Mainframe: the web3 communications layer

Adam Clarke, Austin Craig, Brad Hagen, Carl Youngblood, Clément Jaquier, Diogo Perillo, Luca Tavazzani, Matt Larson, Mick Hagen, Miloš Mošić, Paul Le Cam, Shane Howley, PhD

Abstract

Weaknesses in existing Internet protocols make it impossible to provide robust network security. Mainframe is an incentivized and fully decentralized communications layer that enables reliable, secure packet routing, packet delivery, packet holding, file storage and data services. Its security model not only provides encryption, but also resistance to censorship and surveillance. A detailed description of the Mainframe platform, incentivization model, token economics, and development roadmap is provided.

Keywords: protocols, networking, communications, messaging, tokens, economics, security, cryptography, decentralization, blockchains, smart contracts

mainframe

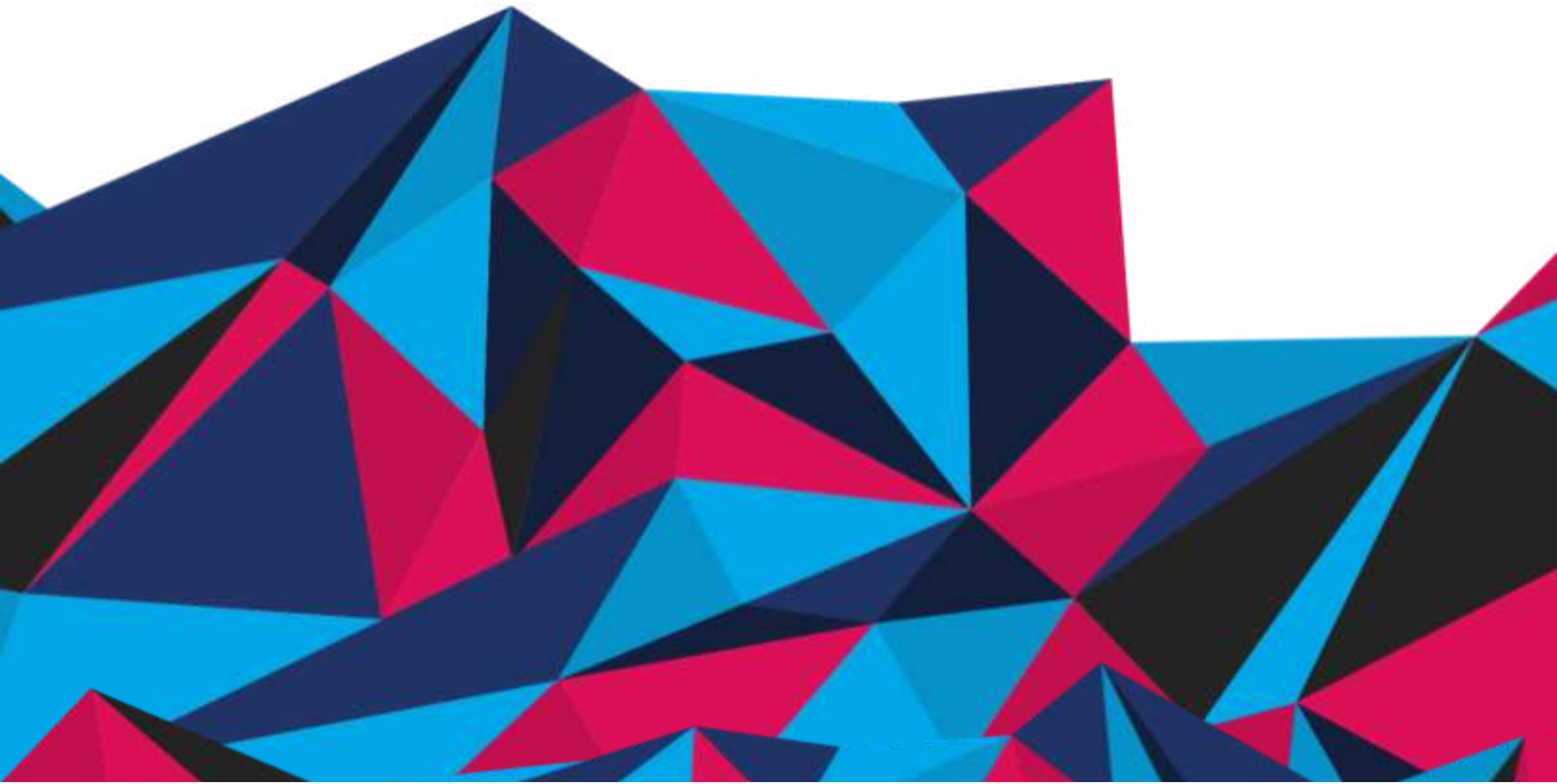
The web3 communications layer

mainframe.com

Contributors

Adam Clarke . Austin Craig . Brad Hagen . Carl Youngblood . Clément Jaquier . Diogo Perillo
Luca Tavazzani . Matt Larson . Mick Hagen . Miloš Mošić . Paul Le Cam . Shane Howley, PhD

Whitepaper Version 1.0



Mainframe

Unstoppable communications

Since its public debut in the mid-nineties, the Internet has enabled unprecedented innovation and creativity, connecting all manner of devices and supporting novel applications far beyond the expectations of its inventors. Nearly four billion people use it today—over half of the world's population.

Overwhelming success also introduced new challenges. Multiple third parties mediate users' information, and often aren't aligned with the users' best interests. Government surveillance and malicious agents pose additional threats. Each takes advantage of the same decades-old architecture on which the entire Internet is built. This threatens privacy rights and the well-being of anyone working with valuable information. In the corporate world alone, recent headlines show security breaches cause stock markets to plummet, deals to fall through, and company secrets to leak. There is little hope of fully securing these holes using existing methods.

New tools are emerging to solve these problems more effectively. Mainframe is an unstoppable communications platform providing unparalleled security, including:

- **Data privacy:** Full end-to-end and group encryption
- **Surveillance resistance:** Node relationships can be hidden
- **Censorship resistance:** Hidden node relationships prevent communications jamming

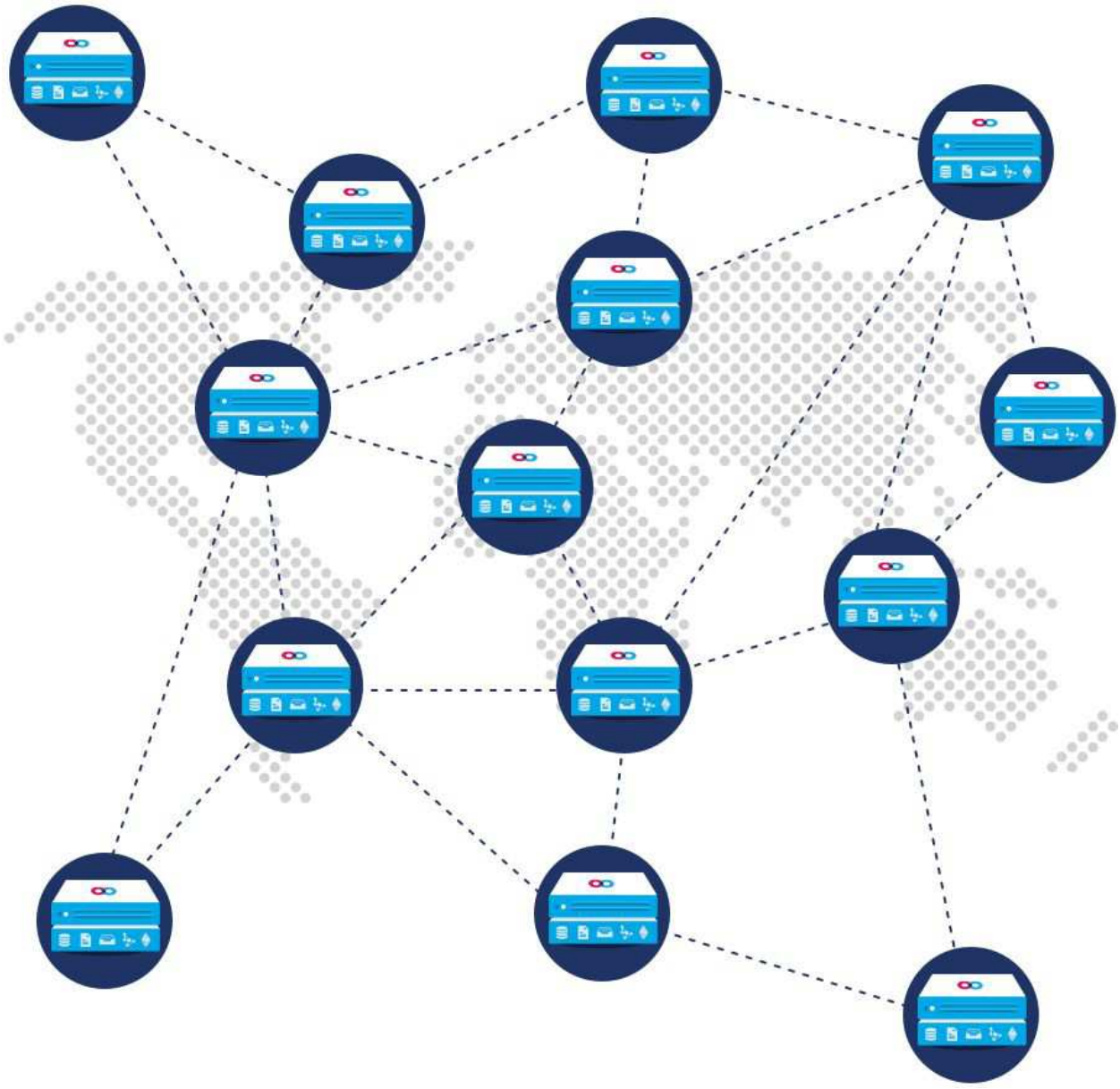
"No right of private conversation was enumerated in the [United States] Constitution. I don't suppose it occurred to anyone at the time that it could be prevented."

Whitfield Diffie
Co-inventor of public-key cryptography

Mainframe's communications infrastructure is entirely decentralized and unhosted. A token economy incentivizes network performance, including:

- Incentivized packet routing
- Incentivized packet storage for later retrieval
- Incentivized file storage
- Incentivized data services

Mainframe provides unprecedented communications security without compromising convenience.



Centralization threatens privacy and freedom

In the genesis block of the bitcoin blockchain, creator Satoshi Nakamoto inserted this note:

"The Times 03/Jan/2009 Chancellor on brink of second bailout for banks"

There was no further explanation or commentary, but it is widely believed the day's headline from The Times of London[1] was included for two reasons: it served to link the birth of Bitcoin to a verifiable historical event; and it indicated the general climate of distrust, abuse of authority, and institutional failure.

A declining trust in governments, corporations, and media worldwide[2] can be clearly observed during the period from 2008 to 2013. This period witnessed economic collapse and government bailouts, as well as the Arab Spring movement, in which millions of citizens organized through platforms like Facebook and WhatsApp, demanding more from their governments. Thousands were killed as regimes were toppled and countries fell into civil war. Early Arab Spring protests inspired the Occupy Wall Street movement, which was also largely organized through social platforms. That movement involved participants from 82 countries and nearly 1,000 cities throughout 2012.[3] It wasn't until the summer of 2013 that the revelations of Edward Snowden exposed dragnet surveillance on US citizens and the world, spanning virtually all platforms and modes of communication. Each of these events highlights the importance of fully decentralized communications, without reliance on third parties, to protect user information. In some cases, the results are a matter of life and death.

Even in the workplace, centralized architecture comes at a huge cost. Regular downtime is expensive and inconvenient, but pales in comparison to malicious threats. The refrain "data is the new oil" is repeated by WIRED,[4] Fortune,[5] IBM,[6] and others. Data theft and leakage result in lost value. Industrial espionage is difficult to quantify, but Juniper Research estimates that the cost of data breaches will reach $2.1 trillion globally by 2019, with the average cost per business to exceed $150 million by 2020.[7] A clear example is Nortel, was one of the world's largest telecoms equipment manufacturers at its peak in 2000, with nearly 100,000 employees and $30 billion in revenues. It declared bankruptcy just nine years later. Nortel's own head of security and others have blamed this colossal failure on serial IP theft from state sponsored Chinese hackers.[8] Reports claim that hackers gained direct access to the email accounts and files of high-level executives.

