399 msec
;; SERVER: 143.89.14.7#53(143.89.14.7)
;; WHEN: Thu Dec 6 10:17:34 2018
;; MSG SIZE  rcvd: 99

Leo-mac:~ zhaocai$

```

Figure 4: the results of using the dig command

4.6 Discussion

Infnote seeder is a crawler for the Infnote network, which exposes a list of reliable nodes via a built-in DNS server. It regularly revisits known nodes to check their availability and bans nodes after enough failures, or bad behaviour. However, Infnote DNS seeds still poses a small amount of risk for the network. As such, Infnote DNS seeds must be run by entities which have some minimum level of trust within the Infnote community.

References

- [1] 2003. I2P: The Invisible Internet Project. <https://geti2p.net/>
- [2] Faten Adel Alabdulwahhab. 2018. Web 3.0: The Decentralized Web Blockchain

- networks and Protocol Innovation. In 2018 1st International Conference on Computer Applications & Information Security (ICCAIS). IEEE, 1–4.
- [3] 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.
- [4] Muneeb Ali, Ryan Shea, Jude Nelson, and Michael J Freedman. 2017. Blockstack: A new decentralized internet. Whitepaper, May (2017).
- [5] Muhammad Salek Ali, Koustabh Dolui, and Fabio Antonelli. 2017. IoT data privacy via blockchains and IPFS. In Proceedings of the Seventh International Conference on the Internet of Things. ACM, 14.
- [6] Collin Anderson. 2013. Dimming the Internet: Detecting throttling as a mechanism of censorship in Iran. arXiv preprint arXiv:1306.4361 (2013).
- [7] Simurgh Aryan, Homa Aryan, and J Alex Halderman. 2013. Internet Censorship in Iran: A First Look.. In FOCL.
- [8] UN General Assembly. 1948. Universal declaration of human rights. UN General Assembly (1948).
- [9] Leo Bach, Branko Mihaljević, and Martin Žagar. 2018. Comparative Analysis of Blockchain Consensus Algorithms. In 41st International Convention for Information and Communication Technology, Electronics and Microelectronics (MIPRO 2018).
- [10] Juan Benet. 2014. IPFS-content addressed, versioned, P2P file system. arXiv preprint arXiv:1407.3561 (2014).
- [11] Iddo Bentov, Ariel Gabizon, and Alex Mizrahi. 2016. Cryptocurrencies without proof of work. In International Conference on Financial Cryptography and Data Security. Springer, 142–157.
- [12] Miguel Castro, Barbara Liskov, et al. 1999. Practical Byzantine fault tolerance. In OSDI, Vol. 99. 173–186.
- [13] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W Hong. 2001. Freenet: A distributed anonymous information storage and retrieval system. In Designing privacy enhancing technologies. Springer, 46–66.
- [14] Bram Cohen. 2008. The BitTorrent protocol specification.
- [15] Lucas Dixon, Thomas Ristenpart, and Thomas Shrimpton. 2016. Network traffic obfuscation and automated internet censorship. arXiv preprint arXiv:1605.04044 (2016)
- [16] David Fifield. 2017. Threat modeling and circumvention of Internet censorship. Ph.D. Dissertation. UC Berkeley.
- [17] Ragib Hasan, Zahid Anwar, William Yurcik, Larry Brumbaugh, and Roy Campbell. 2005. A survey of peer-to-peer storage techniques for distributed file systems. In Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on, Vol. 2. IEEE, 205–213.
- [18] Freedom House. 2017. Freedom on the Net 2017. (2017).
- [19] Don Johnson, Alfred Menezes, and Scott Vanstone. 2001. The elliptic curve digital signature algorithm (ECDSA). International journal of information security 1, 1 (2001), 36–63.
- [20] Channu Kambalyal. 2010. 3-tier architecture. Retrieved On 2 (2010).

- [21] Tamas Kocsis. 2015. ZeroNet Project. <https://zeronet.io>
- [22] 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.
- [23] Iuon-Chang Lin and Tzu-Chun Liao. 2017. A Survey of Blockchain Security Issues and Challenges. *IJ Network Security* 19, 5 (2017), 653–659.
- [24] Andreas Loibl and J Naab. 2014. Namecoin. namecoin.info (2014).
- [25] Bill Marczak, Nicholas Weaver, Jakub Dalek, Roya Ensafi, David Fifield, Sarah McKune, Arn Rey, John Scott-Railton, Ron Deibert, and Vern Paxson. 2015. An analysis of china’s “Great Cannon”. *FOCI. USENIX* (2015), 37.
- [26] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).
- [27] FIPS PUB. 2012. Secure hash standard (shs). *FIPS PUB 180 4* (2012).
- [28] O Solon. 2017. Tim Berners-Lee on the future of the web: “The system is failing”. *The Guardian* (2017).
- [29] Alexandru Stanciu. 2017. Blockchain based distributed control system for edge computing. In *Control Systems and Computer Science (CSCS), 2017 21st International Conference on*. IEEE, 667–671.
- [30] Paul Syverson, R Dingledine, and N Mathewson. 2004. Tor: The second generation onion router. In *Usenix Security*.
- [31] Inc The Tor Project. 2002. Tor Project. <https://www.torproject.org>
- [32] Alex the Vries. 2014. Digiconomist. digiconomist.net
- [33] David Vargas, Robert Tran, and Omar Gonzalez. [n. d.]. Censorship-Resistant File Storage. ([n. d.]).
- [34] Marko Vukolić. 2015. The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *International Workshop on Open Problems in Network Security*. Springer, 112–125.
- [35] Zhensheng Zhang, Ya-Qin Zhang, Xiaowen Chu, and Bo Li. 2004. An overview of virtual private network (VPN): IP VPN and optical VPN. *Photonic network communications* 7, 3 (2004), 213–225.
- [36] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).
- [37] <https://nxtplatform.org/get-started/developers/>
- [38] <https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md#consensus-algorithm-bft-dpos>
- [39] <https://poa.network/>
- [40] <https://github.com/hyperledger-archives/fabric/wiki/Consensus>
- [41] <http://docs.neo.org/en-us/basic/consensus/consensus.html>
- [42] M. Vukolić, “The quest for scalable blockchain fabric: Proof-of-work vs. bft replication,” in *International Workshop on Open Problems in Network Security, Zurich, Switzerland, 2015*, pp. 112–125.
- [43] https://www.dropbox.com/s/cbuwrorf672c0yy/NxtWhitepaper_v122_rev4.pdf
- [44] Ittay Eyal and Emin Gü n Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *Financial Cryptography and Data Security - 18th International Conference, FC 2014*, pages 436–454, 2014.
- [45] Cynthia Dwork, Nancy Lynch, and Larry Stockmeyer. Consensus in the presence of partial synchrony. *J. ACM*, 35, April 1988.

- [46] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang³. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends 2017 IEEE 6th International Congress on Big Data
- [47] Giang-Truong Nguyen* and Kyungbaek Kim A Survey about Consensus Algorithms Used in Blockchain J Inf Process Syst, Vol.14, No.1, pp.101~128, February 2018
- [48] <https://medium.com/poa-network/proof-of-authority-consensus-model-with-identity-at-stake-d5bd15463256>
- [49] <https://digiconomist.net/bitcoin-energy-consumption>
- [50] S. King and S. Nadal, "PPcoin: peer-to-peer crypto-currency with proof-of-stake," 2012 [Online]. Available: <https://decred.org/research/king2012.pdf>.