Archain: An Open, Irrevocable, Unforgeable and Uncensorable Archive for the Internet

Sam Williams
Will Jones

August 2, 2017

Abstract

Widespread internet access has changed almost every aspect of human life. A vast proportion of the combined knowledge of humanity is housed on the internet, and available in our pockets at any time. While the internet has had a staggering impact on the organisation of society, some fundamental flaws with the system remain. Primary among these is the ephemeral nature of the information stored on the network – it can change or disappear at any time. In this paper we present a sister network that seamlessly integrates with the world wide web, providing a permanent cryptographically verified archive for the internet. This archive makes use of a novel blockchain-derivative data structure called a blockweave, as well as a new kind of Proof of Access algorithm.

The archive that we present will be 'trustless', in that users of the archive can be confident of its validity without having to trust those that produce and maintain the system. Trust is instead deferred to cryptography and mathematics, putting the archive beyond the reach of censorship. As traditional blockchain structures would not scale to the size required for an archive of the internet, Archain makes use of a blockweave data structure, allowing scaling to arbitrary size. Archain will also allow developers to easily build custom nodes and apps that run on the network, processing its contents and using services. In order to incentivise participation in the blockweave, a system of exchangeable tokens is produced around the weave, forming the basis for a cryptocurrency. In this paper, we present not only the archive and the cryptocurrency, but we explore how the two form a mutually beneficial relationship.

1 Introduction

Human progress is led by the expansion of our realms of knowledge. Every great scientist, writer, engineer and musician has stood on the shoulders of the giants that came before them. However, in this information age we often succumb to the illusion that because information is readily available, it can never be lost, and progress reversed. This is foundationally untrue [13, 5]. While, in the internet, we have built a monumental system of de-centralised information dissemination, we have yet to build the corresponding system of permanent knowledge storage. We need to build the public library for our

new society.

Modern history is full of examples of the destruction and loss of vital information, from fires at libraries and archives [17, 18, 4, 16], to book burning in authoritarian states [9, 21, 22]. While the internet has significantly improved access to information, it has not solved the fundamental problem of making access to that information permanent. When we look up information on the internet, we are depending on being allowed access to centralised stores of that data. Access to the servers that hold this information can be revoked by their owners at any time. Similarly, as serving information on the internet requires the paying of server and upkeep costs, websites can often simply disappear when funds are no longer available. Further still, a number of governments are taking increasing steps to censor and remove access to politically sensitive information on the internet [25, 27, 8, 6].

The internet has also revolutionised the way that journalists interact with their readers. Where once we would hold a physical and irrevocable copy of the news, we now simply access the information and then discard it. It has become commonplace for media organisations to update the contents of their articles over time. While this provides a number of advantages over the previous system (most prominently, the ability to disseminate real-time updates about unfolding situations), it also allows important context to be lost or become obscured. From a historical perspective, this presents a profound challenge.

We have designed and implemented a prototype distributed, de-centralised internet archive in order to solve these problems, based on a novel blockweave data structure that improves on systems used by other cryptocurrencies like Bitcoin [19] and Ethereum [10, 26]. In the past, archives (internet or otherwise) have typically been maintained by a single institution (or even individual), making them vulnerable to two primary forms of manipulation. The first of these is through the modification of documents during their storage [3, 15]. The second is that the documents could have been forged or modified prior to their entry into storage [2] (for example, the many works attributed to Socrates that are believed to have been penned by his disciples [11]). Archain solves both of these problems. Through a distributed consensus system, the information associated with an internet URL is verified prior to entry onto the Archain. Then, once the document is stored on the blockweave, it is cryptographically linked with every other block (and document) on the weave. This ensures that any attempt to change the contents of the document will be detected and rejected by the network. In this way, no subversion of the information on the Archain is possible.

Archain is a browsable sister network to the internet, providing the long-term knowledge storage features that the internet needs but lacks. Any browser with the Archain extension installed will be able to seamlessly navigate between pages stored on servers on the normal internet, and resources stored on the Archain. When pages on the normal internet are not found, the browser extension will offer to search the Archain for archived copies of the page. Furthermore, Archain will also allow users to 'rewind' the state of a web page, and see what it looked at a previous moment in time.

Over time, as the resources on the weave expand, the system will become the trustworthy oracle of the history of the internet. As the internet is not just a resource of abstract knowledge, but also a catalogue of human events and culture, the weave will naturally come to represent a store of human experience, politics, and understand-

ing over time. Owing to the de-centralised and cryptographically verified nature of this archive, it would be beyond the reach of any organisations or groups that might conspire to censor its contents. This fundamentally changes the relationship between humanity and its past. The Archain would render the Orwellian nightmare of the 'memory hole' [20], through which the evidence of human history could be irreversibly disposed, a fundamental impossibility. Humanity need never be deprived of its history again.

A critical component of the Archain system is that interested users would be able to easily build applications that interface with and use data from the network. These apps, built with the Archain App Developer Toolkit, will act as a node in the network that ingests a live-feed of historically important information from the internet. The functions of these apps will be wide and varied, ranging from news aggregrators and discussion websites, to automated traders and uncensorable microblogging services.