We observe similar threats on a regular basis. The Yahoo breach of 2013 compromised all 3 billion Yahoo accounts and is still the widest breach in history.[9] Yahoo was aware of the breach for years before publicly disclosing anything or even resetting users' passwords. The Sony Pictures breach of 2014 exposed 100 terabytes of email, passwords, social security numbers, finances, marketing plans, four entire unreleased Sony films, and more.[10] In May of 2016, hackers gained access to the email account of John Podesta, campaign manager for US presidential candidate Hillary Clinton.[11] The contents of his email were published by Wikileaks, contributing to the most surprising election upset in US history.

Most alarming is the fact that communication and financial data are increasingly controlled by fewer organizations. Only three companies, Google, Apple and Microsoft, power the email clients receiving over 85% of the trillions of emails sent globally each year.[12] In many instances, these companies' interests conflict with the well-being of their customers. Their fiduciary duty to drive revenues and serve advertisers often gets in the way. When users do not pay for their services, they are not the customer; they and their personal data are the product.

Scott Galloway of L2inc illustrates the problem: "When Facebook sought approval of the acquisition of WhatsApp, they assured EU regulators that it would be impossible for the two entities to share data in the short term. This assuaged regulators' concerns over privacy, and the acquisition was approved. Spoiler alert: Facebook figured out pretty fast how data could jump silos." After paying $19B for WhatsApp, the EU fined Facebook $122M. "This is tantamount to getting a $10 parking ticket for not feeding a meter that costs $100 every 15 minutes."[13]

Communication is part of nearly every human activity. With the above mentioned threats, and software eating the world,[14] the underpinnings of individual and institutional privacy are threatened. Along with them would go any meaningful notion of sovereignty or autonomy. These structural shortcomings may threaten individual lives, businesses, industries, economies, and the even nations, but solutions are possible. By combining new techniques like advanced cryptography, distributed ledgers, and token economics, unsecured data can be secured, and all participants granted full control over their own communications.

Weaknesses in existing protocols and services

Despite being more decentralized than previous mass media technologies, today's web services are built using underlying protocols, services, and other tools that are susceptible to centralized control, surveillance, and manipulation. Fully decentralized applications will require several layers of alternative services without these weaknesses.

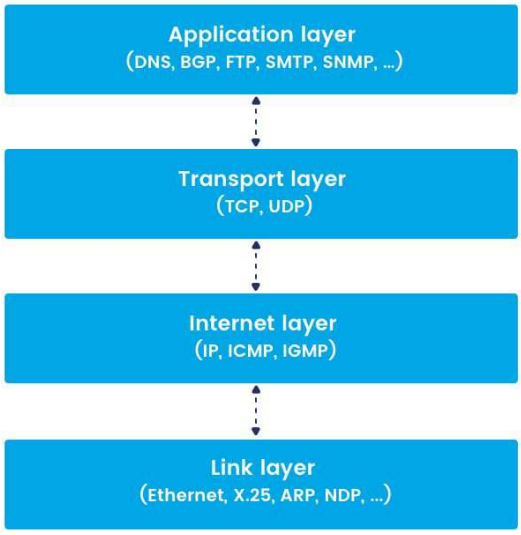
One of the first common ways that people used what eventually came to be called the Internet was to exchange data between universities and other government organizations. Similarly, the World Wide Web was originally built to share scientific documents. These technologies have since evolved far beyond these original use cases.

After the Internet was commercialized in 1995, new businesses began adapting these protocols for their own products and services, driving adoption but also becoming intermediaries between consumers and the underlying Internet protocols being used, capturing value at higher layers. As a result, the largest web properties today, such as Google and Facebook, are primarily in the advertising business, and have strong incentives to keep users in their own walled gardens. Having lost control over their own data, end-users are now locked into using these proprietary applications and services.

The lack of balanced incentives in the protocols increases the likelihood that businesses will monetize them in more exploitative ways. The protocols were also built without user privacy and data security in mind. They were invented in an environment where there was a high level of trust between network nodes, unlike today's Internet.

Technical challenges

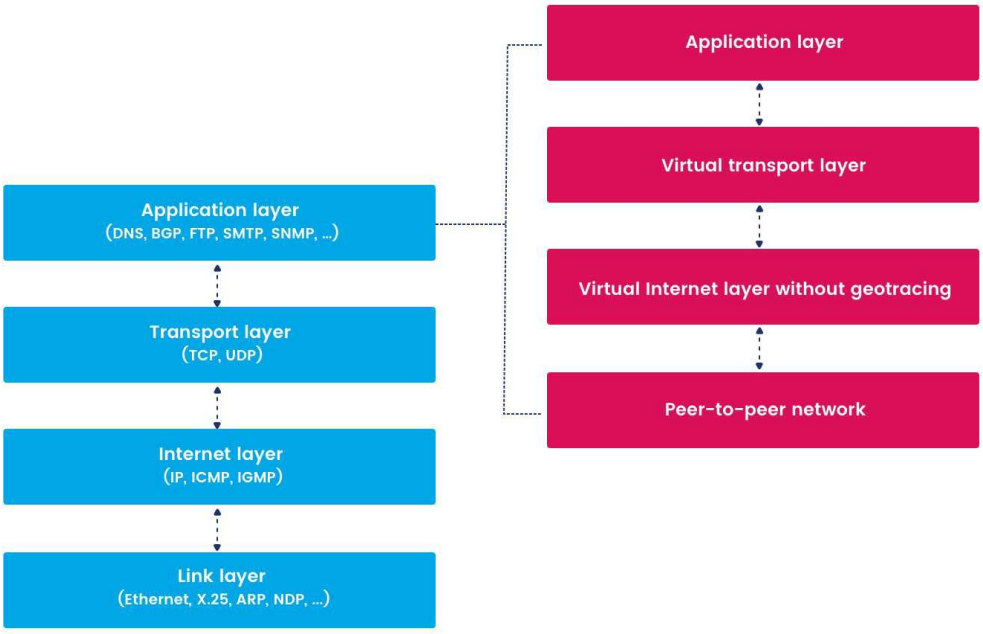
One difficulty in understanding the Internet's present technical limitations is that there are various weaknesses in multiple network layers, and therefore tackling them requires a multi-faceted approach based on a deep understanding of these various layers. A closer look at the Internet Protocol Suite,[15] the networking stack upon which the Internet is based, will help explain the difficulties. The following chart describes the protocol layers that data must go through to traverse the Internet. Each layer depends on the services and technologies in the layer below it.



The Internet Protocol Suite

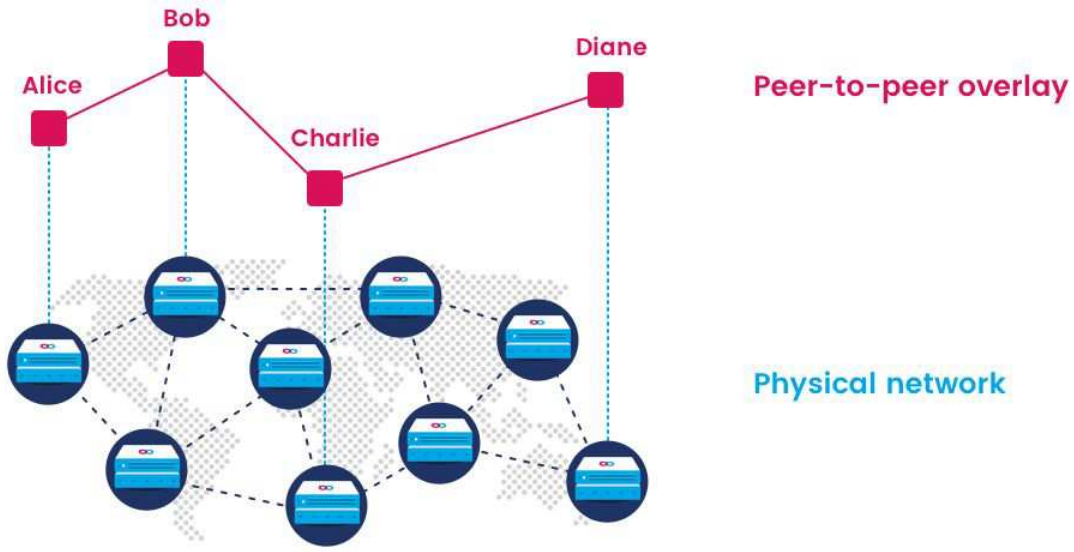
Addressing

In the context of the Internet layer, for example, IP addresses are used to route packets from one destination to another. All connected computers access the Internet through a public IP address or through a proxy with a public IP address. The geographic location of an IP address can usually be pinpointed with neighborhood-level accuracy. Each Internet service provider (ISP) is responsible for different ranges of IP addresses, allowing authorities to request information from them to discover the identities of Internet users fairly easily. Servers and/or content are routinely disconnected from the Internet by filing takedown notices against their ISPs.



Decentralized networking adds virtual abstraction layers above the Internet Protocol Suite

Fully decentralized addressing is achieved using peer-to-peer networks that create a layer of abstraction above which the geographical locations of nodes can't easily be determined, preventing them from being targeted.



Peer-to-peer networks create a virtual network overlay that hides the topology of the underlying physical network

Domain name resolution

The ability to associate memorable text with network addresses is another essential feature for practical Internet communications. Domain Name System[16] (DNS) is currently used for this function. Domain name registration is managed by ICANN,[17] a non-profit organization headquartered in California that is accountable to various international stakeholders. Services that rely on DNS are subject to disruption by powerful individuals and organizations who appeal to these stakeholders, or by governments and corporations that control infrastructure and tamper with DNS responses or publish alternative DNS records. If a jurisdiction wishes to suppress Internet-based information, it is a relatively trivial task to use DNS and/or takedown notices to make it inaccessible. Decentralized applications can avoid these barriers by providing a blockchain-based alternative name resolution protocol.

Certificate authorities

Related to DNS is the problem of Transport Layer Security[18] (TLS) and its reliance on the certificate authority (CA) system. Certificate authorities must be trusted to an unwarranted degree, are centralized points of failure, and are targets of attack. Even recent decentralized infrastructure has suffered from continued reliance on this system. EtherDelta, one of the largest decentralized cryptocurrency exchanges, was exploited by a breach of their DNS account.[19] Control of the domain was used as proof to a CA in order to have them generate a

certificate for that domain. Users returning to the site were then presented with a valid HTTPS certificate for the correct domain, and so were not protected by the security checks in their browser when they were directed to a malicious web server. Some users of the site were reported to have lost significant amounts of cryptocurrency in the attack.

Numerous other trusted organizations have been subject to phishing attacks. In such attacks, malicious actors generate valid certificates that are signed by trusted certificate authorities for similar-seeming domains from which they serve trojan source code. That such actors succeed in obtaining valid certificates from trusted certificate authorities demonstrates the lack of security in the present system. Research is currently underway on decentralized identity and reputation systems that can deliver higher levels of authenticity assurance than the certificate authority system.

Mail protocols (SMTP/IMAP)

Conventional email over Simple Mail Transfer Protocol[20] (SMTP) and Internet Messaging Access Protocol[21] (IMAP) achieve some degree of decentralization in the sense that anyone can set up a mail server and send email. However, they were invented in an era and at a scale of Internet growth in which all nodes were trusted, and had no built-in security mechanisms. This lack of security has allowed spam to proliferate, since the cost of sending unsolicited messages is negligible. While extensions, such as DomainKeys Identified Mail[22] (DKIM), can help to reduce spam, SMTP and IMAP also rely on DNS and conventional client-server architecture, and are therefore not fully decentralized. They also lack the real-time responsiveness that users have come to expect from messaging apps, like Slack or WhatsApp. Finally, SMTP is not end-to-end encrypted by default, and encryption solutions are not easy enough for non-technical users.

Group communication protocols (IRC/XMPP)

Internet Relay Chat[23] (IRC) and EXtensible Messaging and Presence Protocol[24] (XMPP) are some of the more popular first-generation protocols for Internet-based group communications. While these protocols allow for ad-hoc server distribution, they still rely on DNS and conventional client-server architecture, and are not fully decentralized.

Hosted email

Even though it is possible for people to host their own email servers and use PGP[25] for encryption, very few people do, relying on hosted email platforms like Gmail, iCloud, Outlook, Yahoo Mail and others. In addition to having all the weaknesses of SMTP, these services add dependence on third-party infrastructure, lack of control over one's own data, and the need for an unreasonable degree of trust in the organizations providing the services. On services supporting custom domains,

such as GSuite, there is an added requirement of trust in the domain managers, who can archive and surveil the communications of all domain users.

While data retention may have some benefits for certain corporate users, it also carries additional burdens. Companies that retain data may be forced to use it against their own and their customers' interests, as in the 2016 incident when Apple was asked to unlock an iPhone for the United States Federal Bureau of Investigation (FBI)[26]. Although this particular request was seen by some to be in the public interest, it caused concern for many other Apple customers who became worried that their privacy and civil liberties could be threatened if such interventions became commonplace. Apple defended these interests, claiming that backdoors in their products could easily be used for malicious purposes, and that the FBI request constituted government overreach. Other governmental organizations have made similar requests for security backdoors in popular messaging apps, like WhatsApp[27] and Skype.[28] Limiting access to customer data minimizes corporate liability.

