Ombuds: A Public Space With A Single Shared History

Alex Kuck
University of Virginia
ask5rq@virginia.edu

Nick Skelsey
Soapbox Systems
nskelsey@gmail.com

May 11, 2015

Abstract

Recent approaches to resisting censorship on the Internet have focused on restoring access to services that have been blocked. These services, including social media and web forums, provide platforms for activists to spread information to the general public, but the services themselves are subject to manipulation and control. Instead of taking this approach, we have created a resilient platform to share and disseminate information over the Internet without a central authority. Our application, Ombuds, does this by distributing public statements through a peer-to-peer network and storing those statements in a single shared history.

Our tools leverage Bitcoin's network and Bitcoin's block chain. With our software, a connection to the Bitcoin network, and a small amount of bitcoin, activists can store controversial public statements in a highly reliable and permanent place. This paper describes the design of the system, potential attacks against it, and approaches to scaling the system in terms of cost and resource usage.

1 Introduction

Internet censorship is an effective tool to mute political action and stifle organizations that could grow support and raise awareness for their cause. Two categorical approaches to Internet censorship have emerged in recent times: a censor can degrade and prevent access to content through technological means, and a censor can compel services to remove or change content through legal or social pressure.

An arsenal of passive and active technical methods of Internet censorship can be employed against web services hosting unfavorable content. A censor with strong political influence over Internet Service Providers (ISPs) within its borders can monitor Internet communication, poison DNS records, black hole TCP packets, and filter targeted traffic [1]. Examples include the Great Firewall of China and the Turkish BTK firewall [2, 3]. Aggressive censors with computing resources and technical capabilities can compromise or shutdown web services with active attacks, such as distributed denial-of-service attacks, man-in-the-middle attacks and exploitation of computing infrastructure.

Attempts to circumvent passive blocks and filtering have proven ineffective against the Great Firewall of China. Numerous tools that seek to restore access to these censored sites have been constructed, but they are not effective at larger political scales [4]. Tor and related tools that disguise connections into Tor's network can be unmasked and blocked through aggressive probing by a censor [5]. Telex and similar tools that proxy traffic through ISPs committed to fighting censorship requires extra national organization of disparate and often dysfunctional parties [6]. Because multinational companies require access to encrypted one-to-one communication, virtual private networks (VPNs) have remained unblocked [7]. However, encrypted one-to-one communication is only a small part of a broader fight against censorship.

The operators of web services are also susceptible to legal, social, and financial pressures leveraged by censors. With political control censors can require companies to register users with real identities, remove or modify user content, and keep logs of all traffic on the cite[4, 8]. When legal pressure

manifests itself, site operators can find themselves caught between protecting themselves and protecting the information and the rights of their users [9]. Relying on a few individuals or a single web company to consistently resist these pressures is unacceptable, when systems that decentralize trust can be built.

This paper assumes that preserving access to the whole Web, while a legitimate anti-censorship goal, is unnecessary. Instead, this paper argues that digital activists only require a space where dialogue and organization can freely occur between mutually respected participants. This is akin to Jürgen Habermas's conception of the public sphere [10], except instead of meeting in physical space participants communicate digitally. Social media previously provided this digital space, but with censoring capabilities and the moderation on these services increasing a system with stronger protections for speech must be built.

2 A Single Shared History

In order to effectively protect public speech, it is not sufficient to simply provide a platform to broadcast messages. Any user who was not listening at the time of broadcast or any new user will miss statements. As statements are created they must be placed in a single history, so that they can be retrieved asynchronously. This way if authors can get their statements into the history they can be certain that those statements will not be censored and readers can be certain that the statements in the history have not been altered or omitted.

Bitcoin is a peer-to-peer electronic cash system. The design defines a public ledger of transaction history referred to as the block chain. Any full participant[1] in the Bitcoin network can use the block chain to ensure that the rules of the entire cash system are maintained, while a single user with few resources can verify that they have received or spent funds [11].

Ombuds uses Bitcoin's block chain in two analogous ways. A full participant can verify that the set of all public statements stored in the system are available and unmodified, while a user with limited resources can verify that any statement they have submitted has been included in the single shared history. To do this, a single shared history is constructed from public statements stored within Bitcoin's block chain. While the block chain defines a single history for all transactions stored within it, the public record is a subset of those transactions that are formatted as "bulletins" which are placed in "boards."

