Tierion Network

A global platform for verifiable data

Authors: Wayne Vaughan, Jason Bukowski, Glenn Rempe

June 25, 2017

Table of Contents

Introduction 4

History 5

Use Cases 6

Tierion Network 8

Chainpoint Service 10

- Benefits 10
- Chainpoint Proofs 11
- Chainpoint Proof Generation 20
- Chainpoint Service Infrastructure 22

Economics 26

Tierion Network Token 27

Conclusion 28

Appendix A: Full Chainpoint 3.0 Proof Sample 29

Endnotes 33

Introduction

Businesses safeguard and notarize important documents such as property titles and contracts to ensure anyone can prove their veracity. Surprisingly, there isn't a universal equivalent for safeguarding digital data. Much of the world's important information is stored digitally and is susceptible to modification by system administrators or hackers. When data is sent over the Internet, the recipient often can't verify when the data was created or if it has been modified from its original state. Problems with data integrity and digital record-keeping are particularly severe in regulated industries such as healthcare, insurance, and financial services where data corruption or tampering has significant legal and reputational consequences.

Tierion is building a universal platform for data verification. Tierion works by creating a proof that links data to a transaction on a blockchain. This is called anchoring. Anyone with this proof can verify the data's integrity and timestamp without relying on a trusted authority.

Tierion launched in 2015 and has become the most widely used platform for anchoring data to the blockchain. Tierion's key innovation was making it simple to anchor a virtually unlimited amount of data to a single blockchain transaction. Tierion's launch also marked the introduction of the Chainpoint protocol, the first standard proof format for anchoring data to a blockchain.

Several open source projects and multiple vendors have adopted the Chainpoint protocol. In July 2016, Chainpoint 2.0[1] was published. It featured several improvements, including the use of JSON-LD[2] and anchoring into multiple blockchains. Tierion formed the Chainpoint W3C Community in September 2016[3]. Chainpoint is the first technology of its kind to receive public support from a major Internet standards organization[4].

To continue our mission to secure the world's data, we're launching the Tierion Network. Chainpoint has been upgraded to version 3.0 and now runs as a service on the Tierion Network. Together, they provide a universal platform for data verification that operates at massive scale.

Microsoft is joining Tierion in running a core part of the network infrastructure. Anyone can join the Tierion Network by running a node. Each node improves the network's scalability and reliability. Along with this new distributed architecture, a token is being introduced. The Tierion Network Token (TNT) provides an economic incentive to secure the network infrastructure, and serves as a method of settlement between parties to access network resources.

Each Node serves as a mini-Tierion. Node operators can provide users with services using conventional payment and delivery models.

History

Using the Bitcoin blockchain[5] to notarize documents was popularized by Manuel Aráoz with the creation of Proof of Existence[6] in 2012. This system, and others like it, publish a hash of a document in a Bitcoin transaction. Hashes allow computers to compare arbitrarily complex data and determine if they are identical. By comparing the hash of a document with the hash published in the blockchain, it is possible to prove the document existed before the timestamp of the block containing that Bitcoin transaction.

Why use Bitcoin?

Bitcoin's security model is enforced by the entire network instead of a trusted central authority. Once a transaction is confirmed, it becomes part of an immutable ledger that is distributed across a global network of nodes. It is practically impossible for a malicious agent to alter data on the blockchain.

Shortcomings

These early systems had several shortcomings and never achieved significant commercial adoption:

1) **Not Scalable** - One document hash was published per Bitcoin transaction. Bitcoin's current network throughput is approximately three transactions per second[7]. This is far too low to support the world's applications.

2) **Cost** - In June 2015, the cost of anchoring data into Bitcoin was approximately $0.03 USD. In June of 2017, that cost has increased over 100x to $3.40 USD.

3) **Inaccurate** - Bitcoin's block time accuracy is ± 2 hours[8]. This means the timestamp of the block could be an hour before a transaction was published. Time travel violates the laws of physics.

These limitations made it impractical and cost prohibitive to anchor large volumes of data in the Bitcoin blockchain.