Chat

Many third-party services now provide organizations with a convenient place to communicate with all their colleagues in real-time, with richer formatting, file sharing capabilities and telepresence features. Most notable among these is Slack, which boasts over six million users.[29] While customers are attracted to their convenience and rich feature set, their data is visible to the service provider, whom they must trust to safeguard it and keep it up and running smoothly. For example, on January 9, 2018, Slack had an outage, disrupting the workdays of millions of users. The outage became the number one trending topic on Twitter as customers took to the social network to share their frustration.[30] Fully-decentralized messaging eliminates these points of failure.

Encrypted peer-to-peer/group chat

Recent product offerings from Keybase, Signal, and others provide fully encrypted communications between individuals and even groups in some cases, but they rely on centralized infrastructure to operate or store customer data, thus exposing them to data loss and disruption of service. While these solutions achieve better user sovereignty than some, they still require significant compromises, and expose customers to unnecessary risks.

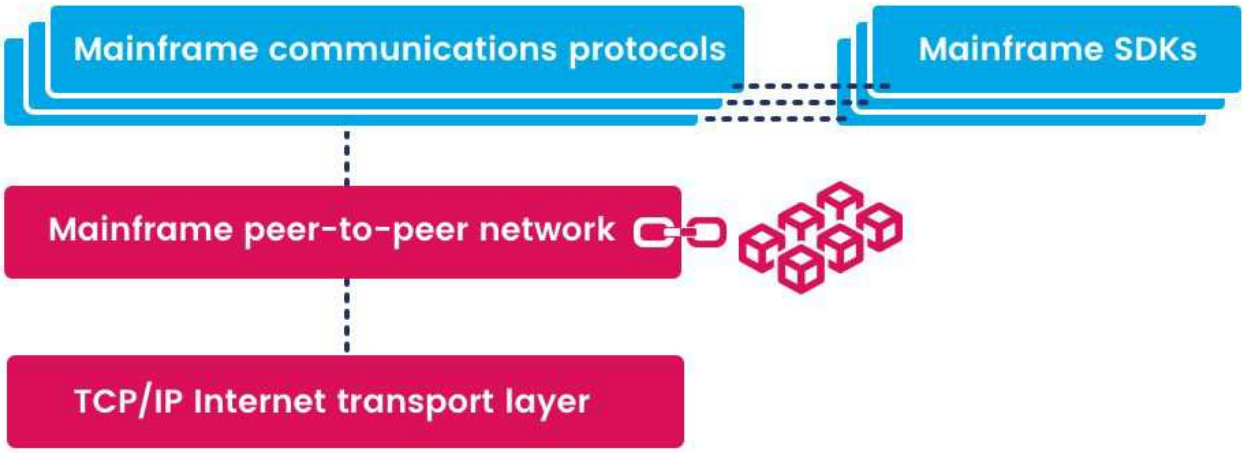
Distributed private chat

Some organizations, such as Matrix, are beginning to provide enhanced user sovereignty by allowing users to host their communications on customer-rented cloud infrastructure rather than third-party infrastructure. We believe this is a step in the right direction, but only the beginning. Fully decentralized messaging requires unhosted architecture, where the entire application infrastructure runs on incentivized peer-to-peer protocols.

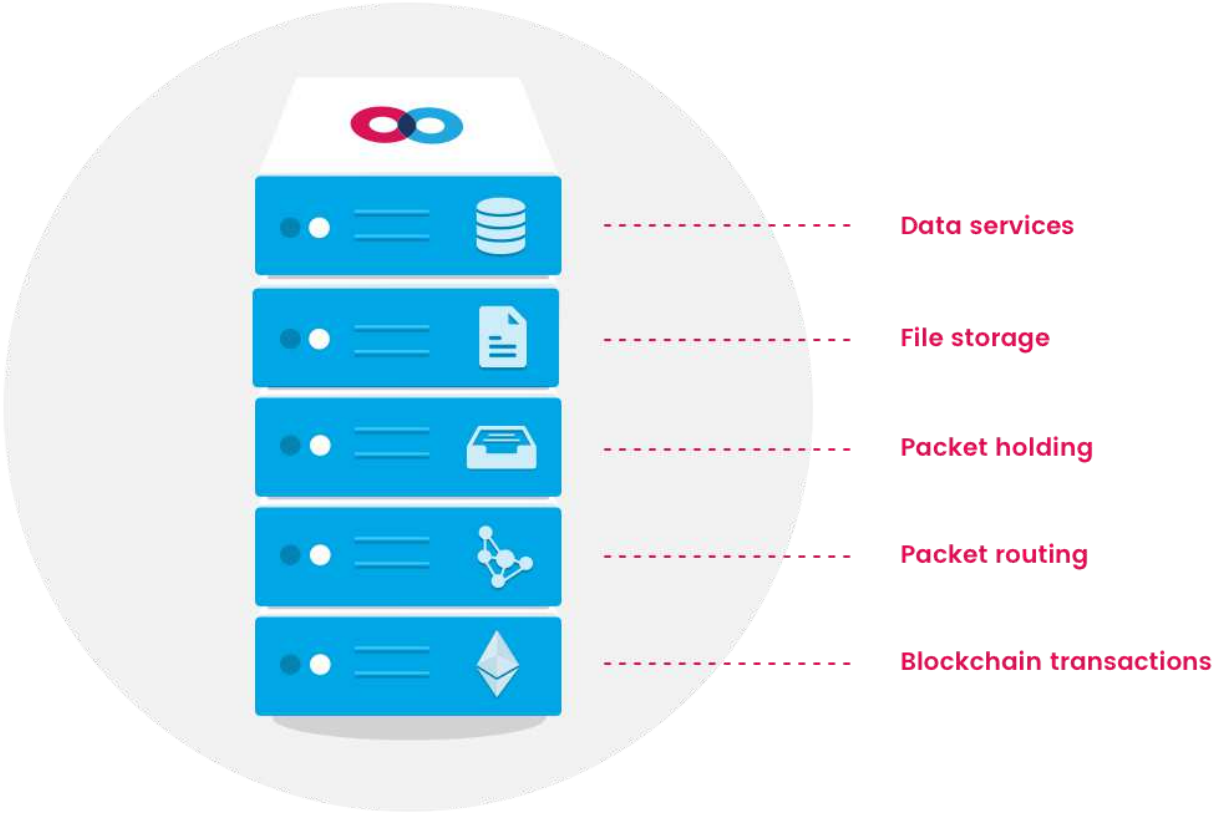
An unstoppable communications platform

Mainframe is an unstoppable communications platform that combines the desired features of today's best networking protocols and applications while also maintaining the highest level of security and user sovereignty. The platform consists of various protocol and transport layers incentivized by token economies, along with software development kits (SDKs) enabling easy integration with a variety of popular languages, operating systems, and devices, as well as smart contracts and oracles for token exchange and ease of interoperability with various popular blockchains.

At its lowest levels, Mainframe consists of a Kademlia-based[31] peer-to-peer network used to propagate and execute blockchain transactions. This network layer abstracts away the underlying geographically-traceable Internet transport layer it relies on, randomly assigning addresses to each peer or node. Monetary value can be exchanged between nodes on this network using blockchain tokens. Mainframe provides additional protocols above this transport layer for secure communications.



Each Mainframe node exposes various p2p service interfaces for application layers to call, including interfaces for blockchain transactions, packet routing, packet holding, file storage, and data services. Each of these p2p services is provided entirely by peers operating in incentivized cooperation with one another, without reliance on any managed infrastructure.

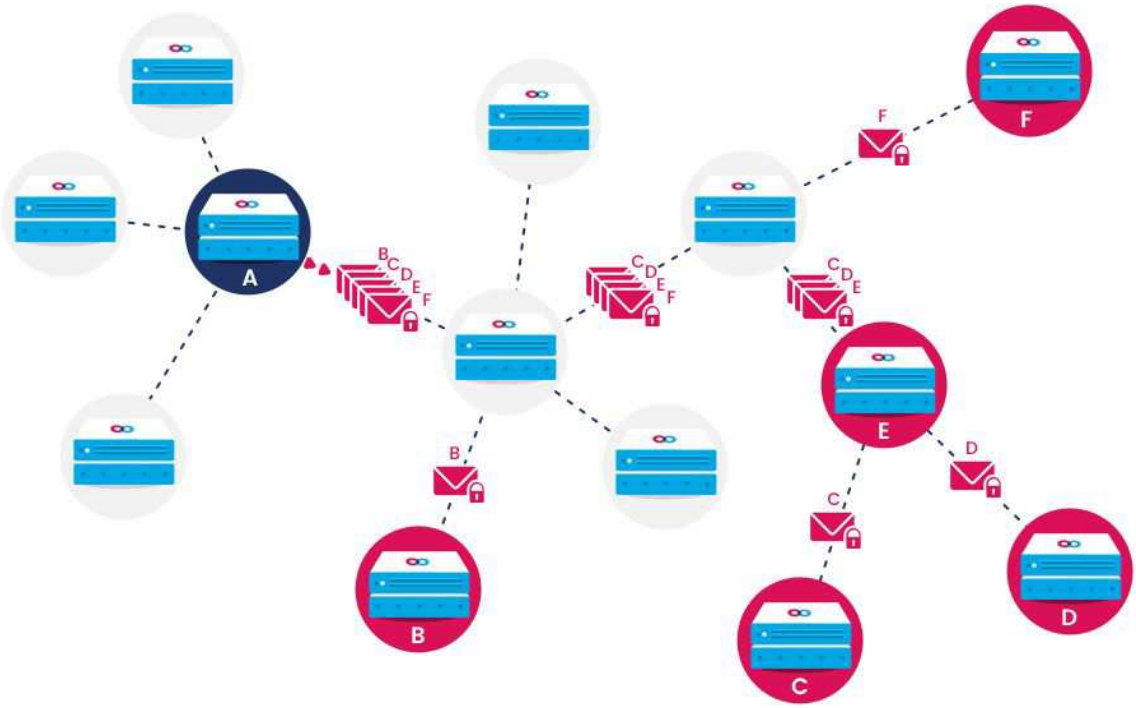


Mainframe node
Conceptual diagram

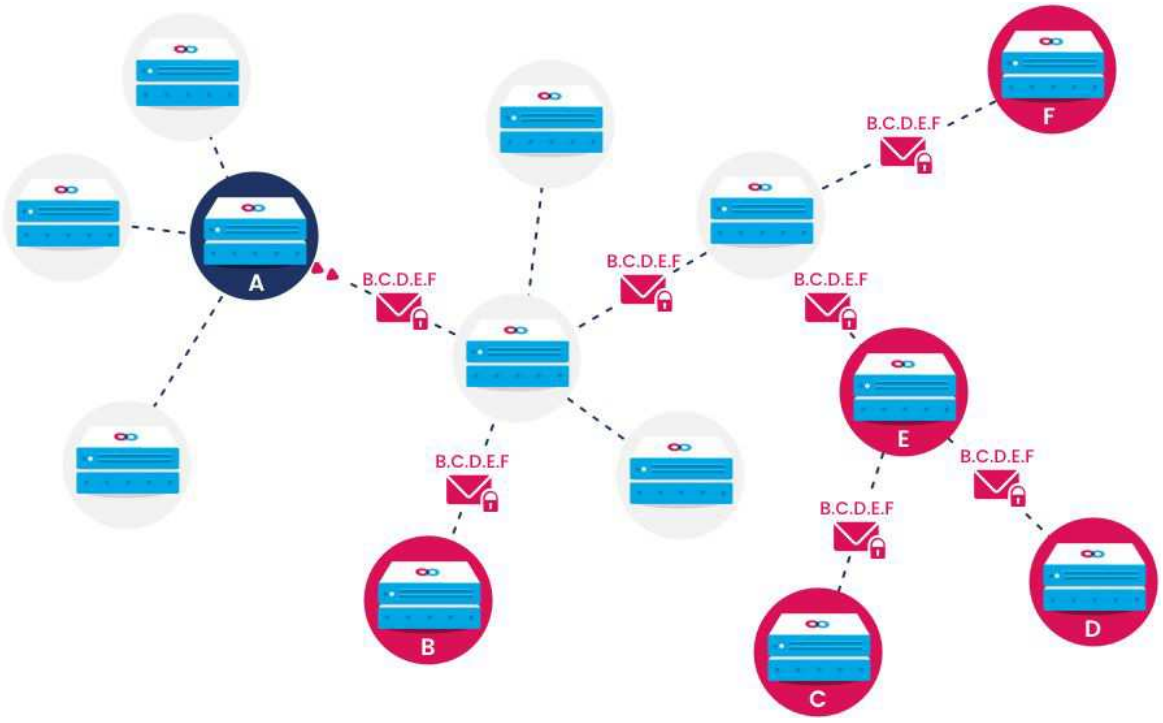
Encryption

Mainframe provides protocols for one-to-one and one-to-many encryption. Each Mainframe node has an asymmetric key pair associated with it. This key is used to decrypt packets intended for this node. When one node wishes to send a packet to another, it encrypts the packet using the public key of the intended recipient node, or a pre-arranged shared key. Forward secrecy is ensured by additional ephemeral symmetric keys. The receiving node's public key is obtained from a prior exchange of contact information that occurs out-of-band, such as by consulting a directory of contacts, or from a direct exchange of public keys between individuals. Packet encryption is an integral part of Mainframe's transport protocols and cannot be circumvented.