Despite Bitcoin's initial design as a digital cash system, we wish to demonstrate that using the Bitcoin network as a bulletin board system is possible and economically feasible for the medium term (within two to four years with 100,000 users). Using Bitcoin to store data imposes negative externalities rest of the network when a transaction is distributed and stored in the chain. These externalities are minimal for any single transaction, but in aggregate the costs, mainly storing and distributing transactions in blocks, represent close to the total cost of using Bitcoin.

Thus, schemes that seek to use Bitcoin transactions for unforeseen purposes must in some way justify their imposition on their peers. We propose that the public good of providing a distributed and resilient platform for free speech outweighs this cost, but we acknowledge that this project is only possible because peers on the Bitcoin network are willing to bear these costs. In section 6, we discuss ways to reduce the negative externalities that Ombuds imposes on Bitcoin.

3 Storing Bulletins in a Block Chain

Public statements, referred to as bulletins, are small text based messages that are wrapped in Bitcoin transactions. The way bulletins are contained within Bitcoin transactions is similar to the way HTTP frames are included in TCP/IP packets. This is illustrated in Figure 1.

[1] A full participant is a participant in the peer-to-peer network with the resources to maintain a complete and up to date copy of Bitcoin's block chain.

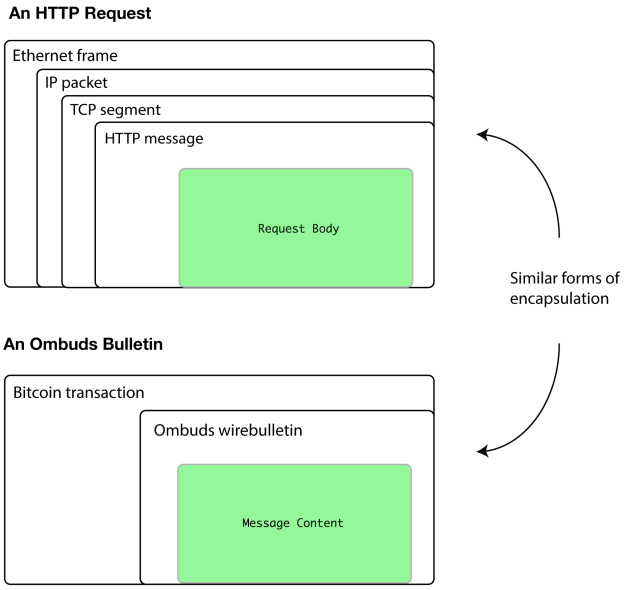

Figure 1: **The OSI model defines the way communications systems are partitioned into functional layers. Wrapping application data in Bitcoin transactions is analogous.**

Several properties of the containing Bitcoin transaction are used to provide additional information about the bulletin within: a bulletin is uniquely identifiable by the hash of its transaction, its author is attributed to the first public key that funds the transaction, and the depth of the block that contains the transaction is the depth of the bulletin. The schema of a bulletin along with all of the attributes that are used from a transaction are illustrated in Figure 2 and further documented in appendix A.

To create a valid bulletin, a user of the system must have bitcoin to spend on both paying a mining fee and "burning"[2] a small amount of bitcoin to store their message. Although the cost of storing a bulletin in the block chain is relatively low now, it is dependent on the market value of bitcoin, the transaction processing fee, the size of the message, and the amount of bitcoin the transaction must burn.

$$Cost = \left(BurnAmount * \left\lceil \frac{|Message|}{|PubKeyHash|} \right\rceil + TxFee \right) * \frac{USD}{BTC} \tag{1}$$

This linear relationship between the cost of storing bulletins and a bitcoin's value means that this use of a block chain is only possible while the market value for the underlying currency is low. Considering a price of 250 USD for one bitcoin, storing a tweet-sized bulletin costs around $(0.00000547 * 140/20 + 0.0005) * 250 = 0.15$ USD.