In order to submit information to the weave, a small number of tokens will be required. These tokens will be used to pay miners for their work in maintaining the weave and network, as well as disincentivising the propagation of spam. Through this system, Archain crowdsources the collation of important documents for the archive. This represents a great improvement over typical archives, in which the artefacts to store are deliberated on by a small group or even a single individual. Similarly, it empowers individuals to ensure that the information they personally care about will be perpetuated into the future.

As the information on the weave matures, becoming more important from a historical standpoint, the incentive to maintain the weave also increases. This in turn further increases the likelihood that the network will propagate into the future, itself further reinforcing the value of the tokens. As these effects compound, we expect that the tokens will become a kind of gold-like asset for the information age: inseparably and intrinsically linked to a vast trove of documents relating to human experience over time, itself an invaluable asset.

2 Use Cases

There are three major families of uses for the Archain: as an untamperable archive, a method of value exchange backed by information, and a way of building applications on a real-time stream of human culture.

2.1 Currency

One of Archain's uses is as a fully functioning system of value exchange (a currency), denoted in units of ARCs. Archain enjoys many advantages over traditional currencies; transfers are highly anonymous, the entire system is decentralised and trustless (so fraud is extremely difficult), and transaction fees are minimal, to name a few. However, ARCs have a further advantage over traditional cryptocurrencies by maintaining a fundamental usefulness that underlies their value: that of the ability to store information on the blockweave.

As well as exchanging value with other users in the network, Archain will allow users to attach arbitrary length messages to transactions. These messages can optionally be encrypted using the public key (wallet address) of the recipient, creating a secure communication channel with identity verification of both parties.

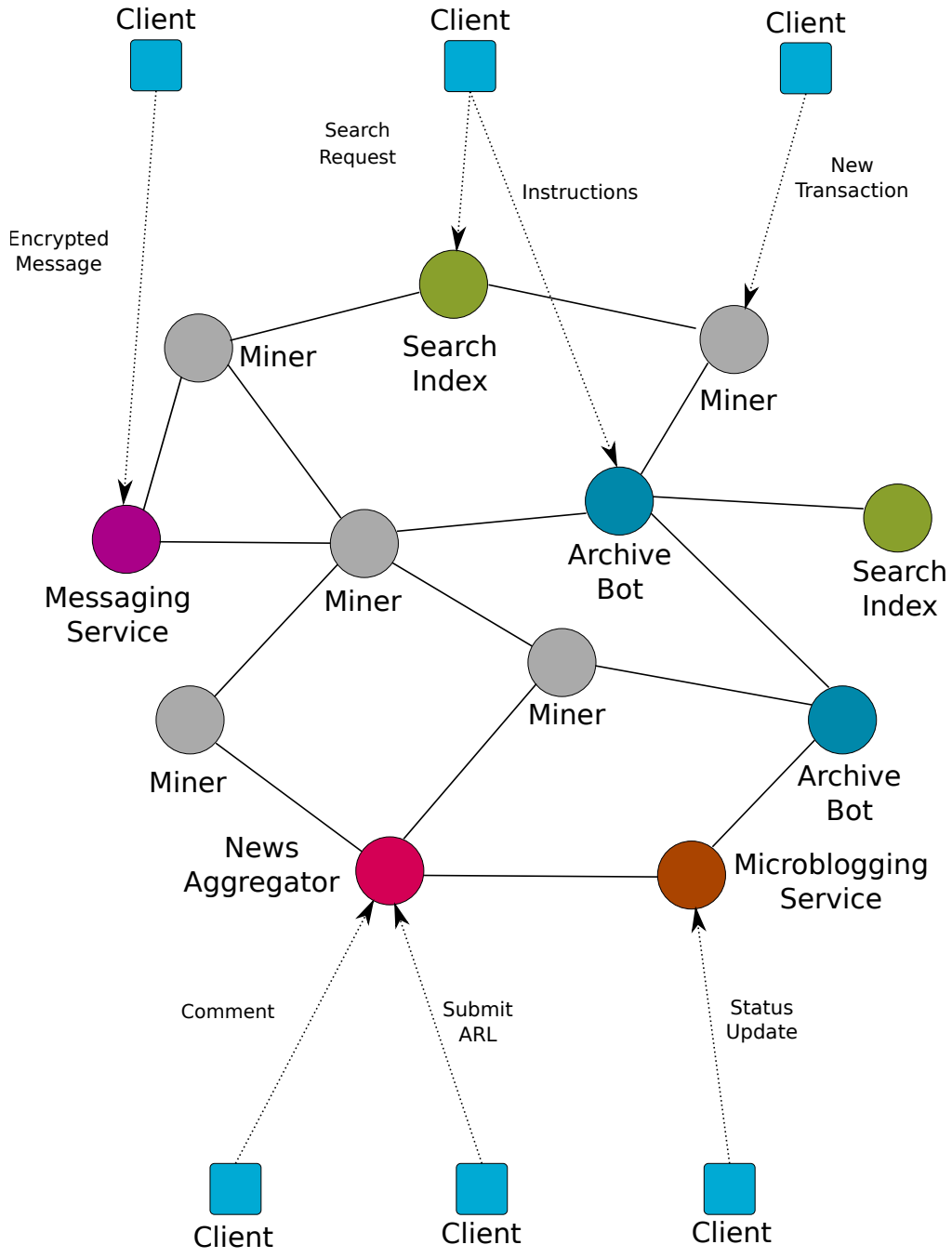

Figure 1: A representation of miners, apps and clients interacting on the Archain network.