Packets intended for multiple nodes can be sent in multicast mode. This allows the sender and nodes routing multicast packets to send a single packet instead of duplicates along any route that will reach two or more of the intended recipients. Mainframe provides protocols for shared key negotiation so that multicast packets can be encrypted only once for multiple recipient nodes. This mode of operation is intended for high-performance applications requiring moderate security, as multiple destination addresses are revealed in packet metadata. It can also be combined with dark routing (explained below).



A sends a packet to B,C,D,E,F nodes WITHOUT multicast

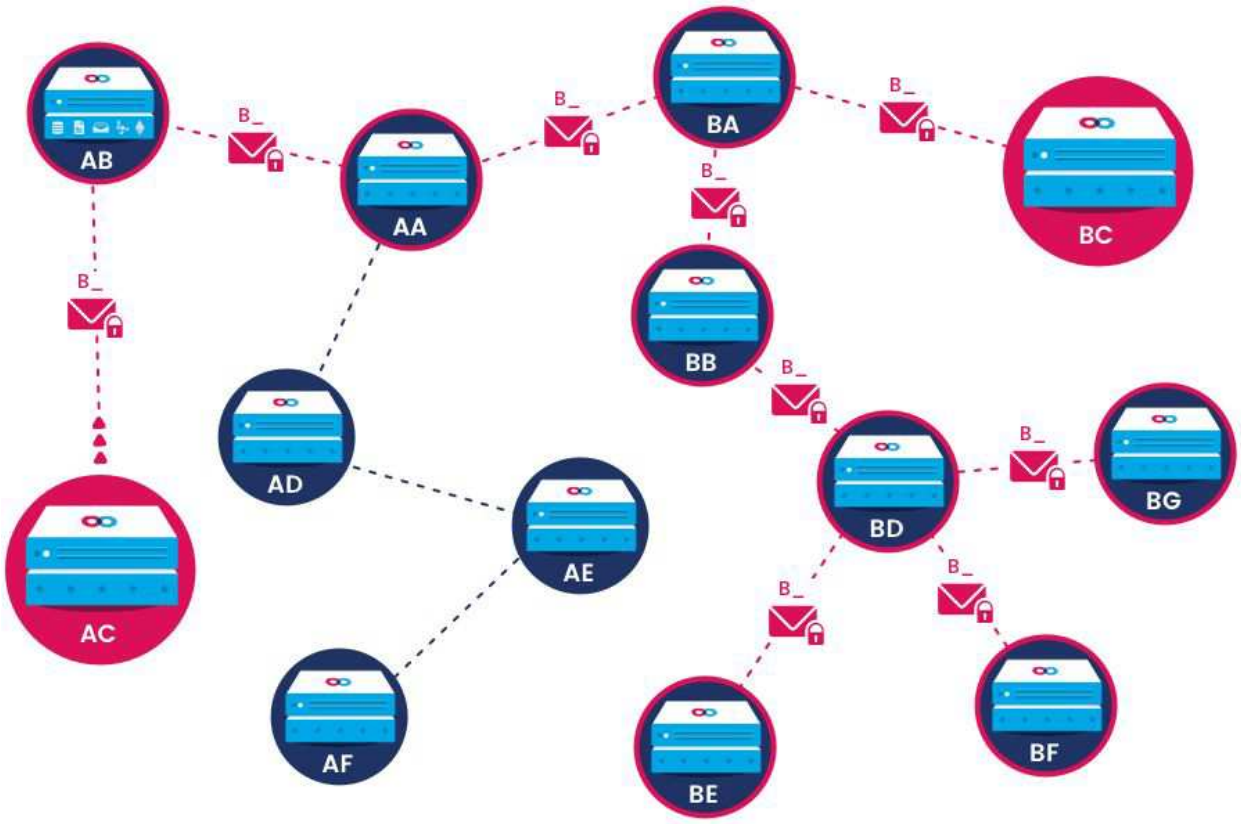


A sends a packet to B,C,D,E,F nodes WITH multicast

Multicast mode allows identical information destined for a group of nodes to be encrypted once and sent across the network without duplicating packets

Dark routing

Despite their use of encryption, conventional networking systems still allow malicious actors to glean information about who is communicating with whom. Mainframe allows nodes to prevent this using configurable dark routing based on Holbrook's work on the PSS protocol[32] which is itself an evolution of the Whisper protocol, originally conceived by Wood.[33] In dark routing mode, packets are routed to each node whose address matches the partially disclosed destination address, moving efficiently towards this subset of nodes, but after that they are distributed to all matching nodes, making it intractable for any network observer to infer the intended recipient within that zone.



Node AC sends a packet to BC with dark routing (partial addressing).
The packet gets routed to all nodes with matching addresses, but can only be decrypted by BC.

Nodes are responsible for selecting the appropriate level of *luminosity* (address specificity) when addressing packets. Too much luminosity increases the possibility that malicious actors could identify patterns of communication between nodes, while too little increases congestion and transmission costs. Mainframe provides algorithms for determining sensible luminosity settings based on network conditions and privacy requirements for different use cases.

Because packets do not need to be fully addressed, session management becomes more challenging. Nodes must have another means of identifying packets they are interested in viewing. This is done using a previously agreed-upon topic ID inside a packet. Mainframe provides session management protocols that help applications keep track of and initiate separate data streams. When a new session is negotiated between nodes or groups of nodes, a topic ID is generated and shared privately with all session participants, who then listen for future packets with the same topic ID.

Even if nodes select a topic ID that is already being used by other nodes, they will only be able to decrypt packets that have been encrypted with a known secret key, and will ignore other packets having the same topic ID, so topic IDs help nodes reduce processing overload for irrelevant packets but gracefully handle collisions. When multiple, disparate communication streams use the same topic ID, it also helps to make it more difficult for would-be eavesdroppers to recognize communication patterns.

Because communication patterns can't easily be discerned in this mode of operation, the network is highly resistant to surveillance attempts, as well as attempts to target specific nodes for denial of service. The combination of encryption with dark routing enables a truly unprecedented level of security.

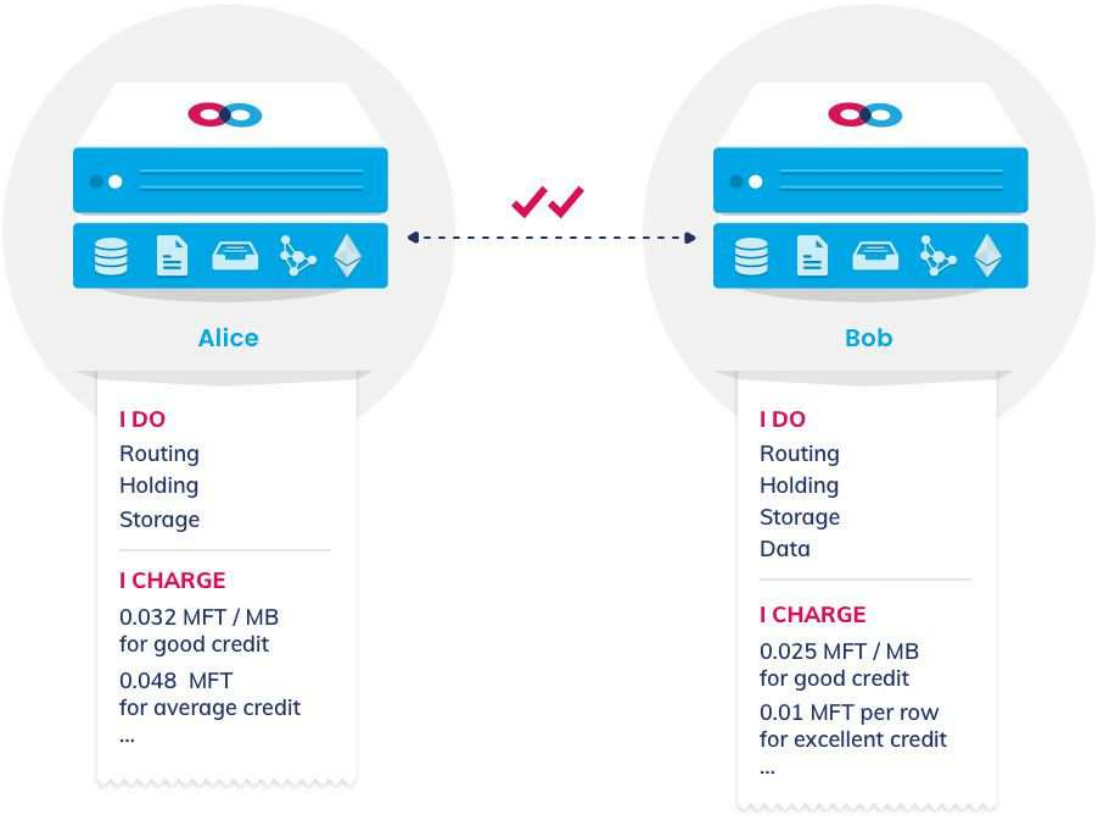
Incentivization

Mainframe communications make use of, and build upon, a generalized model for incentivized service provision called *swap, swear, and swindle*, as described by Trón and Fischer.[34] This model is used to incentivize various important services provided by nodes in the network.

Incentivized packet routing

Peer nodes enter into an agreement to provide services to one another; in this case, the reliable receipt and delivery of packets. They keep track of how much bandwidth they've sent to and received from each other using the SWAP protocol (Swarm Accounting Protocol for Service Wanted and Provided).

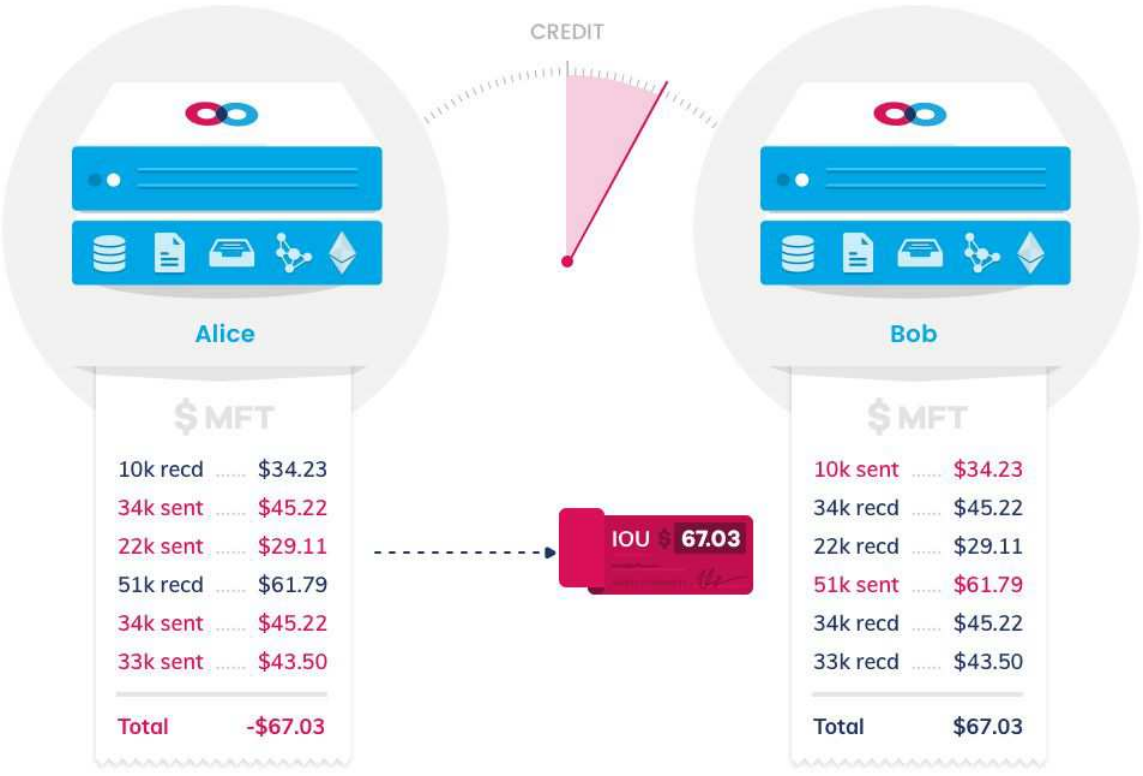
Each participating node in the SWAP protocol uses a checkbook contract to manage the service accounting process, depositing funds into the contract in the form of ERC20 tokens. Mainframe nodes transact in Mainframe tokens (ticker symbol: MFT). Nodes publish their services and prices using a service discovery protocol. By allowing nodes to agree upon services, pricing, and a medium of exchange, the impartiality and interoperability of the network is improved, and a wider array of stakeholders have stronger incentives to participate. If potential peers can agree upon acceptable service prices, they enter into a peering relationship.