Tierion overcame these obstacles and made it simple to link a virtually unlimited amount of data to a single transaction on the blockchain. For the first time, developers had an easy-to-use and affordable service for anchoring data at scale. Chainpoint provided a standard proof format and open source tools for the creation and verification of Chainpoint proofs.

Use Cases

Since launching in June 2015, Tierion has been used by many organizations in a wide variety of industries. Several open source projects have adopted the Chainpoint protocol. Here are some examples that demonstrate how the technology is being used.

Microsoft - ***Attestations & Data Integrity***
Microsoft and Tierion are collaborating to build a service to generate, manage, and validate attestations[9]; e.g. credentials associated with your work history.

Philips - ***IoT Data Provenance***
Collect data from MRI machines to create an audit trail of its maintenance, usage, and calibration history. Prove compliance with regulations and safety inspections.

Blockcerts - ***Blockchain Verifiable Credentials***
Blockcerts is an open source project that emerged from the MIT Media Lab that uses the Chainpoint protocol to issue blockchain verifiable education credentials.

Verifai - ***Machine Learning Auditability***
Verifai creates a cryptographically verifiable audit trail to prove how a neural network has been trained. Verifai marks the first practical use of blockchain technology in artificial intelligence.

Use Case	Description
Process Audit Trail	Cryptographic proof of the order, integrity, and timestamp of any business process. Supply chain, insurance claims, know your customer (KYC), hospital patient care, financial transactions.
Document Timestamping	Several companies offer free timestamping services using the Chainpoint protocol.
IoT Data Collection	Prove the integrity and timestamp of data as it is gathered from connected devices.
Proof of Consent	Non-repudiable evidence that important consent and approvals have been recorded. Patient care, corporate governance, etc.
Registry	A verifiable registry of creative works, real estate listings, etc.

Data Security	Secure the integrity of key IT assets; customer records, databases, log files, backups, and virtual machine snapshots.
Clinical Trials	Provide regulators with proof of the integrity of data for clinical trials.

For each industry specific use case, Tierion delivers three core capabilities.

Trust anchor

In cryptographic systems, a trust anchor[10] is an authoritative entity for which trust is assumed and not derived. The Bitcoin blockchain is particularly well suited to serve as a trust anchor. No authority controls the Bitcoin blockchain. Once a transaction is confirmed, it becomes part of an immutable ledger that is distributed across a global network of nodes. Erasing or modifying this data is virtually impossible. Tierion provides a scalable means to use multiple blockchains as a trust anchor.

Data integrity

Data Integrity is the assurance of the accuracy and consistency of data. Organizations with sensitive data need to prove it hasn't been corrupted or manipulated by insider threats or external hackers. Tierion provides a global mechanism for verifying data integrity.