2.2 Archiving

Before considering the many use cases for Archain archiving, we must first briefly discuss the different types of archiving operations that Archain supports. These are verified internet archiving, and unverified data archiving. While the unverified style of archiving will allow users to submit arbitrary information to the weave, with an associated name (an Archain Resource Locator, or ARL), verified internet archiving works in a different manner. Rather than submitting the contents of the page to store in the archive, an internet URL is submitted to the network and a de-centralised consensus protocol is employed to agree upon its contents before storage.

Verified internet archiving allows submitters to easily ensure that important information hosted on the internet will be available to them and others at any point in the future. These backups will be guaranteed to be free of forgery or tampering, by mechanics described in the technical sections later in this paper. These backups will be trustable by others in the future, as they will be guaranteed to be faithful representations of an internet URL's contents at a given time.

Once documents are stored on the weave, Archain users will be able to seamlessly browse between archived resources and normal internet web pages. This will allow users to view pages that have been removed from the internet, as well as allowing users to view snapshots web pages through time. In this way, the Archain will act as a 'perma-net' sibling to the traditional internet.

Verified internet archiving would be beyond the reach of censorship – giving citizens, journalists, and historians unfettered access to a vast trove of human knowledge. Equally, those uploading information to the Archain, for example victims of oppressive governments or whistleblowers, could be confident that their information will be widely disseminated and persist in an unmodified state.

Academics will also benefit from the Archain, as it will provide a verifiable way to cite the contents of web pages, at a given time. This solves a growing problem in Academia [13]: an increasing amount of useful information is being stored on the internet, but it is not possible to cite it without the potential for the contents of the pages to be modified or removed after publication.

Media organisations themselves have often been accused of modifying or removing their stories without informing their readers. This practice seems to have become more common over time. With the time-stamping and consensus based content verification systems that Archain implements, news providers can be held to account for the information that they propagate.

Bloggers will also enjoy a system that allows them to express themselves freely without being beholden to the goodwill of any particular host. Equally the readers of the blogs can rest assured that the articles that they enjoy will be available for years to come, instead of eventually disappearing into the ether, as is so often the case with small websites [13]. With encrypted weave entries as a possibility, individuals might also enjoy writing public or private journals onto the weave, potentially with the aid of a third-party Archain app. First-hand accounts of the lives of typical citizens form an important part of many historical analyses (for example, Anne Frank's diary and The Gulag Archipelago[14, 23]). Subsequently, it is expected that blogs and journals stored on the weave will be of significant utility to historians in the future. An interesting

effect of this is that bloggers and journal writers will know that their thoughts and comments will potentially form a small part of our historical understanding of the present moment, in the future.

There are many other groups who also possess an interest in storing permanent, unmodified data. For example, the Archain could also be used to store 'insurance' files: encrypted data that would be stored on the weave, which could later be easily made accessible by the release of a decryption key.

As well as providing internet and document storage, Archain can also be used for backing-up important small files. Users can then be confident that these files will available at any time in the future. These files can of course be encrypted and subsequently only accessible to the holder of the decryption key.

2.3 Apps

Archain offers developers the opportunity to build apps that act as part of the network, receiving and processing a feed of human history and culture in real-time. The App Developer Toolkit (ADT) provides a framework and library that allow developers to quickly and easily build these kinds of applications. The ADT is available from the Archain website now [1].

Archain apps that simply process transaction and block data from the weave can implemented in just a few lines of Erlang code (see listing 1). These monitoring apps could be used, for example, to build stock trading applications that process the feed of new information from the weave in order to find signals about potential changes in the trading environment. Similarly, researchers, historians and sociologists could use Archain monitor apps to discover and assess deep trends in society and culture. Journalists could also use Archain monitoring apps to capture stories as they unfold.

As well as allowing developers to build apps that monitor the weave, the ADT also allows the creation of apps that interact with the network. We envisage that developers will use the ADT to build apps that, for example, automatically archive certain web pages at regular intervals, or send messages to users on the network when certain conditions are met (for example, IoT devices reporting temperature data from sensor networks).

We anticipate the Archain ADT will be used to build a variety of social applications making use of the weave. For example, apps that allow users to discuss and collectively archive certain types of web pages and news. We anticipate that systems akin to this could be used to form special interest groups around a variety of topics, collectively processing the contents of the weave together.

We also expect that developers may be interested in building applications that utilise the weave as a permanent data storage mechanism. In this way, the Archain could be used as a kind of 'Data Permanence as a Service' system, in order to build complex apps that require long-term storage and data replication. These apps would otherwise necessitate a large investment in infrastructure development, if they are even possible to build at all.

The Archain ADT is available for developers to download and get started with immediately. Documentation as well as introductory videos and examples (progressing from the very basics of Erlang Archain apps to the development of a complex stock tip generator) are available on the Archain website [1].