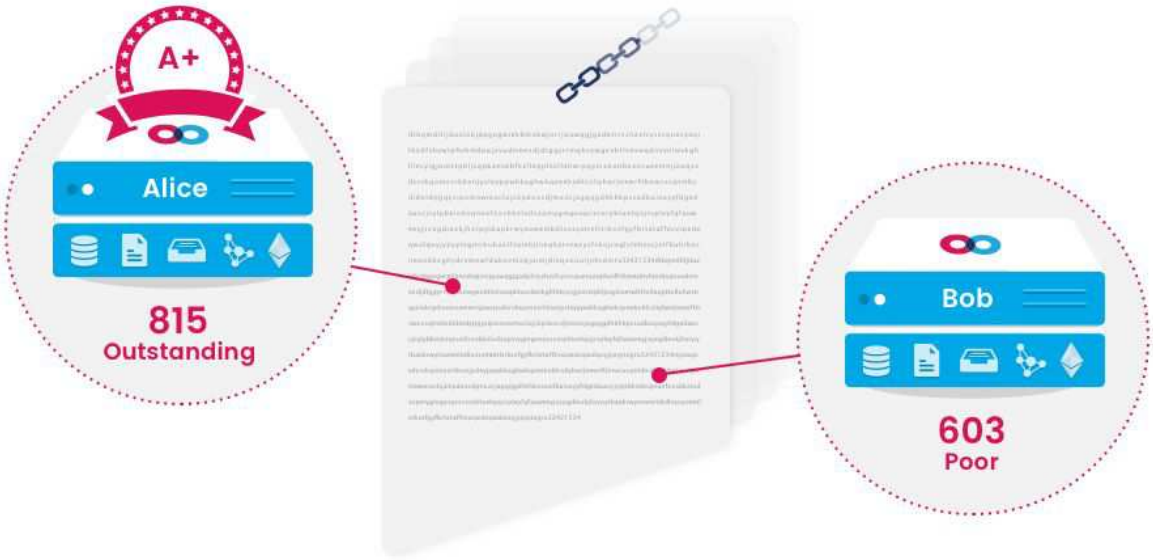
Mainframe's service discovery protocol
Used to determine each node's services and rates

Periodically, when the amount of bandwidth provided by a node's peer exceeds a certain threshold, the peer issues an "invoice" requesting additional funds. The node issues a "check" to the peer in return by cryptographically signing a piece of data that can be used as an input to the checkbook contract. A node may cash the check immediately, or it may hold the check under the assumption that it will eventually owe money in return. If this occurs, it may simply void the check to "pay back" its peer, or it could issue another check. Each node has a configurable threshold for how long it will wait to accumulate checks before it attempts to settle its accounts. Checks are exchanged off-chain. This increases the risk of default, but allows the participants to avoid frequent costly transactions.

Checks also may or may not be backed by sufficient funds. If a node attempts to cash a check for tokens and there are insufficient funds in the checkbook contract, the node incurs a loss, but bounced checks also damage the reputation of the sender and will eventually prevent it from participating on the network. If a node doesn't send a check within its peer's declared deadline or bounces a check, these actions are visible in the blockchain ledger for all other nodes to see.

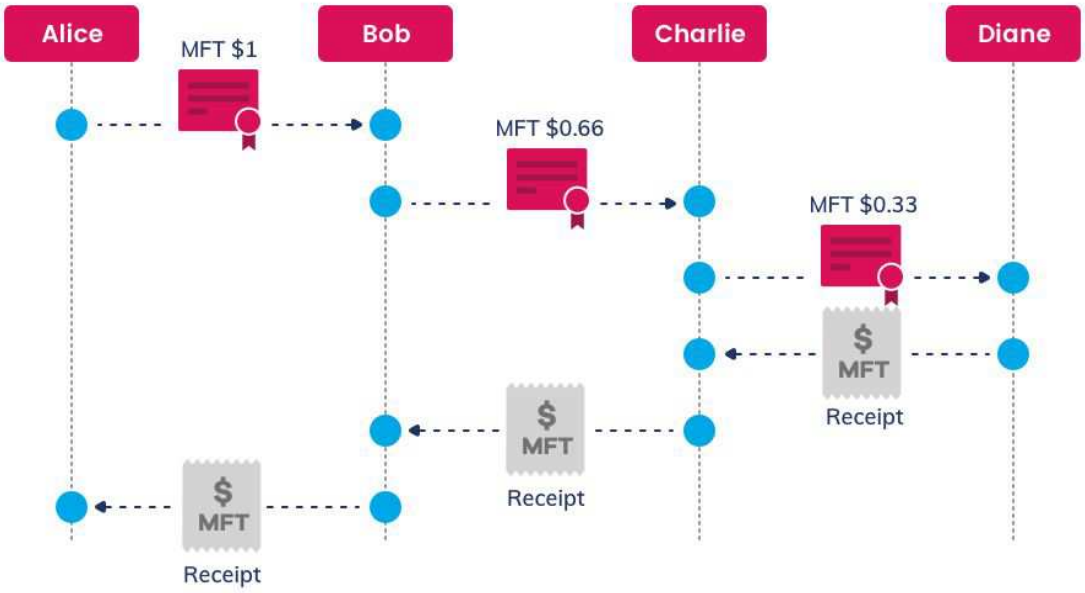


Nodes may retrieve the credit scores of other nodes using the Mainframe credit scoring API, factoring these scores in with additional criteria of their own, if desired. Nodes with little or no history have a middling score that increases as they pay their bills punctually and fund their checks reliably, or decreases as they fail to pay bills and default on payments. Each node's service discovery manifest advertises the prices it charges for different credit score thresholds, and a node may even choose to disconnect altogether from a peer with a credit score that is too low.



Incentivized packet delivery

In tandem with the direct economic relationship between peers, Mainframe incentivizes packet delivery across multiple peer-to-peer hops in the network with the concept of cryptographically certified delivery. This mode of operation allows a sending node to issue a conditional bond specifying an escrow condition that some data be obtained from the receiving node proving that a packet was delivered. Performance and reliability can be balanced by only verifying some fraction of packets this way.



Alice sends a packet to Diane with a bond for certified delivery.
Each node collects a portion of the bond, sends another one on,
And returns a receipt proving delivery to its peer.

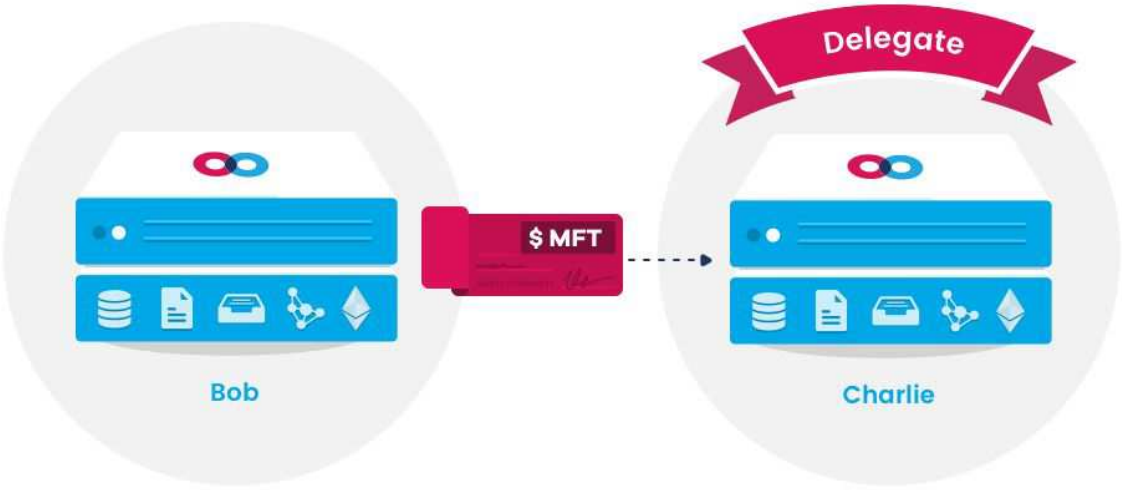
Each node can show the receipt from its neighbor as *takeover proof*, or proof that the packet was successfully delivered to its neighbor, challenging that neighbor to show similar proof until delivery of the packet is confirmed. Nodes that cannot show proof of delivery must either prove that the receiver was offline and that they are likely to be nearest neighbors of the receiver using their Kademlia addresses, or they can be blamed for failed delivery and forfeit staked tokens. This process is called *finger pointing*,[35] and essentially ensures guaranteed delivery.

In dark routing mode, rather than verifying delivery to all matching addresses, at each hop in the dark route, a bond is only sent to a randomly selected fraction of the neighbors, thus maintaining the anonymity of the receiver while providing a reasonable incentive for relays to deliver packets reliably. Because of its promiscuous duplication of packets, incentivized delivery in dark routing mode costs more. Network inference allows senders to get an estimate of how much packets will cost to send, depending on their luminosity settings and the configuration of their neighbor selection algorithm.

Incentivized packet holding

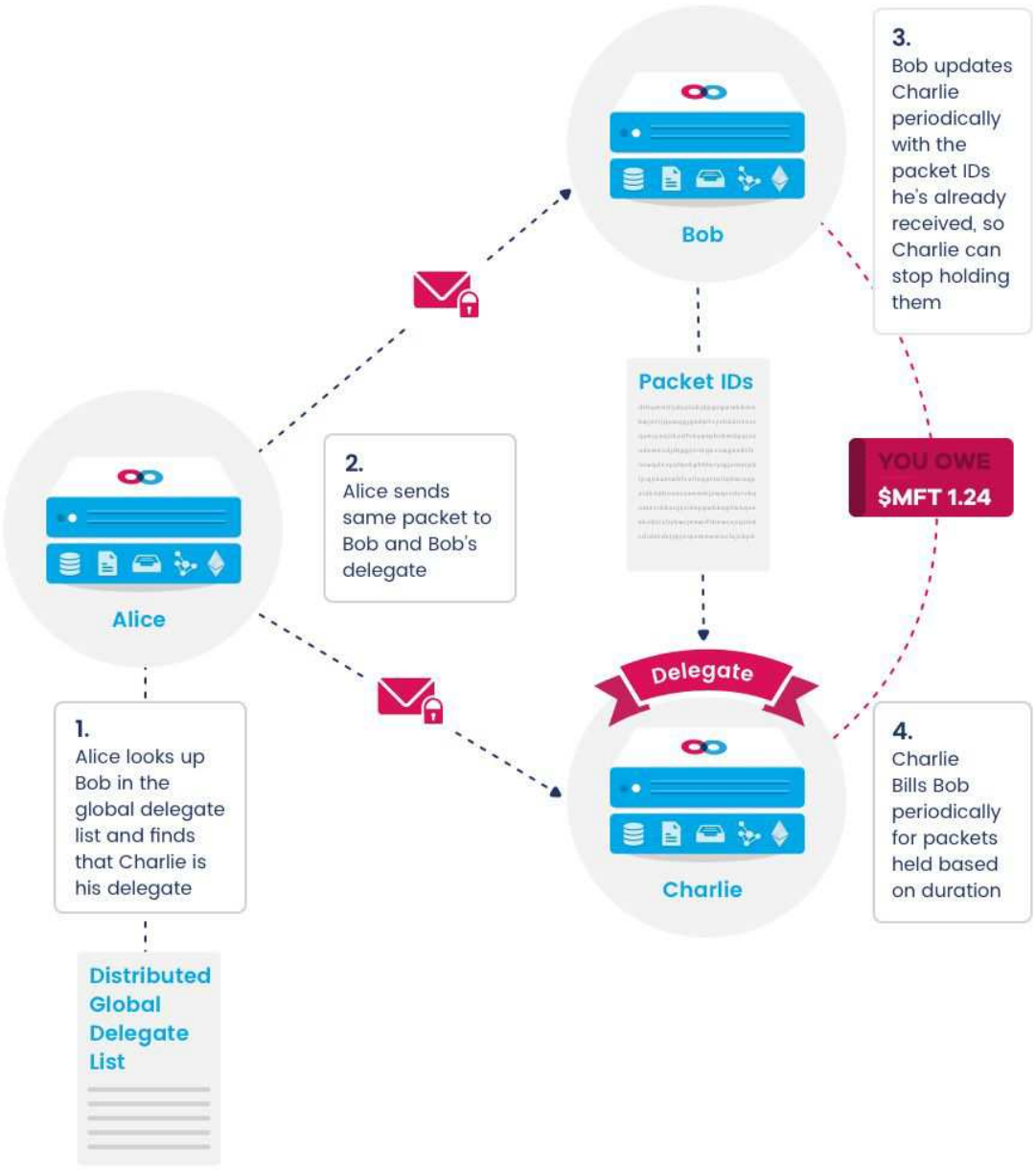
Mainframe communications must continue to function even when some nodes are disconnected from the network. Since Mainframe infrastructure is entirely based on services provided by nodes in its underlying peer-to-peer network, some use cases require that packets intended for a node that is offline be held in reserve by other nodes until it goes back online. This service is incentivized.