Timestamp

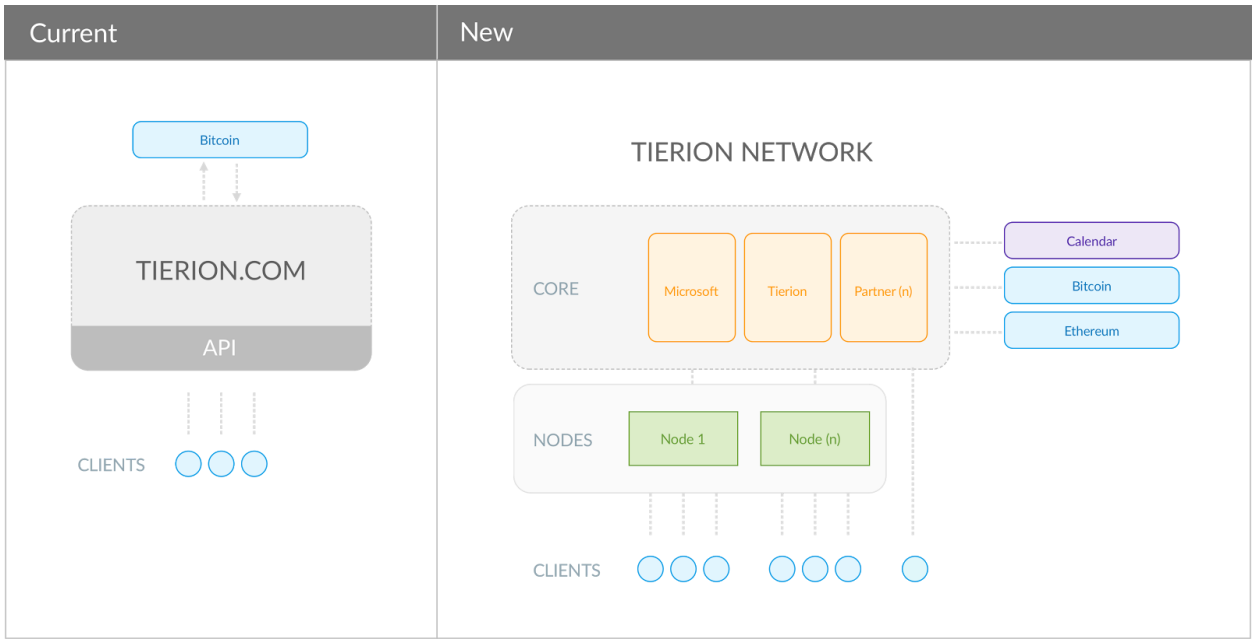
Public blockchains make poor timestamp authorities because of their low time accuracy. For example, Bitcoin's block time accuracy is approximately ±2 hours. Trusted timestamps are accurate, but require you to trust the time provided by an authority. Anchors to a public blockchain provide a low accuracy but trustless timestamp. Chainpoint solves this dilemma by including multiple trusted timestamps and multiple trust anchors in each proof. This allows Chainpoint proofs to simultaneously possess accurate and trustless time attestations.

Tierion Network

Developers get started using the current version of Tierion by visiting the website and signing up for a free account. This gives them access to tools for collecting data, creating and verifying Chainpoint proofs, and integrating data with 500+ popular software applications.

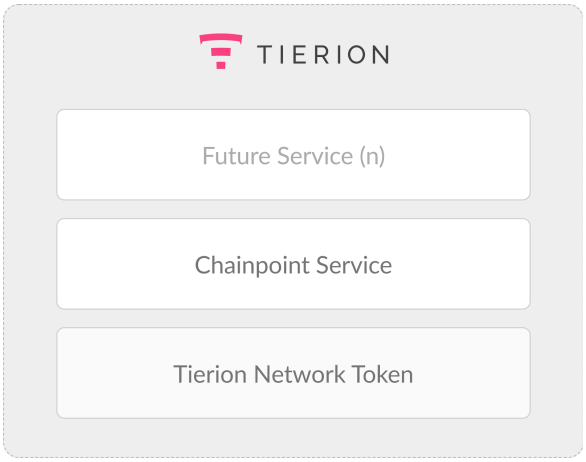
With this next step, Tierion is evolving into a distributed network that offers services that utilize the blockchain as a trust anchor.

Figure 1. Current Tierion vs New Tierion Network



Chainpoint is the first service on the Tierion Network and serves as the technical foundation for future services. Chainpoint provides a global utility for anchoring data to the blockchain and a universal platform for data verification.

Future additions to the Tierion Network may include services for securing and sharing verifiable data, document notary and archival, and attestations related to blockchain verifiable identities.



The Tierion Network Token (TNT) serves two primary functions:

1) A method of settlement between parties to access network resources
2) An incentive for network participants to operate and secure the network.

Anyone can join the Tierion Network and earn TNT. End users will not require a token to use the network. See the “Tierion Network Token” section of this document for more details.

Roadmap

The original version of Tierion has a nearly two year track record and has been used by thousands of organizations. The application will continue to operate. Current customers will be able to migrate to a new version of Tierion.

The Tierion Network and Chainpoint Service has been operating in private beta with our partners. An open beta is planned to launch in August, which marks the two year anniversary of Tierion’s launch. Microsoft’s infrastructure is planned to come online shortly thereafter.

The Tierion Network is planned to launch before the end of 2017.