Once the bulletin is submitted to the network it undergoes the same processing that a regular Bitcoin transaction does. A modified full Bitcoin node can monitor the network and build the public record from bulletins it sees relayed and mined in the Bitcoin network. If the fee paid was not high enough to incentivize miners to process the transaction, the bulletin will not be included in a block and therefore left out of the public record. If the bulletin is left out of the public record, that statement will not be seen by new participants to the network. The feasibility of attacks based on the exclusion

[2] "Burning" refers to the practice of sending bitcoin (or tiny amounts of bitcoin) to addresses that have no corresponding private key and are thus unspendable. In our case, we are burning the minimum transferable amount of bitcoin to store data. The current dust amount is 547 Satoshi which is roughly 0.0015 USD.

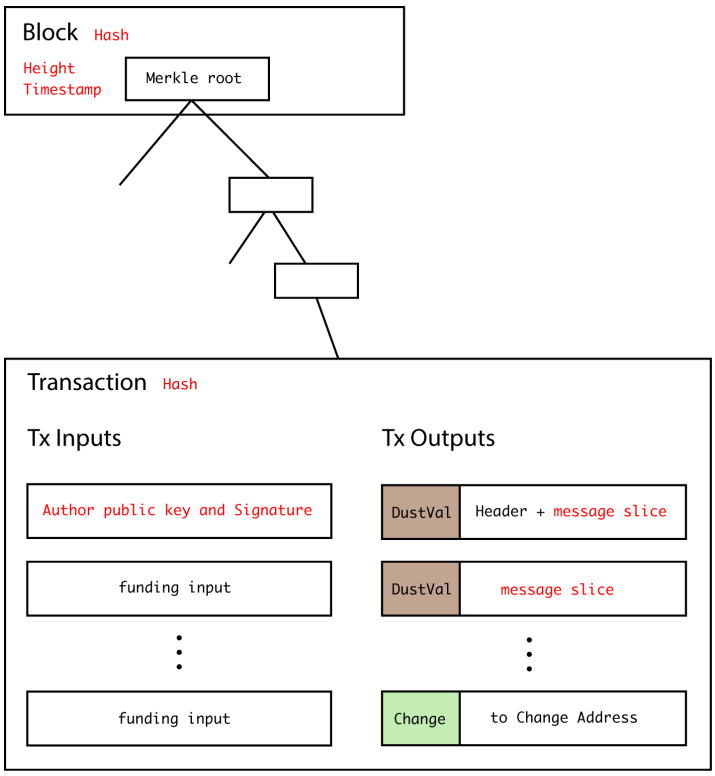

Figure 2: **An encoded bulletin** with aspects of the transaction and block that are stored in the public record highlighted in red.

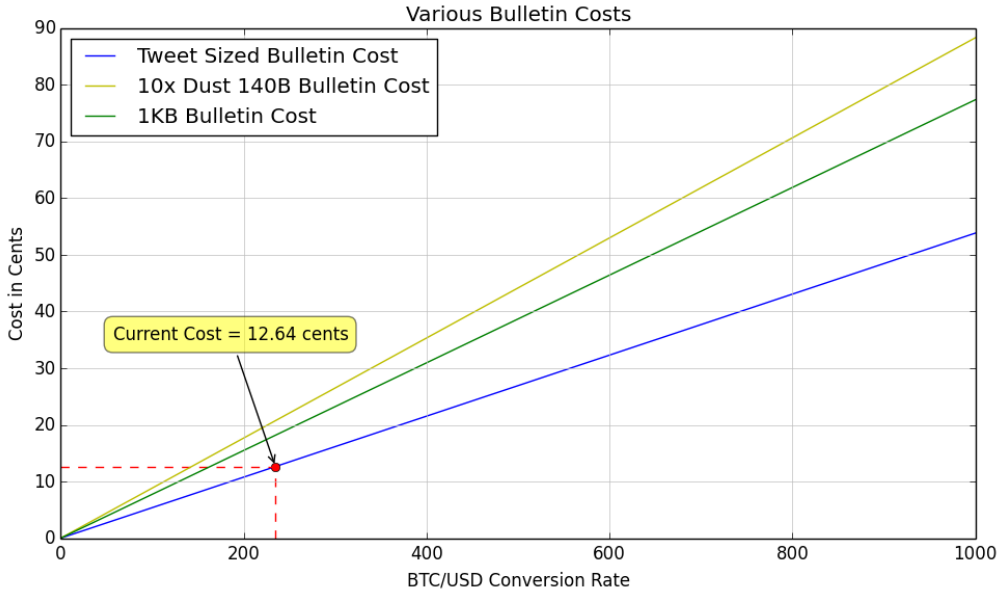

Figure 3: **Bulletin costs** reflected by manipulating variables in equation 1.

of bulletins from blocks is discussed in section 5.