Nodes may advertise packet holding in their service discovery manifest. Other nodes requiring packet holding execute a service discovery query to find nodes offering packet holding for acceptable prices. Once they find acceptable packet holding delegates, they enter into a service agreement with these nodes and use the data services interface to update their packet holding delegate list in the global packet holders table. At this time, the delegate also stakes a certain number of tokens against the possibility that it may fail to hold packets according to its agreement.



Nodes may pay delegates to hold packets for them while they are offline

Nodes wishing to send packets to other nodes must first check the global packet holders table to see if the node they wish to send packets to has a delegate. If this is the case, the sender must send packets to all the receiver's delegates in addition to the receiver. The receiver periodically updates its delegates with a signed list of the packet IDs it has recently received. Delegates may then delete these from their queue of held packets, charging nodes for the time they were held. Nodes that have been offline for a while may also request held packets from their delegates.

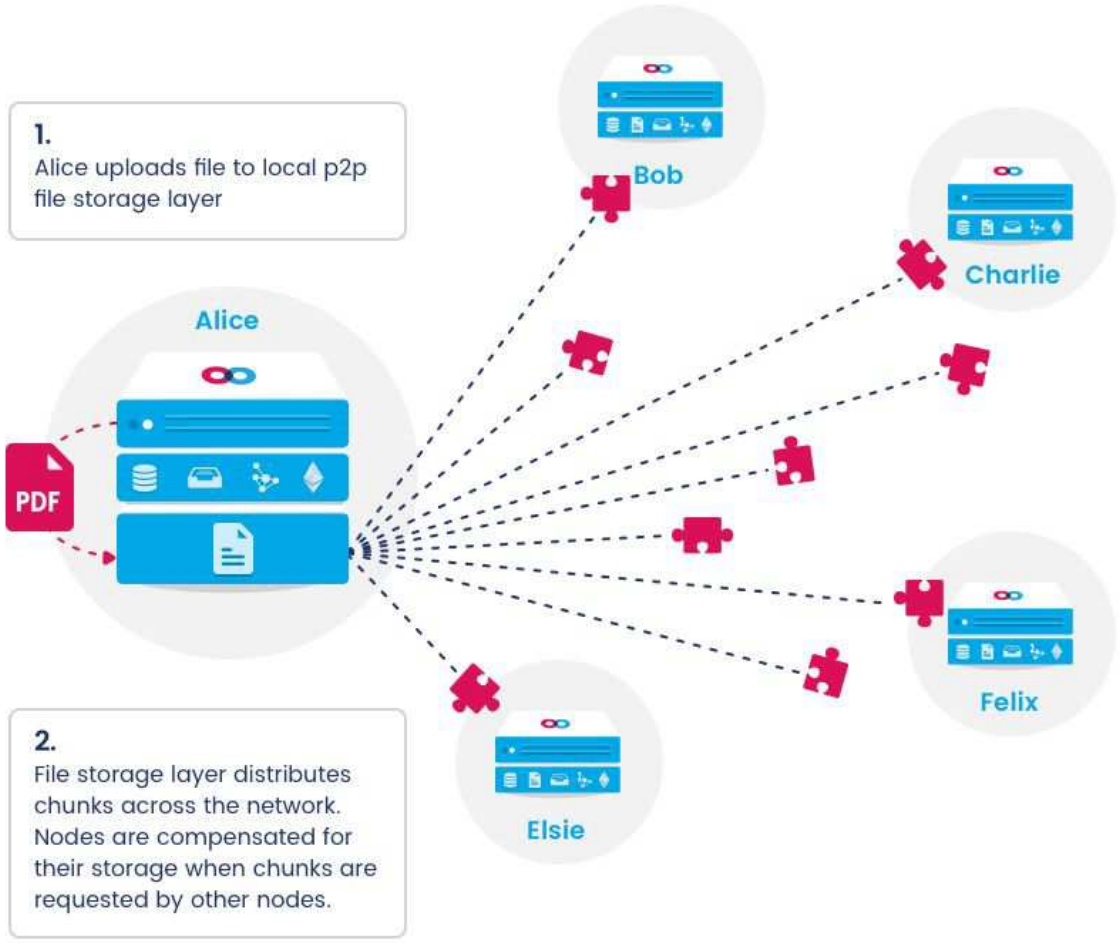


Delegating a packet holder ensures that packets are held while nodes are offline for later retrieval

If a delegate is suspected of violating its service agreement, it can be challenged by opening a case with the *swindle* contract. The contract behaves as a state-machine conducting a court proceeding; witnesses are called and a guilty/not-guilty verdict is reached based on their testimony. Delegates that are proven guilty forfeit the funds they previously deposited into the *swear* contract. The swindle verdict is recorded on-chain, and can be used to quantify a node's trustworthiness.

Incentivized file storage

Mainframe's incentivized file storage layer makes use of the *swear* and *swindle* stages of service provision, in addition to the swap stage used in packet routing. Nodes attest (swear) that they are storing random portions of a file (chunks) by staking tokens on the possibility that they might fail a storage test. Then they can be queried for a cryptographic proof of storage for any particular chunk periodically. If they are unable to return valid chunks, a swindle proceeding may be made against them, causing them to lose their staked tokens if proven guilty. Storage nodes are incentivized by earning tokens when file chunks are requested by other nodes.[36]

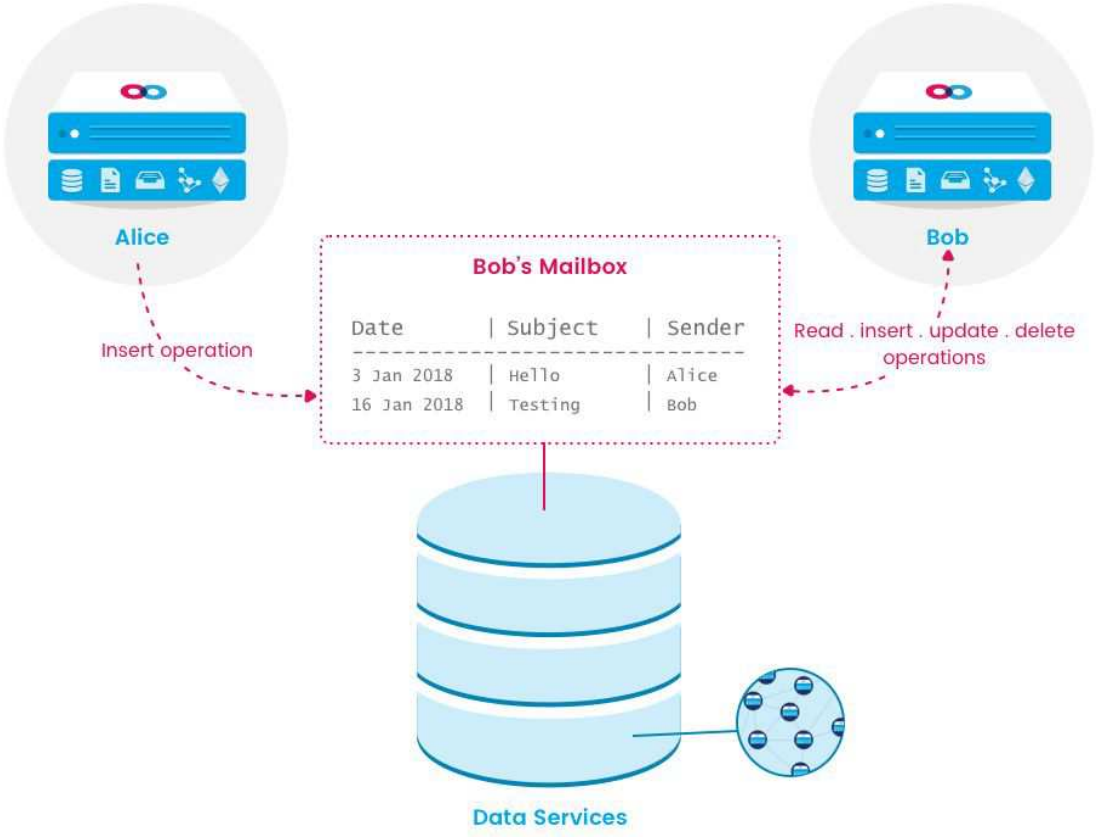


Incentivized data services

While file storage supports many use cases, it is inefficient to have to load and save a large file whenever small changes are made to it. For these kinds of use cases, Mainframe provides incentivized database services.

Nodes wishing to earn tokens for providing these services advertise their availability using the service discovery protocol. Other nodes may store indexed datasets with them for subsequent retrieval and querying. These datasets are stored with a user-specified level of redundancy. The greater the redundancy, the more compensation a node can earn in the unlikely event that its data is lost. Reliable service is incentivized using swindle witnesses, whose testimony may result in the confiscation of providers' staked tokens if they fail to provide requested data.[37]

Dataset owners can specify who is allowed to read, write, update and delete rows from the set. This fine-grained access control enables many useful application features. For example, mailbox functionality could be implemented by allowing any user (or, alternatively, any previously-authorized user) to insert rows into a dataset representing another user's mailbox, but only the mailbox owner could read, update or delete rows from the dataset. Users with read privileges can subscribe to receive notifications of any changes to the dataset.



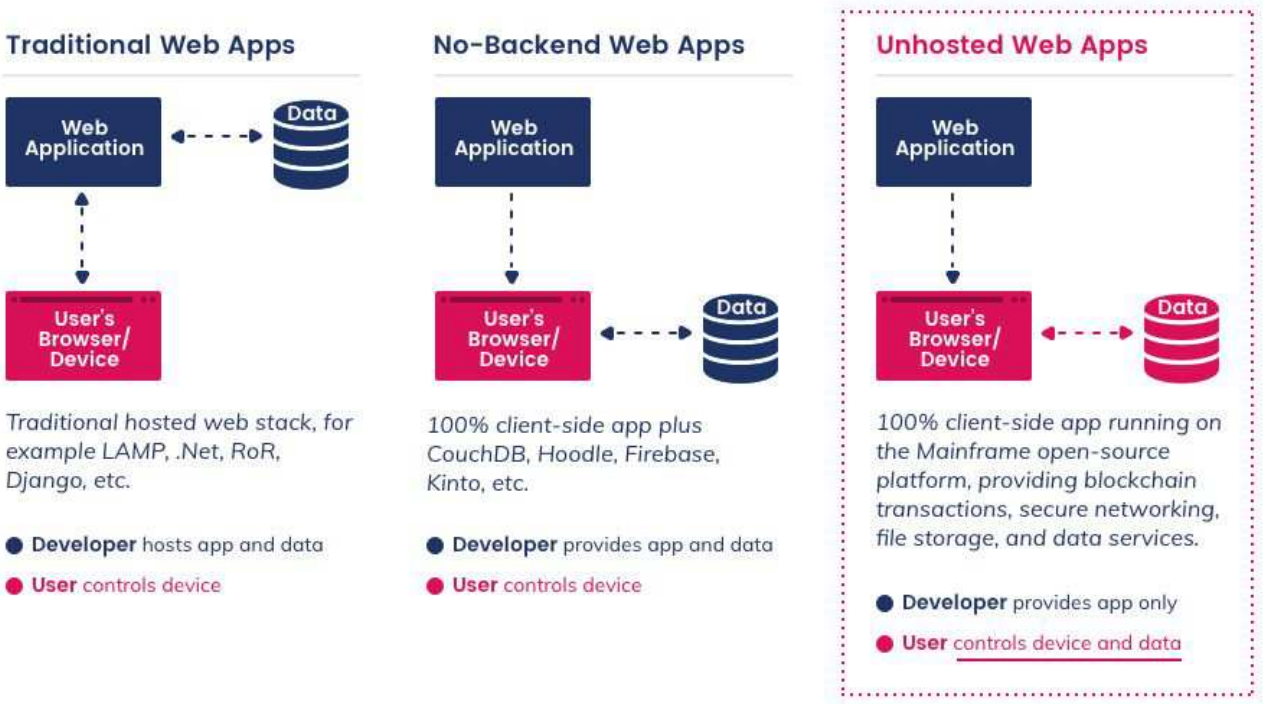
Example application using Mainframe data services. Alice can write new messages to the table representing Bob's mailbox, but she cannot read or alter the messages in the mailbox. As the mailbox owner, Bob can read and modify the messages. The Mainframe data services layer ensures that multiple copies of these datasets are stored across the network for high availability and redundancy.