Chainpoint Service

The Chainpoint Service is a global utility for creating and verifying Chainpoint proofs that runs on the Tierion Network. This section of the white paper provides a detailed technical overview of Chainpoint 3.0. It is organized into four sub-sections:

Benefits - benefits of the new Chainpoint 3.0 service
Chainpoint Proofs - an overview of the elements of a Chainpoint proof
Proof Generation Process - how Chainpoint proofs are created
Chainpoint Infrastructure - a description of Chainpoint's global network infrastructure

Benefits

Chainpoint's new distributed architecture provides several significant advantages.

1) **Scalability** - Chainpoint uses a highly scalable architecture that is designed to generate millions of proofs per second.

2) **Accuracy** - Chainpoint includes time data from Network Time Protocol (NTP) servers and the National Institute of Standards and Technology (NIST) with anchors to the Bitcoin and Ethereum blockchains, allowing Chainpoint proofs to simultaneously possess accurate and trustless time attestations.

3) **Responsive** - Chainpoint responds immediately when a hash is submitted. Proofs are automatically upgraded as they are anchored to the Bitcoin and Ethereum blockchains.

4) **Trust** - Chainpoint periodically anchors into the decentralized Bitcoin and Ethereum blockchains. This allows the Chainpoint proofs to inherit the security properties of multiple blockchains.

5) **Cost Effectiveness** - Chainpoint's scalability makes it cost-effective for the world to anchor data to a secure public blockchain. This is particularly relevant as Bitcoin transaction fees have increased more than 100x in the past two years and are likely to continue to rise.

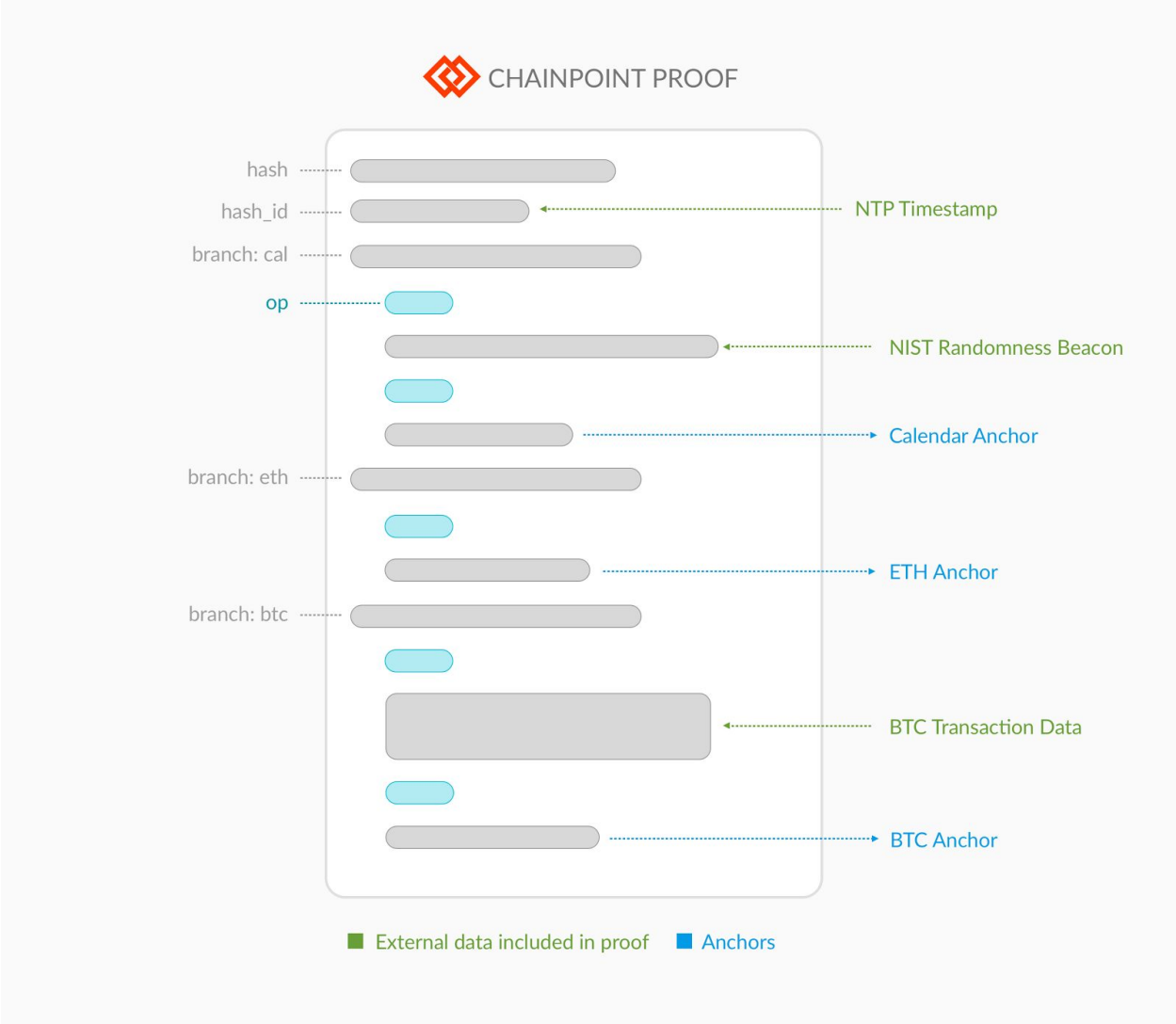
6) **Global Calendar** - Chainpoint servers work in consensus to generate a global, publicly auditable, blockchain called the Chainpoint Calendar. This makes it easier to verify Chainpoint proofs and audit the network.