To organize discussion, the public record is divided into "boards." Each bulletin must specify which board it has been posted to, otherwise it will be placed in the 'nil' board. The content of a bulletin may optionally be formatted in markdown to give it rich text formatting. We structured discussion this way to encourage threaded conversation and prevent the duplication of spam across tagged conversations.

4 Client Models

Software that interacts with and creates these messages can work in several different ways depending on resource constraints and usage needs. The more resources a machine has available the less it must trust third parties. For Bitcoin, three general client architectures have emerged: a full node, a Simple Payment Verification (SPV) client, and what we will define as a web relay. A web relay is a trusted third party running a full node that presents block chain data and bitcoin functionality as a website or web API. These models trade convenience and resource consumption for trust in other actors. To view Ombuds public statements, the same three tiered models can be applied: a modified full node, a Simple Statement Verification (SSV) client, and a web relay.

It is important to understand that none of these clients claim to hide the identity of users nor mask their use of the system. A careful user of existing anonymity tools and Bitcoin could feasibly make it extremely difficult for a determined organization to discover their identity [12], but we can make no guarantees. Additionally, while we have not specified a public key infrastructure to use with any of these clients, it is entirely possible to link bulletin signing keys to identities.

4.1 Ombuds Full Node

An Ombuds full node is a modified full node which receives every Bitcoin transaction and every Bitcoin block. The software processes all this information normally, but also extracts bulletins, and indexes them locally. Every bulletin published in the chain is stored locally by the modified node. This complete record comes at the cost of disk space, processing power, and bandwidth.

Unlike a traditional Bitcoin full node that drops transactions that take too long to confirm, unconfirmed bulletins not included in any block are still recorded locally. If the bulletins are never confirmed, the node can mark these bulletins as excluded and inform its operator of unusual and potentially malicious network behavior.

4.2 Simple Statement Verification

A Simple Statement Verification (SSV) client is a SPV client which utilizes Bloom filters[3] to "listen" for relevant wallet transactions and bulletins. A client can request to be informed of transactions and bulletins from its peers that are relevant to the client and its attached wallet. Trusting peers to honor bloom filters reduces total bandwidth usage and only recording block headers reduces total disk space [13]. Unlike an Ombuds full node, a SSV client does not have a complete history of all bulletins. Instead the client can verify that bulletins it creates are included in the global public record if it can find peers willing to honor its bloom filters. These compromises are necessary for devices with limited resources like mobile phones.

This means that a SSV client's peers could deny it relevant information and in the worst case prevent it from publishing. This can occur only if an attacker has full control over the devices ability to connect to "honest" peers. This attack is discussed further in section 5.2.

[3]A Bloom filter is a compact data-structure that tests an element's membership in a set. In Bitcoin's case, Devices like mobile phones will give their peers a semi-randomized filter so that they will receive any transactions relevant to them and a fixed percentage of all other Bitcoin transactions. The extra-transactions are included to help obfuscate the transactions the device is actually interested in.

4.3 Web Relay

A user running an Ombuds full node possesses the entire public record. This record can be formatted and delivered as an API or a web page which is viewable to anyone with a browser and an uncensored Internet connection. The person viewing the web relay's content must completely trust the operators of the relay to not censor or manipulate content.

However, if the relay signs the bulletins it serves a concurrent auditing system could verify each bulletin. Users could check the relay's responses against real responses returned by a node the auditor runs. If a difference is found the auditor then has signed proof from the relay that it is changing or withholding data. A proposal and implementation of such a protocol called WRS can be found in the references [14].