Unhosted architecture

The two aspects that ensure the resilience of the Mainframe network are that it is both distributed and decentralized. Distributed networks are relatively commonplace today, mostly seen in the form of CDNs (Content Delivery Networks), where they provide highly-available access to online content. This type of architecture allows the network to seamlessly survive loss of or attacks against any subset of the nodes. The distributed nature of the Mainframe network means as participants in the network join and leave, the network has the ability to grow and heal around them. As the network scales up, the number of possible routing paths between any two nodes increases, dissipating potential hot-spots and further diminishing the overall effect nodes dropping out will have on the health of the network. The Kademlia network protocol will ensure that nodes try to discover new pathways when a peer is dropped in order to maintain an optimal overlay network.

Decentralized networks by definition have no central authority that can exert control over the flow of data across them. This makes them exceedingly difficult to shut down by a potential adversary, as there is no single point of failure or attack surface. Bittorrent operates this type of network, and while court orders were used to shut down websites advertising torrent content, the underlying file-sharing network could not be stopped effectively. The Mainframe network operates in a similar mode and is not reliant on any specific set of nodes being present or connectable at any time. No entity (even Mainframe itself) will be able to control or disrupt the operation of the network.

In addition to this resilient network structure, the various service layers provided by the network must be architected in a way that reduces reliance on specific nodes as much as possible. For this reason, services have been designed with redundancy in mind, storing individual shards of data across multiple nodes and designating multiple nodes to perform requested services. The network service layers have been designed to support the development of truly “unhosted,” or fully decentralized applications.[38] This means that developers on our platform are not required to provision or manage their own infrastructure, as would be required for a traditional web service, and consumers do not need to trust third parties to maintain such infrastructure. Mainframe will continue to develop services to better meet the needs of fully decentralized applications as this space matures.



Interoperability

Mainframe is designed to make it easy for any application to make use of our platform for its communication needs. We will create developer SDKs for working with Mainframe code on all popular platforms and languages, in order of developer demand. We will also provide oracles and smart contracts for interacting with the token-based incentivization layers on multiple blockchains, including Ethereum, NEO, RSK and Tezos.

In addition to these SDKs, Mainframe will strive to accelerate adoption by focusing on developers, providing detailed tutorials and documentation, providing and supporting forums where developers are discussing and working on Mainframe-based applications, and providing paid services for support and consulting.

Token Utility

Mainframe tokens have various uses within the Mainframe ecosystem:

- They incentivize the timely and efficient relay of packets between peers;
- They incentivize the delivery of packets from sender to receiver;
- They incentivize reliable decentralized file storage;
- They incentivize reliable decentralized data services;
- They can be used as a medium of exchange for marketplaces on the Mainframe platform where digital goods and services can be bought and sold.

Use Cases

The Mainframe platform supports a wide variety of products and services. The first application built on the platform is a messaging application that takes full advantage of the provided decentralized protocols. An alpha version of this application, called Onyx, was released on December 19, 2017.

Onyx messaging app

Onyx, a messaging app built on the Mainframe platform, provides a first-class user experience on all major mobile and desktop platforms, including Windows, MacOS, Linux, iOS and Android. It is the product of years of experience creating tools for real-time corporate messaging. The app is optimized to make minimal use of system resources. It provides both peer-to-peer and group chat, tracking the context that your messages take place in and allowing you to quickly navigate between all your conversations.

Configurable dark routing

Onyx is the first application that allows the user to take advantage of dark routing. Users can configure the luminosity settings for all messages sent in group and private channels.

Rich content and microformats

Onyx allows typical rich formatting features, as well as file attachments and inline viewer widgets for images and videos. In addition to these typical features, Onyx has numerous microformats for displaying rich interactivity beyond simple messaging apps, including polls, task assignments, group checklists, and others.

Full-text search

Onyx caches contact lists and message content locally for fast retrieval of archived messages and full-text search through message content and contact directories.

Directory services

Onyx provides decentralized contact storage, editing, and sharing. Contact information that a user wishes to make public is addressed using a decentralized name service, such as ENS,[39] and stored in Mainframe decentralized storage. Contact information that individuals and organizations wish to share privately is shared using identity claims over networks like uPort[40] or the Sovrin ledger.[41]

Signaling for presence and activity

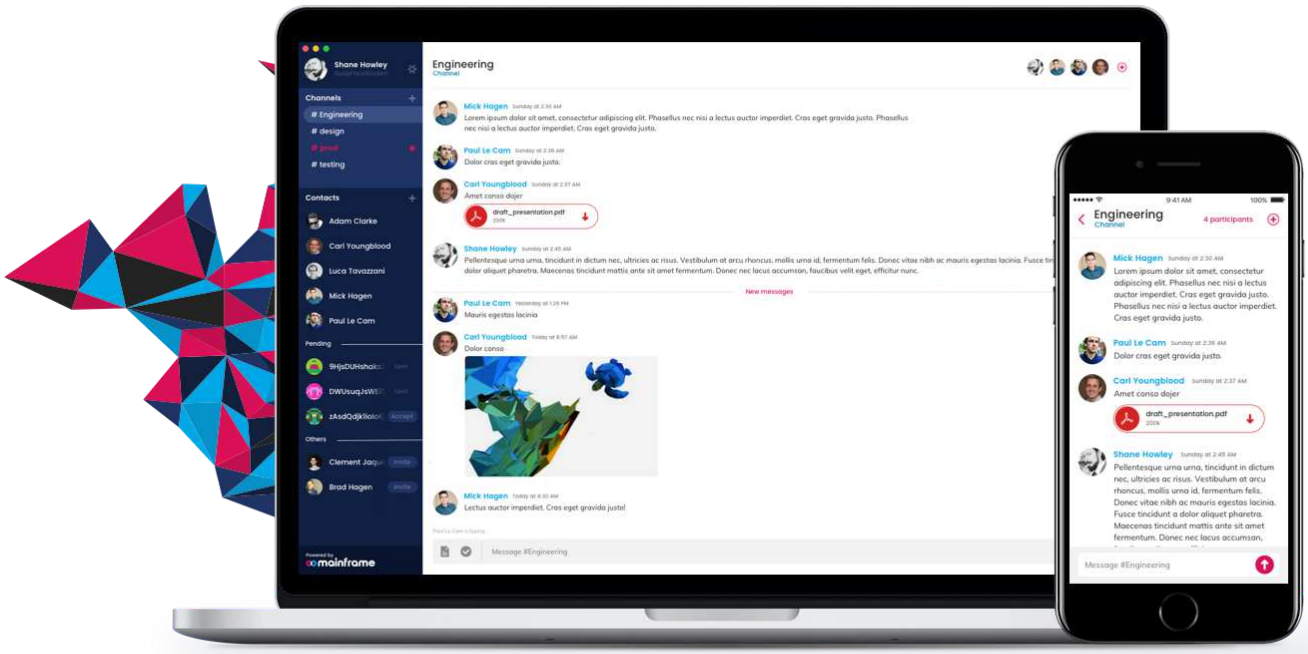
Onyx makes it easy for users to alert others of their status both in their profile and during conversations by indicating whether another user was last seen active by their client and whether they are currently typing a message. These notifications can be disabled.

Bot marketplace

Onyx is highly extensible, allowing users to create apps that are custom-tailored for specific industry and individual needs and buy and sell them using Mainframe tokens.

Examples of useful bots include:

- autoresponders listening for keywords in channels and responding to the group with relevant information;
- agents that can be commanded to perform various domain-specific tasks
- apps that provide other cloud-based and/or decentralized services to users and organizations



Additional use cases

There are undoubtedly numerous unanticipated ways that the Mainframe platform can be used, but here are some areas of innovation that we believe it is well-suited for.

Enterprise

The General Data Protection Regulation (GDPR) passed by the EU in 2016 requires enterprise IT practices to comply with strict privacy measures. Granting IT Admins a platform focused on user sovereignty, corporations can design streamlined systems without the risk of leaking information in transit. Liability is reduced when sensitive data is isolated within a secure system.

In a similar fashion, Healthcare IT professionals in the US could use this platform to build solutions that are compliant with the Health Insurance Portability and Accountability Act (HIPAA). Professionals in finance and academia need to control their own data for similar reasons.

Consumer products

Much attention has been paid to the Internet of Things, and almost as much to the security threat these devices pose. Hardware with sensors and inputs (but fewer direct user interactions) can become a prime target for hacking. The Mirai botnet knocked out Internet access for much of the US East Coast in October 2016.[42] Mainframe's addressing and routing mechanisms, as well as its integration with decentralized identity and reputation layers, can significantly reduce the attack vector for connected devices.

Government

Perhaps counterintuitively, these same tools may be just as useful to governments. Sensitive information is an essential part of government operation, and passing data within and between agencies is crucial. Agencies could use the Mainframe network to reduce the risk of information theft and leakage, as well as malware infections.

Social networking and fintech

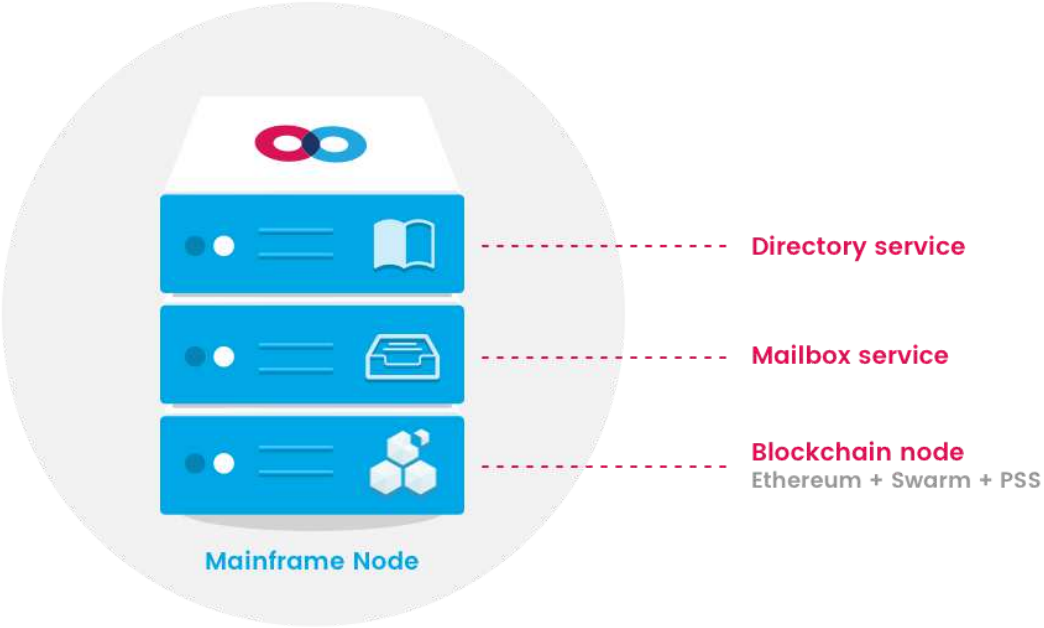
The Mainframe platform could be leveraged for community building, creating decentralized systems like Twitter or Reddit. Payment systems are proliferating rapidly, but most real-world consumer applications are still highly centralized and thus subject to surveillance and interference. The Mainframe platform could power a variety of decentralized, consumer-facing finance applications, such as WeChat or Venmo, but independent of any individual host or provider.

Development Roadmap

Mainframe has identified the following milestones for the rollout of the platform and Onyx.