Chainpoint Proofs

Each Chainpoint proof contains a set of operations that cryptographically link a hash to multiple blockchains. These links are called anchors. Chainpoint proofs are verified by replaying the operations and checking each anchor for an expected value.

The new proof format has the flexibility to support the inclusion of external data, multiple hash types, and branches. Chainpoint proofs can be verified with any Chainpoint compatible verification tool.

*Figure 2. **Conceptual depiction of the structure of a Chainpoint 3.0 proof.***



Sample Chainpoint Proof

This sample proof has been truncated for readability. See Appendix A for a full proof.


```
á
  íî[gfl]plíÆ í`llhkÆ××oüa\Çgj_×[`Yafhgafl×nüíÄ
  ílqh]íÆ í#`YafhgaflíÄ
  í`Yk`íÆ íþû\YúYZ[úÿùãZ^üÃYûùý^ü\Ãÿÿýþýú^Y\ãã\Z\Ãúùúý[\ü]þ^ùþýûZÃã[ùùZÃãÃ[íÄ
  í`Yk`ëa\íÆ íããþüZúÃùÈÿùÿúÈúú]ÃÈÄüûûÈýþüþýãýÃ]ÿûÃíÄ
  í`Yk`ëkmZeall]\ëYlíÆ íûùúÃÈùÃÈùý4ùûÆüÿÆùÃ:íÄ
  íZjYf[`]kíÆ ß
    á
      ídYZ]díÆ í[YdëYf[`gjëZjYf[`íÄ
      íghkíÆ ß
        á
          ídíÆ íããþüZúÃùÈÿùÿúÈúú]ÃÈÄüûûÈýþüþýãýÃ]ÿûÃí
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
        á
          ídíÆ
íúýÃÃúüþÃÿùÆããÃùüÿ[Y[ÿ^ý\Ã\Z^[úüÿÄü\YûYþþã^ÿþ^[ûýÿã\ûÿ]^ü^üÃüýZã\þ]úÃ]ãÿ\ÃÿúÿÄÃü]ý\ûý\]
Z[[\^ûÿÃý\úÃþ^ÿÿÃûý[\[^þúÃãýùÿ[úÃÿÃ\ãüYüþZÃYùù\ü^úû[Zí
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
        á
          ídíÆ íúýùÃÆúýÃÃúüþÃÃúÃûÃÆúÆYÇ[`YafhgaflÇgj_Æ[YdÆúýùÃí
        âÄ
        á
          íjíÆ í]ÿú]]ZZ]ûÃÃ]\ûÃÿ\ûùùùþ\[\úýÿãùþ[úãýÿYýZÃùÃùÿÿ[ýùYúÃÿÃùü^ãýýÿýY[]í
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
        á
          íYf[`gjkíÆ ß
            á
              ílqh]íÆ í[YdíÄ
              íYf[`gjëa\íÆ íúýùÃíÄ
              ímjakíÆ ß
                í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úýùÃ×`Yk`íÄ
              à
            â
          à
        â
      à
    â
  à
â
```

Chainpoint Proof Overview

This section provides a overview of the Chainpoint proof elements in ***Figure 2***.

hash

The `hash` element of a Chainpoint proof represents some data that you want to prove is anchored to the blockchain.