While requiring institutional trust, this model is convenient for people who are not in a hostile environment. This also gives people using the secure tool a wider audience to communicate with. Additionally, users of existing web platforms can transparently post and link to bulletins stored in the public record trusting the record to faithfully store the original bulletin, while letting a relay host it.

5 Security Analysis

There are certainly attacks on this system and Bitcoin that we are not aware of. However, there are several attacks against this system that are feasible by state-level actors who expend the capital and resources to mount them. Attacks against Bitcoin have been discussed extensively in prior work, but we will briefly analyze some of these attacks in terms of a nation-state attempting to censor bulletins. Additionally, we assume that an attacker is willing to take every measure available to them including the banning and blocking of Bitcoin at the legal and technical levels.

5.1 Excluding Transactions

As discussed above, bulletins submitted to the network may not be mined if they do not pay the fair market fee. Attackers that amass the hash can also prevent bulletins from being mined. Prior work has suggested that an attacker with less than 50% of the total hash power of the Bitcoin network could produce credible threats that further increases the attackers fraction of hash power [15]. But with close to or above 50% hash power the public record becomes manipulable [16].

Not only can an attacker arbitrarily censor bulletins by orphaning blocks that include bulletins, but an attacker could pick some arbitrary point in the past and generate blocks that do not include any bulletins, reorganizing the chain by excluding specific blocks. Nodes can detect both attacks, but there is little they can do to stop them. This is illustrated in Figure 4.

Any Ombuds node that receives a bulletin will store it and attempt to relay it to its peers, but a bulletin's inclusion in a block is subject to mining power just like regular Bitcoin transaction is. So, a bulletin could be denied indefinite entry into a block by an attacker with sufficient hash power. Bitcoin's honest hash power[4] is so high at the moment that such an attack requires a significant amount of physical and computational resources. However, recent work suggests that a state-level attack is feasible if the capital investment is made, which was estimated at around 150 million USD [17]. So if the malicious infrastructure is built or the honest hash power drops or it chooses to stop mining bulletins, exclusion from blocks becomes an existential threat to Ombuds.

5.2 Blocking Access

Like blocking access to web sites, a node's connection to peers can be filtered, blackholed, or actively reset. Just like current Internet censorship these attacks are effective to various degrees. A peer to

[4]Honest hash power in this context are miners who are observing the current consensus rules and including bulletins in their blocks.

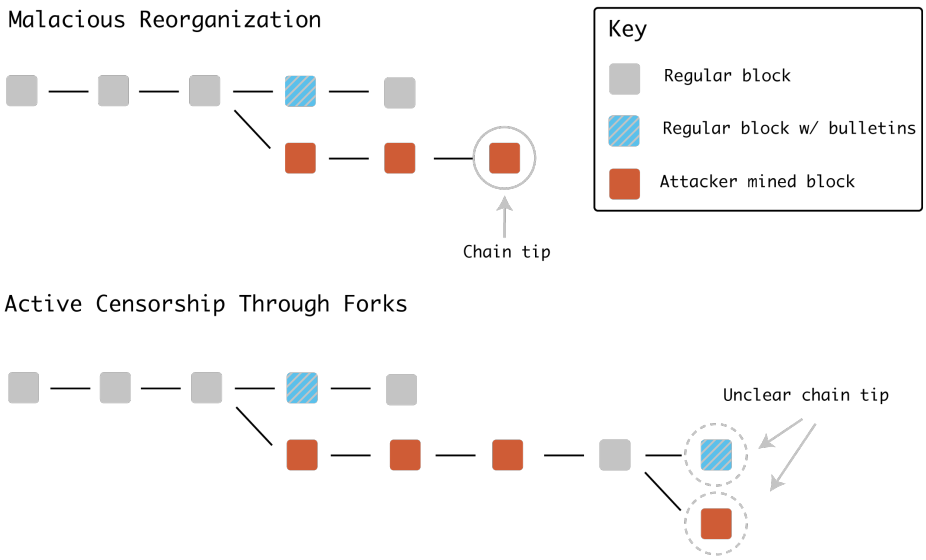

Figure 4: **A high hash power attacker** can outpace the honest chain and exclude or include bulletins at will.

peer network that uses a block chain is inherently better suited to dealing with these attacks because application data can be stored offline by peers.