We have presented only a handful of the wide

Listing 1: A simple complete Archain ADT app. This application monitors the network for new transactions that contain a given word or phrase. Such an application could be used to monitor a brand's image online or as an open source intelligence aggregation system.

```
-module(monitor_app).
-export([start/0, new_transaction/1]).

%%% An Archain monitoring app. Scans new transactions for
%%% archived web pages that match a given regular expression.

%% Start a new monitor server, calling back to this module.
start() ->
        adt:start(?MODULE).

%% This function is called and passed a new transaction every
%% time one is added to the blockweave.
new_transaction(T) ->
        % Scan the new transaction's data chunk for strings that
        % match our target regular expression.
        case re:run("REGEX", T#tx.data) of
                {match, _} ->
                        report(T);
                _ -> ok
        end.

%% Report that a new transaction matches our criteria.
%% In practise this function could be modified to send an alert
%% email or create a record of the transaction in a database.
report(T) ->
        % Log the transaction information to the console.
        io:format("TX ~p matches selection criteria!~n", [T#tx.id]).
```

and varied uses of Archain in this section. We fully expect that users will find and build many more applications that utilise the various functions of the Archain.

3 Proof of Access and the Blockweave

Archain employs a novel mining system that offers a number of major benefits over previous algorithms, while maintaining the security properties of the trusted SHA2-256 [12] Proof of Work (PoW) system (used by Bitcoin [19]). The Archain Proof of Access (PoA) algorithm essentially incorporates data from a deterministically randomly chosen previous block in the calculation of new blocks. This system means that the amount of the Archain that a miner has available to it is directly proportional to the amount of time they can spend constructively mining. Further, the PoA system encourages miners to store blocks that are not widely mirrored by other nodes in the network. In this way, the system self-organises in order to ensure an even replication of all of the blocks on the blockweave.

Archain is a distributed archive, whose content is expected to grow to extraordinary size. Typical cryptocurrencies require miners and nodes in the network to maintain a large proportion, if not the entire blockchain in local storage. Archain's novel PoA system is designed to cope with block structures that will grow too large for a single individual to practically host, while still maintaining the cryptographic verifiability of traditional blockchains.

3.1 Algorithm

While typical PoW systems only depend on the previous block in order to generate each successive block, the PoA algorithm incorporates data from a randomly chosen previous block. In this way, the data structure at the heart of the Archain is in fact a kind of 'weave' of blocks (a blockweave), rather than a traditional chain of blocks (a blockchain). The 'recall block' to incorporate into the next block is chosen by taking the hash of the current block and calculating its modulus with respect to the current block height. The transactions in the recall block are hashed alongside those found in the current block in order to generate the next block.

When a miner finds an appropriate hash, they distribute the new block along with the recall block to other members of the network. This allows the other members of the network, even those without their own copy of the recall block, to independently verify that the new block is valid.

3.1.1 Synchronisation Blocks and Lists

As the weave will not be stored in its complete form on every miner's machine, a system of lists and synchronisation blocks is employed in order to allow the verification checks that the weave requires. Synchronisation blocks are special blocks generated once every set period (Archain will start at once every 12 blocks – once an hour). Synchronisation blocks contain a full list of the balance of every wallet in the system and a hash for every previous block, and no transactions. Miners maintain their own copies of these lists, updating them with new information from every new block they accept. Synchronisation blocks are then only accepted by the network if the lists

stored in the block match precisely the lists that the miners maintain.

Instead of having each miner verify the entire block structure from the genesis block to the current block when they join the network, Archain uses a system of 'ongoing verification'. When miners join the Archain network, they will download each previous block from the current block to the last synchronisation block. New blocks are verified by every miner in the system, ensuring that new miners can start operating without immediately verifying the entire weave themselves. Full weave verification is of course available to any node that wishes to perform it.

In order to allow new transactions to be verified by miners that do not have access to the full weave, a list of wallets and associated balances is maintained by each miner, and included and verified in each synchronisation block. In this way, miners do not need to find the previous transaction associated with a wallet in order to verify a new transaction. Instead, miners would simply need to verify that the transaction has been appropriately signed by the wallet owner's private key.

To prevent recall block forging attacks, a list of hashes of each of the previous block's contents is distributed with every new synchronisation block. Recall blocks received from miners claiming to have found a new block can be verified by checking the contents against the hash from the list. As this data is repeated for each successive synchronisation block, each node only needs to store the most recent hash list.

3.2 Benefits

This approach provides a number of highly significant benefits over previous PoW systems. The most significant of these advantages is that it allows true de-centralisation of a potentially massive block structure, with miners individually incentivised to provide the most even possible storage and redundancy of data across the weave. These incentives scale with the risk of block loss. In the scenario in which a block is only accessible by a single node, other miners become highly incentivised to replicate that block. Storage of rare blocks ensures the ability to mine among a much smaller proportion of the hashing power of the network – increasing the likelihood of finding a block. This increases the financial incentive for the block to be stored. Even in such a scenario that a recall block appears lost, the incentive to find old copies of that block (perhaps on machines no longer attached to the network) becomes extraordinary, as it almost guarantees the mining reward. As the recall block must be distributed with the newly mined blocks in order to be accepted onto the network, rare blocks are unlikely to stay rare for very long.

As the Archain grows it will become infeasible to store the entire weave on one machine. This will lead miners to spend time waiting for blocks to be mined that require a recall block they do not have access to. As the proportion of the weave that is stored by the individual miners decreases, so too will the effective hashing rate of the network. As a result of this, the difficulty of the PoW part of the PoA algorithm will decrease, keeping the block time consistent but consuming significantly less electricity, at a network level. This has significant positive effects on the environmental scalability of the system. Alternatively, miners could chose to mine Bitcoin or another cryptocurrency with their idle hardware in periods where they cannot mine on the Archain network.