A hash is a cryptographic digest of any data. Hash functions always produce the same hash given identical input. Hash functions output a fixed length string regardless of the size or type of input.

Input	Size	Hash (SHA256)
text string	10 bytes	`]^ĀĀĂĀ]úü\üYĀþþûÿĂýÿYüZ[^ùù\Y][Ă^[Ă[Ă[ý\þú\\[Ā[[þ\^āāā^Āý\\ýüý\ú`
image	400MB	`þúÿúZ[ùþā]ÿü\ā\ZùùĂZûú\][ZYû^[ýþþ^ā]ý[^]āþüÿ]ĂYYúúù]]Ā\\Ăúý[YÿāY`
database	56GB	`ü\ûûZĀ]ýYü]\āāāü\ĂüĂþ^ÿ]þùÿý[Z]ZĀ\ûĀ\ÿĂÿûûāùûÿY[ZþāùĂúÿĀāýāýùùþ]`

Hashes allow computers to compare data. If hashes match, the source data must be identical. Hash functions cannot be reversed to discover anything about the input. This makes hashes useful for proving the existence and integrity of data while keeping the source confidential.

hash_id & NTP timestamp

Chainpoint uses Network Time Protocol (NTP)[11] to keep time synchronized with a worldwide network of atomic clocks. Chainpoint generates a unique identifier for each hash it receives. This hash_id is an RFC 4122 Version 1 UUID[12] which contains a high precision timestamp that reflects the NTP time. A `hash_id` is included in the cryptographic operations, thus the exact time Chainpoint received the hash is embedded in each proof.

branches

Chainpoint proofs are organized into a tree. The `hash` serves as the root. Each branch contains a list of operations which terminates in an anchor; a claim that a value is published in an external system. ***Figure 1*** contains three branches, calendar, ethereum, and bitcoin.

operations (ops)

Operations are performed in sequence to verify a proof. Operations include left concatenate, right concatenate, and a set of hashing functions. Chainpoint can be extended to include additional operations.

The `Yk` field of a Chainpoint proof is used as the starting value when performing operations. The result of each operation is used as the input for the next operation. Consider the following example which begins with an empty string (instead of a hash) as the starting value:

```
á
  ídíÆ í[`Yafí
âÄ
á
  íjíÆ íhgaflí
âÄ
á
  íghíÆ ík`YÈûþÿí
âÄ
```

The above sample translates to: SHA256(‘chain’ | ‘point’)
resulting in `[ā[ÿù]Ă[ü[ú]ÿĂü]ùZýY[Ă[\û\YāĂ[ÿĀ\^úZāü\ýYþYûĀZ]ĀZYYāýú[úú[^[ĀZùĂ`

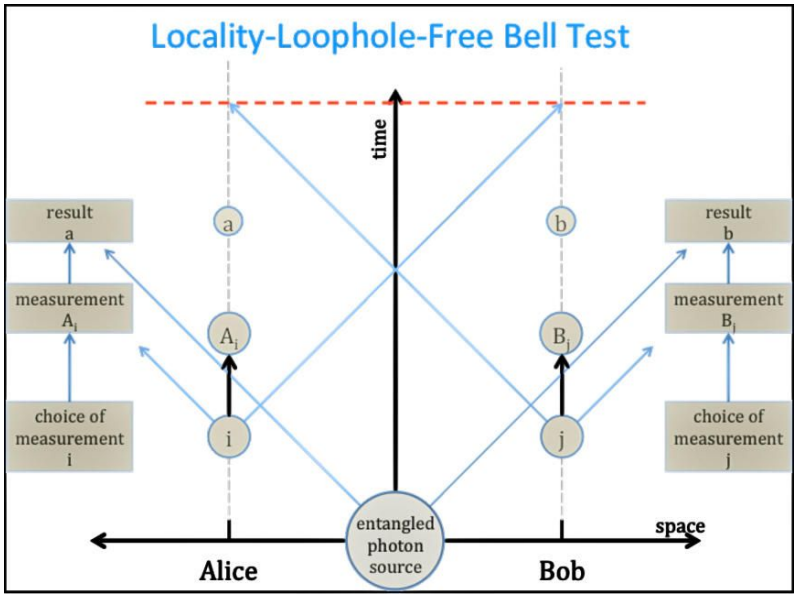
Operations replay the calculations made when a hash goes through the proof generation process. A proof can be verified by executing the operations and checking each anchor for an expected hash value.