A website must respond to HTTP requests to serve content to its users every time they request a page. By storing bulletins in a block chain, a dissident must only know that she has the honest chain and thus the complete public record. This means that a censor, who wants to effectively stop a user from reading what has been said, must block every possible way to distribute that block chain. This is infeasible even when a censor has complete control over internet infrastructure within its borders.

Stopping bulletins from getting out of a nation is even harder. A bulletin can be at most 10 KB in size. If the originator of that bulletin can find a single open channel to a bitcoin peer that is connected to the broader network, that bulletin can get into Bitcoin's block chain. Since that bitcoin transaction is signed and uniquely identifiable, it is a digital object that can be taken offline and brought back online when an uncensored connection is available.

5.3 Constructing Potemkin Networks

A censor could try to segment the Bitcoin network and create a fake subnet that maliciously mined blocks or intentionally dropped transactions submitted by peers. It could try to fool nodes into thinking their messages are getting out to the broader network and into the honest chain. This attack, like the last one, can be mitigated if the peer can find a single channel out to the larger honest network. From that channel bulletins can be broadcast and blocks can be brought into the affected region to subvert the attack. Additionally, if the attacker does not have comparable hash power to the honest network, its fake chain will reflect a significant drop in difficulty after it starts mining blocks.

6 Dealing with Block Chain Growth

The current maintainers of Bitcoin have taken active steps to reduce data stored within Bitcoin's block chain by disabling block chain data storage functionality [21]. For the immediate future anyone can store bulletins within the block chain, but we foresee a time when restrictions may be enforced.

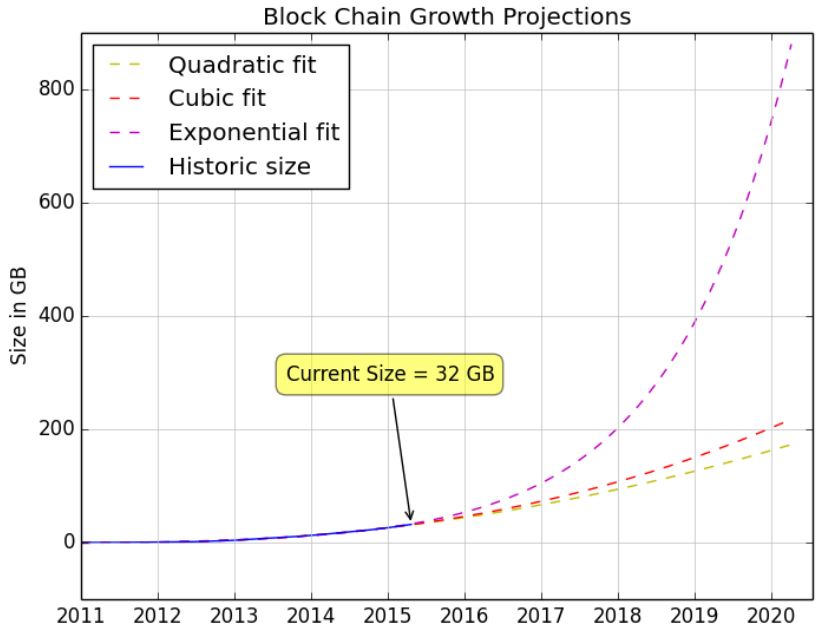

Figure 5: **Historic block chain size** projected forward with several regressed functions. Depending on the growth rate, the block chain may quickly become extremely large.

Suppose that this system ends up with 100,000 users who are actively submitting bulletins over a two year period. If every user sends 1,000 bulletins in that period, and it costs around the current market fee of 0.15 USD to send a bulletin, then from the users perspective they will spend 150 USD over two years to protect their public speech.

In this scenario, 100 million bulletins are created in total. If we assume that every bulletin is 140 characters long, each public key is 32 bytes, and each signature is 73 bytes long, then it would take $(140 + 32 + 73) * 10^8 * 2^{-30} = 22.8$ GB to store the raw data. With a 1 MB block size, the Bitcoin network can only process 7 transactions per second. If every transaction processed by the network was a bulletin, it would take about 23 weeks of exclusive mining to include all 100 million bulletins in Bitcoin's block chain.