The Archain PoA system also provides an incentive for miners to price storage appropriately.

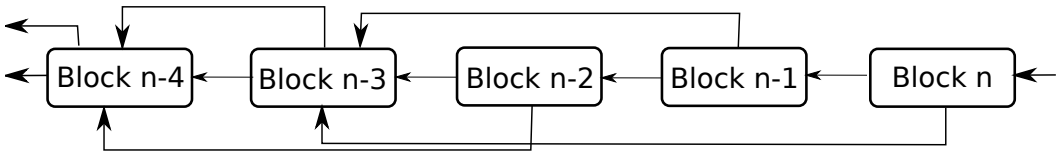

Figure 2: An Archain cryptographic blockweave, with arrows denoting dependence.

Miners in the Archain network will understand that they will have to store the data, or have their idle time increase (decreasing their profit), discouraging them from accepting large transactions too cheaply. Further, other nodes in the network will not accept blocks that contain large transactions that are too cheap. As the miners will not receive their rewards if the block is not accepted by the wider network, they are further incentivised not to allocate storage too cheaply. A converse pressure is also provided by the fact that they will not receive any mining rewards for transactions they do not accept (disincentivising inappropriately high fees for storage). In this way, miners are selfishly encouraged to price data storage sensibly.

3.3 Discussion

3.3.1 Storage Requirements

The only added storage requirement of the PoA system is the addition of the wallet and hash list, to be stored alongside blocks from the weave. Miners need only store one copy of each of these lists, drastically reducing the storage implications of the PoA mechanism. The effect of adding these lists to each synchronisation block will increase by roughly 13mb every year, assuming a Bitcoin-like growth trajectory. The size of the synchronisation blocks will be lowered by the lack of inclusion of transactions – likely making them smaller or similar in size to normal transactions. While these costs are not insignificant, we expect that because Archain is a storage network, they will be dwarfed by data storage transactions. Additionally, we expect that network interconnect speeds will similarly outpace the bandwidth cost growth over time. Finally, as the number of wallets and block hashes grows, the period between synchronisation will be adjusted.

3.3.2 New Wallet Fees

As the full wallet and balance list must be distributed in every synchronisation block, precautions must be taken against the needless creation of new wallets. In order to solve this problem, Archain charges a new wallet fee. This fee will be a small multiple of the basic transaction fee. In this way, users are highly discouraged from producing wallets that they do not intend to use.

3.3.3 Storage Pools

One potential theoretical attack against an Archain that has become extraordinarily large is that miners may work co-operatively to maintain a single copy of the weave, which they all access to retrieve recall blocks. While this kind of behaviour may at first seem problematic, this is not in fact the case. If such 'storage pools' were employed by a large proportion of the miners, the incentive for other miners to store rare blocks increases. This is because if the centralised stores

become unavailable, miners with a copy of the rare blocks will be highly likely to receive the reward when that block becomes the recall block in the future. This self-interested behaviour provides a risk-offsetting function to the network, which scales as the potential for data loss (caused by centralised storage pools) grows.

4 Building Apps with Archain

Many recent cryptocurrencies have attempted to create systems in which the network can perform arbitrary computation and application hosting services [26]. While these cryptocurrencies (particularly Ethereum) have gained extraordinary popularity, the utility of their application hosting mechanisms remains unclear. In light of the many problems associated with on-chain application development (for example, the difficulties associated with getting data to process into the systems), Archain takes a slightly different approach. Rather than attempting to be a 'global computer', Archain instead provides a 'global harddrive' and the ability to interface directly with traffic on the network.

The Archain app hosting mechanism allows developers to build applications in a more familiar manner, while maintaining the majority of the benefits of app deployment on a cryptocurrency network. Archain apps are currently written in Erlang or Elixir (languages with exceptional support for distributed computation and fault tolerance [7, 24]), but support for other languages will be introduced in the future. Apps developed for the Archain system become members of the network, and subsequently can send and receive messages in the network. This allows easy access to Archain's distributed 'Permanence as a Service' mechanism, as well as allowing apps to read the new data that is being stored on the system.

Apps built on the Archain network can be decentralised as they share a negotiated, distributed storage mechanism. Subsequently multiple instances of the app can be run simultaneously at any one time, with the application state shared between them. The Archain app developer toolkit features a number of sample applications, including a stock intelligence generator and a decentralised, uncensorable microblogging platform.

5 Technical Details

In this section we will first describe the basic cryptographic mechanisms employed by Archain, then detail how these systems are used in the implementation of the system. Those already comfortable with hashing algorithms, public-private key encryption and signing may wish to skip this primer.

5.1 Cryptography Primer

The core cryptographic methods used in the Archain are hashing and public-private key cryptography. In particular, the digital signing mechanisms that public-private key cryptography allows. A hashing function takes a number of input bytes and returns a fixed length arbitrary value. If the function is given the same input again, it will re-produce exactly the same output (the 'hash' of the value). In the case of a cryptographically secure hash function (like the one employed by Archain), particular attention is paid to ensuring that the algorithm is not easily reversed. That is, given the an output hash, one cannot easily find the input value.