NIST Randomness Beacon

In the movies, kidnappers provide “proof of life” by taking a photo of the victim holding a current newspaper. This proves the photo was taken after the newspaper was printed. Our team collaborated with the National Institute of Standards and Technology (NIST) to create an analogous technique to prove a Chainpoint proof was created after the `Yk`ëa\ is generated[13].

The NIST Randomness Beacon project, led by Dr. Rene Peralta, continues work started by Haber and Stornetta at AT&T Bell Labs[14] in the 1990’s. Each minute, the Randomness Beacon publishes a value created by a network of random number generators. Beacon values are generated by specialized hardware that ‘uses quantum effects to generate a sequence of truly random values, guaranteed to be unpredictable, even if an attacker has access to the random source’[15].

*Figure 3. **A space-time diagram illustrating a locality-loophole-free Bell test***



NIST Beacon data is included in every Chainpoint 3.0 proof. Since the random values are unknowable before they are published, we can assert that each Chainpoint proof:

- Should have a NIST timestamp earlier or equal to when a hash was received
- The NIST time should generally be within one minute of when a hash was received

This is the first time publicly verifiable data is being used to improve the accuracy of a blockchain timestamp proof.

Calendar Anchor

The Chainpoint Calendar is similar to a hash calendar[16]; it is a blockchain that is created and maintained by the participants of the Chainpoint Network. Each Chainpoint proof is anchored to this global calendar within seconds of hash submission. Once anchored to the global calendar, Chainpoint generates a partial proof, eliminating the need to wait for Bitcoin and Ethereum transactions to confirm.

```
á
  íYf[`gjkíÆ ßá
     ílqh]íÆ í[YdíÄ
     íYf[`gjëa\íÆ íúùûÃíÄ
     ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úùûÃ×`Yk`í à
    âà
â
```

The Calendar also contains data for verifying Chainpoint proofs. Calendar data is mirrored by a distributed network of Chainpoint Nodes. More on this later.

Ethereum Anchor

The anchor hash is published in an Ethereum transaction in the data field. The ETH address is included in the Chainpoint proof.

```
á
  íYf[`gjkíÆ ßá
    ílqh]íÆ í]l`íÄ
    íYf[`gjëa\íÆ í\ü]Ã][ãý[ü\Z]ãÿ^Ã\ÃYã]YÿãY]ý\]\ÿ[ùZùúûZ]þúĂ^ýüüYùÃ^úþZ\ÿúû^ZýĀYĂíÄ
    ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úùýã×\YlYí à
   âà
â
```

Bitcoin Anchor

The anchoring hash is included in OP_RETURN of a Bitcoin transaction. As a consequence, this value is included in the raw transaction body, allowing the transaction ID and the Merkle path from that transaction to the Bitcoin block's Merkle root to be calculated.

Chainpoint waits for six confirmations after publishing an anchoring transaction, determines the Merkle path from the transaction id to the block's Merkle root, and appends this data to the Chainpoint proof. This ensures proofs can be verified if OP_RETURN is pruned, or if a verifier has dataset that only contains block header data.

```
á
  íYf[`gjkíÆ ßá
     ílqh]íÆ íZl[íÄ
     íYf[`gjëa\íÆ íýüýÃùûíÄ
     ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úùþÿ×\YlYí à
    âà
â
```

Chainpoint Proof Elements

Complete list of Chainpoint 3.0 proof elements. See Appendix A for a sample proof.