Since every bulletin is wrapped in an unspendable Bitcoin transaction, this level of usage would grow the size of Bitcoin's block chain permanently. This kind of growth will eventually render Bitcoin's block chain impractical as a global public record. Block chains are not ideal for storing data, but if sufficient hashing infrastructure can be committed to supporting the processing of public statements, then a new block chain can be launched.

This new block chain could implement a one-way peg from Bitcoin to this new side chain. Users could insert bulletins into the side chain rather than the Bitcoin block chain. If the incentives to store and distribute blocks of that side chain are constructed properly then the storage burden could be moved off of Bitcoin and onto this independent network, without requiring any forking changes to the Bitcoin protocol.

Another option to reduce the data storage requirements is to store only a commitment to the data rather than the data itself. Factom [18] and other projects [19, 20], have taken this approach to storing commitments to arbitrarily large merkle trees[5] of data in Bitcoin's block chain. There are two issues

[5]A merkle tree is a hash tree, that composes a cryptographic hash function on every level of a tree to produce one root hash. They are used to provide compact repeatable proofs of authenticity for data included in their leaves.

we see with this approach.

The first issue is that while the commitment may be replicated and highly resistant to any attempt to remove it, the data itself is distributed by a different peer-to-peer network that may have no incentive to redundantly store that data. In cases where that data is illegal, nodes may voluntarily drop it. When a bulletin is placed in a block chain, any node that wants to fully validate the entire chain must examine that bulletin.

In essence, the strong protection of free speech that comes from storing bulletins in Bitcoin's block chain comes from the fact that bulletins are *not prunable* and therefore must be replicated. Thus any network or data structure like a distributed hash table that offloads this storage responsibility must provide a content-blind guarantee on the availability of the data stored within it even if the data stored is public.

The second issue is that a bulletin's inclusion in a commitment may be subject to moderation. The current architecture of Bitcoin incentivizes miners to mine bulletins based on a fee that is paid to a block's miner. This is egalitarian in the sense that any miner can place bulletins in Bitcoin's block chain. This means that censorship of a particular bulletin can only occur in scenarios where a large percentage of the total hash power will not mine that bulletin.

So the key aspects that must go into a more scalable system to replicate and distribute a single history of public statements are as follows: it must be content neutral, it must relinquish publishing control to peers in the network, it must be highly redundant, and it must economically incentivize these properties. And while a pegged side chain could satisfy these properties, significant hash power must be invested in it from the start or else it will be susceptible to the transaction exclusion attacks discussed in section 5.1.

7 Conclusion

Our protocol is designed to give activists and journalists a platform for free speech online that is not controlled by any single entity. It does this by using Bitcoin to store public statements that cannot be modified and are difficult to censor. This approach changes the ways in which a censor can operate and ultimately makes it more difficult for existing authorities to control speech on the Internet.

The application has been designed to store every statement in Bitcoin's block chain for several reasons explained above. Because Bitcoin cannot be a permanent host for this application, several options to move the storage burden off of Bitcoin's block chain were explored. We have not yet determined whether a side chain or a separate storage network can provide the necessary properties to support a global public record that is open to all. For that common good, we are committed to exploring new possibilities and approaches to this system and systems like it.

References

[1] Verkamp, John-Paul, and Gupta M.. "Inferring mechanics of Web censorship around the world." *In Free and Open Communications on the Internet* (2012).

[2] Clayton, Richard, Murdoch S., and Watson R. "Ignoring the great firewall of china." *In Privacy Enhancing Technologies*. Springer Berlin Heidelberg, (2006).

[3] Florio D., Andrea, et al. "Bypassing Censorship: a proven tool against the recent Internet censorship in Turkey." *Reliability and Security Data Analysis. IEEE* (2014) p 113.

[4] King, G., Pan J., and Roberts M. "How censorship in China allows government criticism but silences collective expression." *American Political Science Review 107.02* (2013). pp. 326-343.

[5] Winter, P. and Lindskog, S. "How the Great Firewall of China is Blocking Tor." *In Free and Open Communications on the Internet, USENIX* (2012).