Public-private key cryptography is based on two asymmetric keys, known as the public key and the private key. Once a message has been encrypted by one key, the only way to decrypt the message is with the other key. The canonical use of the public-private key system is in private communication. To use the public-private key system, a user (say, Alice) makes one key public while the other is kept private. Anybody (say, Bob) who wishes to communicate privately with Alice can encrypt their message with Alice's public key, and then broadcast it. As only the holder of Alice's private key can decrypt the message, it remains secret despite the encrypted text being transmitted publicly.

However, public-private key cryptography is not just limited to secret communication, it also has another important use in the area of digital signing. Digital signing allows one to cryptographically prove that a message has not been modified, through public-private key cryptography. It also proves the identity of the signer. If Alice wishes to digitally sign a message, she does the opposite of the procedure for encrypting a secret message, she encrypts it with her private key. Now, anybody who wishes to be sure that it really was Alice who sent the message (not an imposter) can use her public key to decrypt the message. This gives them certainty that:

- Alice was the sender of the message (since only Alice knows the private key linked to her public key), and
- The content of the message has not been modified since Alice signed it, because the only way to modify the message would be to possess Alice's private key, which once again, is only known to her.

Hashing is often an important addition to digital signatures. Often, a message will be hashed and then the hash signed with the private key. Even though the hash is a one way function, it is still possible to check that the signature originated from the sender by applying the same hashing to the contents of the message and comparing it to the decrypted signature. The major benefit of this approach is that the output of the hashing function is a fixed, short length, regardless of the size of the input. In many practical applications (such as Archain) this is important as encrypting large amounts of data is a time consuming process.

5.2 Archain Design and Implementation

Hashing and public-private key signing form the basis of security in Archain. Each user creates a 'wallet', consisting of a public and private key pair. The public key is the publicly disclosable 'address' of the wallet and the private key secures its ownership. ARCs are transferred by the wallet owner digitally signing a transaction (which may optionally include data and verification chunks), which is then verified and placed onto the next block in the weave by miners in the system. Miners will not accept transactions that attempt to spend a greater number of tokens than are found in the outputs of the previous transaction involving the sender's wallet address. As token holdings are only identified by public and private keys, new wallets can be created arbitrarily. If care is taken to separate key identity from personal identity, this system offers a high level of privacy.

In order to solve the problem of transaction ordering, we implement a blockweave. Each node in the network attempts to create a block containing all of the unconfirmed transactions it is

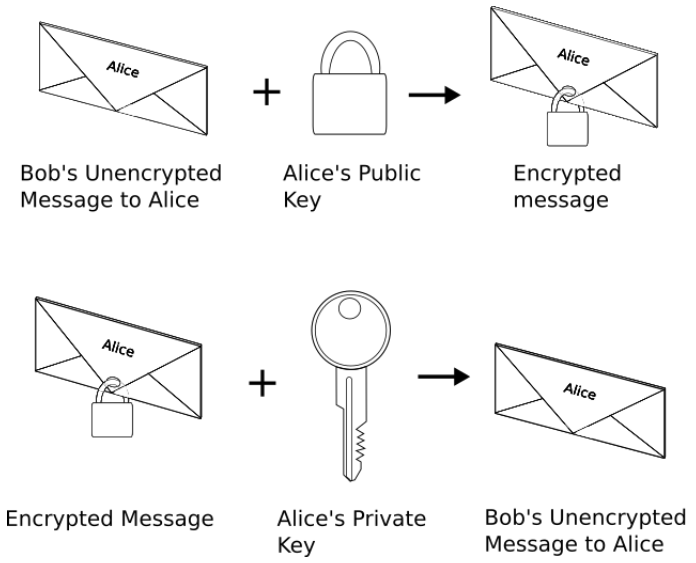

Figure 3: Public-private key encryption schemes.

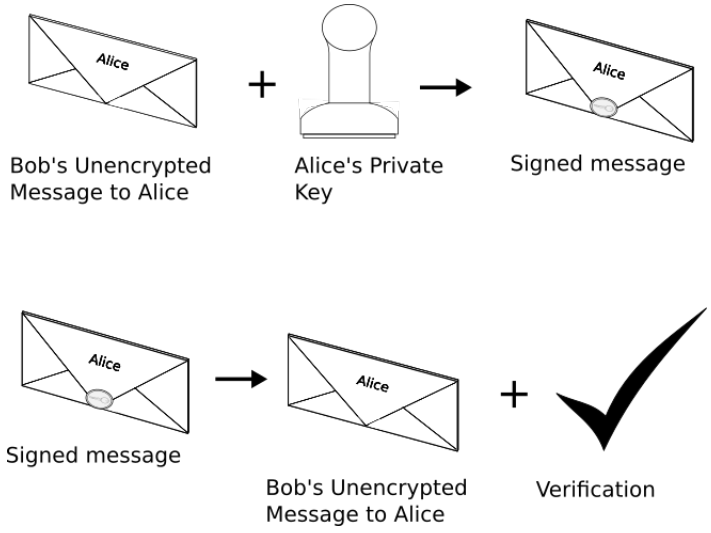

Figure 4: Cryptographic signing with public-private key cryptography.

aware of, as well as a hash based on the content of the new block, the previous block and the recall block. In this way, each transaction is cryptographically 'weaved' into the existing block structure. This serialises the transactions and allows Archain to identify 'double spending' attacks.