Milestone 1 — “Apollo”

This first milestone will require transitional managed infrastructure before all the protocols for fully unhosted applications become available. The managed infrastructure required for each user or organization is called a Mainframe node. A Mainframe node consists of a blockchain node with a messaging layer and various services necessary for storing messages and managing contacts. The first supported blockchain will be Ethereum and will make use of the Swarm distributed storage platform and PSS secure messaging protocol for surveillance-resistant packet routing and file storage. Mainframe shares the Ethereum Foundation’s strategic vision for Swarm, and is devoting significant engineering resources towards improving it and accelerating its delivery and adoption.



Mainframe will provide maximum user sovereignty within this hybrid mode of operation, offering both a managed node service that can be subscribed to with Mainframe tokens and a self-hosted option with convenient utilities for deployment to popular infrastructure platforms.

A beta version of Onyx will be released providing:

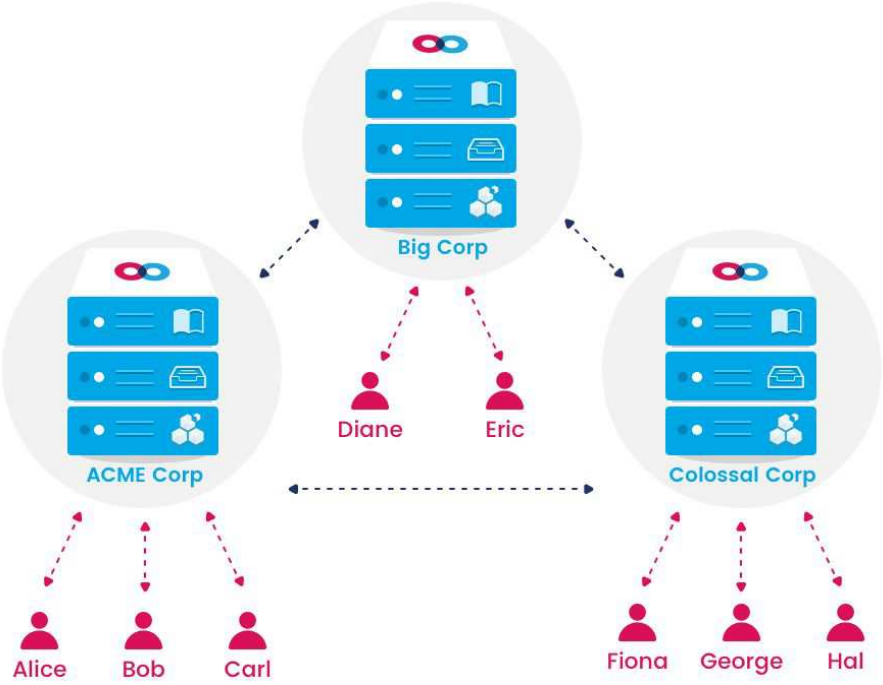
- reliable direct and group messaging
- secure file attachments with viewers for common MIME types
- full-text search for messages and contacts

Milestone 2 — “Hawthorne”

This release will see the beginnings of incentivization with incentivized packet routing, including node service discovery, functional swap contracts and invoicing.

Onyx will continue with the hybrid partially managed infrastructure, adding a shared mailbox mode of operation. In its default and most secure mode of operation, each Mainframe node services a single user, ensuring that users' stored messages never reside in a shared database. However, some companies operate in industries that require data retention and auditing, and are willing to accept the necessary additional trust in their users that is required by such a mode of operation.
In *shared mailbox mode*, all clients within an organization share the same Mainframe node. This node serves as a transport layer for all messages sent by any individual in the organization. Messages are encrypted by an organization member before sending them out over the Mainframe node, which encrypts them again for the transport layer. The same process occurs in reverse when a Mainframe node receives messages for a user in shared mailbox mode. This ensures that even when running in this mode, messages continue to be visible only to private key holders.

When running in shared mailbox mode, however, a greater level of trust is required after the message is decrypted. It is then passed on to the mailbox service to be saved in an unencrypted format so that it can be searched and retrieved by all of the user’s other clients. Client access is still authenticated and users can only access their own messages, but the messages are stored in a shared database. This database can be audited and reviewed by administrators with the correct permissions.



Shared mailbox mode

Additional Onyx features will be implemented, including:

- Organization-wide contact management
- Threads
- Reactions
- Emojis
- Reminders
- Task sharing
- Presence

Milestone 3 — "Gettysburg"

This milestone will be the culmination of our efforts to build a fully-decentralized communications platform. It will support full incentivization of all protocol layers, including packet routing, delivery, hodling, file storage, and data services, thus enabling Onyx to run in fully unhosted mode.

Richer identity and user credit rating APIs will become available, making it easier for nodes to avoid default risk.

This release will also include the launch of our developer SDK, enabling individuals, startups and other organizations to develop applications on the Mainframe platform.

We will also release a fully-decentralized, user-curated marketplace for the products and services that developers build, allowing everyone to participate in this new token economy.

Finally, this release will include smart contracts, oracles, and libraries for interacting with the incentivization layers from other blockchains and development paradigms.

Endnotes

1. Elliott, Francis; Duncan, Gary (2009). *The Times of London*: "Chancellor Alistair Darling on brink of second bailout for banks." https://www.thetimes.co.uk/article/chancellor-alistair-darling-on-brink-of-second-bailout-for-banks-n9l382mn62h

2. Anonymous (2017). *Edelman Insights*: "2017 Edelman TRUST BAROMETER™- Global Results." https://www.slideshare.net/EdelmanInsights/2017-edelman-trust-barometer-global-results-71035413

3. Rogers, Simon (2011). *The Guardian*: "Occupy protests around the world: full list visualised" https://www.theguardian.com/news/datablog/2011/oct/17/occupy-protests-world-list-map

4. Yonego, Joris Toonders (2014). *Wired*: "Data Is the New Oil of the Digital Economy." https://www.wired.com/insights/2014/07/data-new-oil-digital-economy

5. Vanian, Jonathan (2016). *Fortune*: "Why Data Is The New Oil" http://fortune.com/2016/07/11/data-oil-brainstorm-tech

6. Akhtar, Omar (2014). *DMN*: "Why Data is the New Oil" http://www.dmnews.com/marketing-strategy/big-data-is-the-worlds-natural-resource-for-the-next-century--ibm-ceo-ginni-rometty/article/346991

7. Anonymous (2015). *Juniper Research*: "Cybercrime Will Cost Businesses over $2 Trillion by 2019." https://www.juniperresearch.com/press/press-releases/cybercrime-cost-businesses-over-2trillion

8. Simmons, Randy (2017). "How Chinese hacking felled telecommunication giant Nortel" https://www.linkedin.com/pulse/how-chinese-hacking-felled-telecommunication-giant-nortel-simmons-1

9. McMillan, Robert; Knutson, Ryan (2017). *The Wall Street Journal*: "Yahoo Triples Estimate of Breached Accounts to 3 Billion." https://www.wsj.com/articles/yahoo-triples-estimate-of-breached-accounts-to-3-billion-1507062804

10. Gara, Tom; Warzel, Charlie (2014). *Buzzfeed*: "A Look Through The Sony Pictures Data Hack: This Is As Bad As It Gets" https://www.buzzfeed.com/tomgara/sony-hack

11. Franceschi-Bicchierai, Lorenzo (2016). *Motherboard*: "How Hackers Broke Into John Podesta and Colin Powell's Gmail Accounts." https://motherboard.vice.com/en_us/article/mg7xjb/how-hackers-broke-into-john-podesta-and-colin-powells-gmail-accounts

12. Lewkowicz, Kayla (2017). "The Top 10 Most Popular Email Clients of 2016." https://litmus.com/blog/the-top-10-most-popular-email-clients-of-2016

13. Galloway, Scott (2017). *L2*: “Lie to Me.” https://www.l2inc.com/daily-insights/no-mercy-no-malice/lie-to-me

14. Andreesen, Mark (2011). *The Wall Street Journal*: “Why Software is Eating the World.” https://a16z.com/2016/08/20/why-software-is-eating-the-world

15. Braden, Robert et al (1989). “Requirements for Internet Hosts -- Communication Layers” https://tools.ietf.org/html/rfc1122

16. Mockapetris, Paul (1987). “Domain Names: Concepts and Facilities.” https://tools.ietf.org/html/rfc1034

17. “Bylaws for the Internet Corporation for Assigned Names and Numbers.” https://www.icann.org/resources/pages/governance/bylaws-en

18. Dierks, Tim; Rescorla, Eric (2008). “The Transport Layer Security (TLS) Protocol Version 1.2.” https://tools.ietf.org/html/rfc5246

19. Russell, Jon (2017). "Cryptocurrency exchange EtherDelta suspends service following alleged hack" https://techcrunch.com/2017/12/20/etherdelta-suspends-service/

20. Postel, Jonathan (1982). “Simple Mail Transfer Protocol.” https://tools.ietf.org/html/rfc821

21. Crispin, Mark, R. (1994). “Internet Message Access Protocol - Version 4” https://tools.ietf.org/html/rfc1730

22. Hansen, Tony et al (2009). “DomainKeys Identified Mail (DKIM) Service Overview.” https://tools.ietf.org/html/rfc5585

23. Oikarinen, Jarkko (1993). “Internet Relay Chat Protocol.” https://tools.ietf.org/html/rfc1459

24. Saint-Andre, Peter (2011). “Extensible Messaging and Presence Protocol (XMPP): Core.” https://tools.ietf.org/html/rfc6120

25. Zimmermann, Philip (1991). “Why I Wrote PGP.” https://www.philzimmermann.com/EN/essays/WhyIWrotePGP.html

26. Grossman, Lev (2016). TIME: “Inside Apple CEO Tim Cook’s Fight With the FBI.” http://time.com/4262480/tim-cook-apple-fbi-2

27. Ong, Thuy (2017). *The Verge*: “WhatsApp reportedly refused to build a backdoor for the UK government.” https://www.theverge.com/2017/9/20/16338128/whatsapp-reportedly-refused-request-uk-government-access-encrypted-messages

28. Reimer, Jeremy (2008). *Ars Technica*: "Bavarian government caught looking for Skype backdoor." https://arstechnica.com/information-technology/2008/01/bavarian-government-caught-looking-for-skype-backdoor

29. Conrad, Alex (2017). *Forbes*: "Slack Passes 6 Million Daily Users And Opens Up Channels To Multi-Company Use" https://www.forbes.com/sites/alexkonrad/2017/09/12/slack-passes-6-million-daily-users-and-opens-up-channels-to-multi-company-use

30. Shaban, Hamza (2018). *The Washington Post*: "Slack went down, posing a momentary crisis in offices around the country." https://www.washingtonpost.com/news/the-switch/wp/2018/01/09/slack-went-down-posing-a-momentary-crisis-in-offices-around-the-country

31. Maymounkov, Petar; Mazières, David (2002). "Kademlia: A Peer-to-Peer Information System Base on the XOR Metric." https://pdos.csail.mit.edu/~petar/papers/maymounkov-kademlia-lncs.pdf

32. Holbrook, Louis (2017). "Postal Services over Swarm." https://github.com/ethersphere/go-ethereum/blob/ddfc0a2a02ce574f4c252068ce81f0f5ada1c1ff/swarm/pss/README.md

33. Wood, Gavin (2015). "Whisper Wire Specification." https://github.com/ethereum/wiki/blob/965b7cd71fdcf2b8b2de8d36061b0b45678072d2/Whisper-PoC-2-Protocol-Spec.md

34. Trón, Viktor; Fischer, Aron (31 Dec 2017). "Generalized swap, swear, and swindle games." https://www.dropbox.com/s/7r3jasjho35ojc7/sw3paper.pdf

35. Ibid, 19

36. Trón, Viktor; Fischer, Aron; Nagy, Dániel; Felföldi, Zsolt; Johnson, Nick (2016). "Swap, Swear, and Swindle: Incentive System for Swarm." http://swarm-gateways.net/bzz:/theswarm.eth/ethersphere/orange-papers/1/sw%5E3.pdf

37. Anonymous (2017). "Wolk SWARMDB: Decentralized Database Services for Web 3." https://wolk.com/whitepaper/WolkTokenGenerationEvent-20170717.pdf

38. The term "unhosted" and the conceptual diagram have been adapted from remotestorage.io.

39. "Ethereum Name Service." https://ens.domains

40. "uPort: Self-Sovereign Identity." https://www.uport.me

41. "Sovrin: Identity for All" https://sovrin.org

42. Newman, Lily Hay (2016). *WIRED*: "What We Know About Friday's Massive East Coast Internet Outage" https://www.wired.com/2016/10/internet-outage-ddos-dns-dyn