[6] Wustrow, E., Wolchok, S., Goldber I., and Halderman A. "Telex: Anticensorship in the Network Infrastructure." *Proceedings of the 20th USENIX Security Symposium.* (Aug. 2011).

[7] The Open Internet Tools Project. "Collateral Freedom: A Snapshot of Chinese Internet Users Circumventing Censorship." (Apr. 2014). https://openitp.org/pdfs/CollateralFreedom.pdf

[8] Fu, King-wa, Chung-hong Chan, and Michael Chau. "Assessing censorship on microblogs in China: Discriminatory keyword analysis and the real-name registration policy." *In Internet Computing, IEEE 17.3* (2013). pp. 42-50.

[9] Levinson L. "Secrets, Lies And Snowden's Email: Why I Was Forced To Shut Down Lavabit." *The Guardian.* (2014).

[10] Habermas J., Lennox S., and Lennox F. "The public sphere: An encyclopedia article (1964)." *New German Critique 3* (1974). pp. 49-55.

[11] Nakamoto, S. *"Bitcoin: A Peer-to-Peer Electronic Cash System."* (May 2009).

[12] Reid, Fergal, and Harrigan M. "An analysis of anonymity in the Bitcoin system." Springer New York, (2013).

[13] Hearn M., and Corallo M. "Connection Bloom filtering" *Bitcoin Improvement Proposal 37.* (Oct. 2012)

[14] Skelsey, N. "WRS: The Web Resource Signature Extension." (Dec. 2014). http://github.com/NSkelsey/httpv/wiki

[15] Kroll J., Davey I., and Felten E. "The economics of Bitcoin mining, or Bitcoin in the presence of adversaries." *In Proceedings of WEIS volume 2013* (2013).

[16] Eyal, I., and Sirer, E. G. "Majority is not enough: Bitcoin mining is vulnerable." *In Financial Cryptography and Data Security.* Springer Berlin Heidelberg, (2014). pp. 436-454.

[17] Kim, A., Daryl S., and Soyeon Y. "The Stateless Currency and the State: An Examination of the Feasibility of a State Attack on Bitcoin" (May 2014).

[18] Factom Project. http://factom.org

[19] Proof of Existence. http://www.proofofexistence.com

[20] Gipp B., N. Meuschke, and A. Gernandt. "Decentralized Trusted Timestamping using the Crypto Currency Bitcoin." *In Proceedings of the iConference 2015*, (Mar. 2015).

[21] See the op_return discussion. https://github.com/bitcoin/bitcoin/pull/3737

```
CREATE TABLE blocks (
    hash        TEXT NOT NULL,
    prevhash    TEXT,
    height      INT,        -- The number of blocks between this one and the genesis block.
    timestamp   INT,        -- The timestamp stored as an epoch time

    PRIMARY KEY(hash)
    FOREIGN KEY(prevhash) REFERENCES blocks(hash)
);

CREATE TABLE bulletins (
    author      TEXT NOT NULL,  -- From the address of the first OutPoint used.
    txid        TEXT NOT NULL,
    board       TEXT,           -- UTF-8
    message     TEXT NOT NULL,  -- UTF-8, must have some content.
    timestamp   INT,            -- Seconds since Jan 1, 1970
    block       TEXT,

    PRIMARY KEY(txid),
    FOREIGN KEY(block) REFERENCES blocks(hash)
);
```

Figure 6: **Sqlite3 Database Schema**

```
message WireBulletin {
    optional string board       = 1;
    required string message     = 2;
    optional int64 timestamp    = 3;

}
```

Figure 7: **Protocol Buffer Message**

A Public Record Schema

Figure 6 illustrates the schema used in the public record, which is created from monitoring bitcoin transactions and blocks as they reach the modified full node. Notice that the author is attributed to the bitcoin address of the first outpoint that signs the transaction.

B Wire Bulletin Format

Ombuds uses Google's protocol buffers to specify the exact format of the structure of bulletins contained within transactions. Figure 7 specifies the structure of the message as it is converted into and out of bytes. Notice that the only required field is the message itself and the time stamp is self reported. These bytes are then fronted with a header and then sliced into Pay to Pubkey Hash transaction outputs of a Bitcoin transaction.