During the 'mining' of each new block the following Proof of Access process is followed. First, a deterministic method is employed to randomly select a previous block in the weave to use as the 'recall' block. Next, the nodes in the Archain network that have a stored copy of the recall block race to find a hash that begins with a number (referred to as the difficulty') of zeros at the start. Finding these hashes is computationally expensive. This is known as Proof of Work (a sub-component of the Archain Proof of Access system), and effectively weights the chance of a node being chosen to form the next block by the amount of computational power they provide. Our choice of Proof of Work algorithm is SHA2-256, which has been extensively been shown to be secure [12].

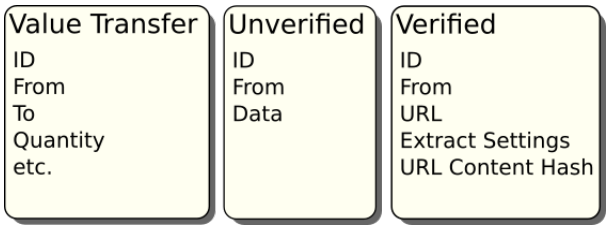

Figure 5: Archain transaction types.

In order to incentivise mining, the node that successfully 'mines' each block is rewarded with a number of ARC token. Clearly, as computational power increases, if the difficulty of the hashing problems stays the same then this behaviour has the potential to drive inflation. Therefore, Archain re-adjusts the difficulty of the hash required for each block based on the amount of time the previous block took to mine, normalising around a solution time of 5 minutes. As an additional counter to inflation, the amount of currency issued for each successive mining block decreases such that 10% of the tokens in the genesis block are issued in the first year, halving every year thereafter.

5.3 Archiving Data

In order to submit data to the Archain, users must send a special kind of transaction, containing either an internet URL or a chunk of data. As well as verified internet archiving, Archain allows users to perform unverified archiving – storing data on the weave without external supervision. This mode allows arbitrary submission of information. We anticipate that Archain users will use this mode in order to write blogs, journals, and backup important data. The validity of unverified archiving data submitted to the blockweave is secured in the same manner as normal transactions: it is digitally signed by the private key of the sender.

While unverified archiving is powerful, it makes no assertions about the validity of the information that is submitted. Verified archiving allows users to take a snapshot of a web page that cannot be faked or altered by either the person requesting the upload or the node adding it to the weave. In this case, instead of submitting raw data to the network, a URL is submitted, along with a set of parameters for extracting data from the page, and a digitally signed hash of the expected contents of the resource. The miner then downloads and processes the web page, according to the instructions of the client. If the hash of the node's version of the page matches

the hash given to them by the client, the contents that the miner has downloaded is submitted to the blockweave. When a newly mined block is distributed in the network, each node confirms that every verified archiving transaction's client-signed hash matches the miner-provided content, before accepting the block. In this manner, we can be confident that the webpage uploaded to the archive is exactly as it was on the internet at the time of submission.

As Archain is built to function over the period of tens or hundreds of years, minimising the size of the weave is important. In order to minimise the size of each individual resource, Archain will focus on the storage of text data. While arbitrary data storage will be possible, the price will be considerable (and proportionate to the size). Archain will include a universal data extraction mechanism that condenses web pages into a much smaller text and hyperlink only format, radically reducing the cost of storage.

Given the size of image data, some clients will not wish to upload raw images. However, an image often tells a thousand words, therefore the Archain data extractor will be able to replace images with hashes, and store the images themselves off-weave, on the client's computer. As the hash of the images will be stored on the weave, the off-weave image can be verified as legitimately belonging to the web page at any later point. This will provide users with a cheaper alternative to storing web pages, while also maintaining the richness of multimedia web pages.

Much of the success of the internet is based on its explicit system of referencing and connecting web pages. We have engineered Archain with the ability to maintain hyperlinks between other archived documents and typical web pages. This will allow users to explore the Archain as an extension of the internet, freely switching between Archain pages and normal web sites. Further, as verified submissions will transfer their previous internet-based addresses to the Archain, archived pages no longer available from the internet can be seamlessly replaced with their Archain-based copies.

In order to protect miners, the default node behaviour will only accept archiving requests from a large pre-approved list of websites (likely, the top 100,000 non-adult websites). This 'whitelist' can be disabled by miners that wish to aid users submitting less popular content to the weave. Further, miners can choose not to store certain blocks, or to ignore blocks whose contents match a given criteria. They will, however, then not be able to take part in mining of new blocks for which an ignored block is chosen as the 'recall' block.

5.4 Browser Integration

Users will typically interact with the Archain using the web browser extension. This extension will seamlessly integrate the Archain perma-net' alongside the normal internet, allowing users to browse Archain resources as if they were any other website.

As Archain addresses will be understood by any Archain equipped browser, users will be able to share and link to Archain resources as if they were normal web addresses. Archain resources can even be included as hyperlinks on the web. We hope that this will help foster the creation of new online communities to discuss information that is stored on the chain. We envisage, for example, content aggregation and bulletin board forums in which users swap and comment on interesting Archain resources around a given topic.

The Archain browser extension will achieve this internet interoperability by allowing the

browser to interpret 'archain[s]://' web addresses, and redirect these to the weave. Typically, Archain resources will be named as they are found on the internet, so switching to an Archain copy of an internet page will be as simple as exchanging the 'http[s]://' protocol statement with the appropriate Archain one.

Additionally, Archain offers searching functionality that allows users to explore the archive. For example, should a user encounter a 404 ('not found') error when trying to access a given webpage, the browser extension will offer to search the weave for this resource automatically.

Finally and most fundamentally, the Archain browser extension will also allow effortless uploading of data to the blockweave. Within a few clicks, users will be able to submit URLs to the weave for storage. As well as submitting web page backup requests, the extension will also allow users to submit unverified archiving transactions to the weave.

5.5 Prototype and App Developer Toolkit

Development of the Archain is already well underway. Our prototype system, written in Erlang, implements all of the core features of the Archain network and blockweave, as well as allowing for simulations of large-scale networks.

The current version of the prototype implements all of the core methods of the Archain archive (including the Proof of Access algorithm) and cryptocurrency that we have described in this paper. Verified submission of web pages is in active development.

The Archain App Developer Toolkit is included in the Archain prototype, available now from the Archain website [1]. The Archain prototype allows users to develop and test apps on a simulated Archain network prior to the seeding of the initial Genesis block. More information, getting started guides and tutorials about Archain app development are available on the website.

6 Conclusion

In this paper we have presented the case for Archain: a de-centralised, cryptographically verified archival network for the internet, built on a new kind of block storage technology and mining algorithm. This blockweave technology is capable of scaling to sizes untenable with traditional blockchain-based systems. We have demonstrated the urgent need for such a system in our modern world, as well as elucidating a technical path to implementing this service. We foresee Archain as a powerful tool for combating the spread of censorship, and ensuring the protection of historical knowledge. Development of the Archain as an open source system is well underway and delivery is expected within a year of writing.

References

[1] Archain. `https://www.archain.org`. Accessed: 2017-07-30.

[2] The national archives: Investigation into forged documents discovered amongst authentic public records: Documents purporting to have been created by members of the british government and members of the british armed services relating to leading nazis figures and axis power governments. `http://discovery.nationalarchives.`

gov.uk/details/r/C16525. Accessed: 2017-07-30.

[3] North's ex-secretary tells of altering memos. http://www.nytimes.com/1989/03/23/us/north-s-ex-secretary-tells-of-altering-memos.html. Accessed: 2017-07-30.

[4] The patent fire of 1836. http://patent.laws.com/patent-act-of-1836/patent-act-of-1836-patent-fire-of-1836. Accessed: 2017-07-30.

[5] Raiders of the lost web: If a pulitzer-finalist 34-part series of investigative journalism can vanish from the web, anything can. https://www.theatlantic.com/technology/archive/2015/10/raiders-of-the-lost-web/409210/. Accessed: 2017-07-30.

[6] Mustafa Akgul and Melih Kirlidog. Internet censorship in turkey. *Internet Policy Review*, 4(2):1–22, 2015.

[7] Joe Armstrong. Erlang. *Communications of the ACM*, 53(9):68–75, 2010.

[8] Simurgh Aryan, Homa Aryan, and J Alex Halderman. Internet censorship in iran: A first look. In *FOCI*, 2013.

[9] Fernando Baez. *A universal history of the destruction of books: From ancient Sumer to modern Iraq*. Atlas Books, 2008.

[10] Vitalik Buterin et al. Ethereum white paper, 2013.

[11] Anton-Hermann Chroust. Socrates–a source problem. *The New Scholasticism*, 19(1):48–72, 1945.

[12] Quynh H Dang. Secure hash standard (shs). *Federal Inf. Process. Stds.(NIST FIPS)-180-4*, 2012.

[13] Robert P Dellavalle, Eric J Hester, Lauren F Heilig, Amanda L Drake, Jeff W Kuntzman, Marla Graber, and Lisa M Schilling. Going, going, gone: Lost internet references. *Science*, 302(5646):787–788, 2003.

[14] Anne Frank and Storm Jameson. *Anne Frank's diary*. Vallentine, mitchell, 1971.

[15] Gerhard A Gesell. United states v. oliver l. north. *Federal Sentencing Reporter*, 2(2):59–64, 1989.

[16] Joan van Hoeven, Hans van der; Albada. Lost memory: libraries and archives destroyed in the twentieth century. 1996.

[17] Brewster Kahle. Fire update: Lost many cameras, 20 boxes. no one hurt. https://blog.archive.org/2013/11/06/scanning-center-fire-please-help-rebuild/. Accessed: 2017-07-30.

[18] Birmingham Public Libraries. *Notes on the history of the Birmingham Public Libraries, 1861-1961*. Birmingham Public Libraries Birmingham, 1962.

[19] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[20] George Orwell. *Nineteen eighty four*. Prabhat Prakashan, 2001.

[21] James M Ritchie. The nazi book-burning. *The Modern Language Review*, pages 627–643, 1988.

[22] Jonathan Rose. *The holocaust and the book: destruction and preservation*. Univ of Massachusetts Press, 2008.

[23] Aleksandr Solzhenitsyn. *The Gulag Archipelago, 1918-56: An Experiment in Literary Investigation*, volume 3. Random House, 2003.

[24] Dave Thomas. *Programming Elixir 1.2: Functional*. Pragmatic Bookshelf, 2016.

[25] Barney Warf. Geographies of global internet censorship. *GeoJournal*, 76(1):1–23, 2011.

[26] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 151, 2014.

[27] Xueyang Xu, Z. Morley Mao, and J. Alex Halderman. *Internet Censorship in China: Where Does the Filtering Occur?*, pages 133–142. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.