Name	Description
@context ***string, required***	the JSON-LD context for the proof
type ***string, required***	the JSON-LD type definition
hash ***string, required***	hash value between 40 and 128 hex characters. Must be even length.
hash_id ***string, required***	a Version 1 UUID with embedded timestamp. Random number used as MAC input. Timestamp represents server time (UTC) of hash submission.
hash_submitted_at ***string, required***	Human readable ISO 8601 timestamp extracted from time embedded in the hash_id
branches - an array of branch objects. Branches can be nested without limit and MUST be traversed only after executing 'ops'. ***(required only at root)***	
label ***string, optional***	text string representing the branch name
ops ***array, optional***	an array of operations objects. Operations are performed in sequence to arrive at an intermediate hash prior to entering a nested branch.
branches ***array, optional***	nested array of branch objects. Each branch contains ops; labels and additional nested branches are optional.
ops - an array of operation objects ***(required under every 'branches' object)***	
l ***string, optional***	left concatenate a value. If the value is a hexadecimal string, it will be read as a hexadecimal byte array, otherwise the string will be converted to its byte value assuming UTF-8 encoding.

r ***string, optional***	right concatenate a value. If the value is a hexadecimal string, it will be read as a hexadecimal byte array, otherwise the string will be converted to its byte value assuming UTF-8 encoding.
op ***string, optional***	an operation to perform on the current value combined with a previous 'l' or 'r' operation. Current operations: 'sha-224', 'sha-256', 'sha-384', 'sha-512', 'sha3-224', 'sha3-256', 'sha3-384', 'sha3-512', or the special purpose 'sha-256-x2' which applies 'sha-256' twice.
anchors - an array of anchor objects ***(required under every 'ops' object).***	
type ***string, required***	one of 'cal' (Calendar), 'btc' (Bitcoin), or 'eth' (Ethereum) anchor types
anchor_id ***string, required***	an identifier used to look up embedded anchor data. e.g. a Bitcoin transaction or block ID.
uris ***array, optional***	an array of special purpose string URI's, each of which can be used to lookup and retrieve the exact hash resource required to validate this anchor. The URI MUST return only a Hexadecimal hash value as a string. The URI MUST also contain the 'anchor_id' value to lookup the URI resource. This strict requirement is to allow automated clients to retrieve and validate intermediate hashes when verifying a proof. The body value returned by the URI MUST be of even length and match the regex [a-fA-F0-9].

JSON-LD & Binary Formats

Chainpoint proofs are commonly used in their JSON-LD format, as seen in the many examples used throughout this document. The JSON-LD format makes proofs human readable and easy to integrate into other JSON-LD documents.

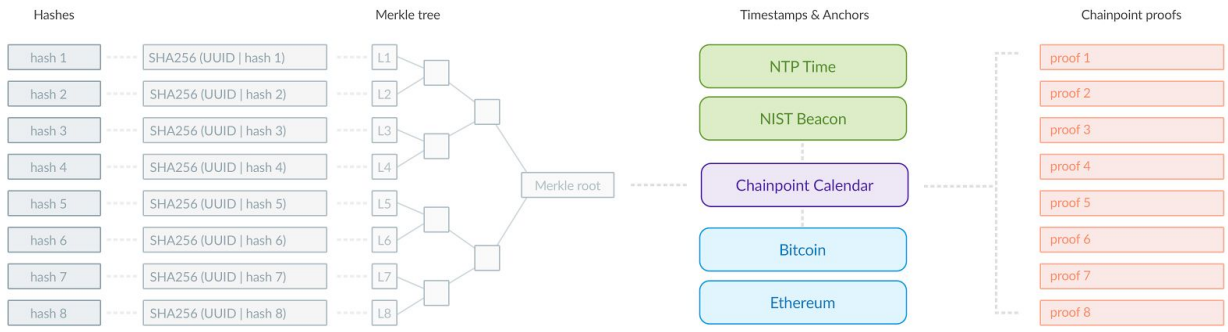
Chainpoint proofs can converted to a binary format. The binary format uses MessagePack and zlib to substantially reduce the proof size. For example, a 5,098 byte JSON formatted proof is reduced by 72% to 1,442 bytes when converted to binary format.

Information on the Chainpoint binary format can be found at https://github.com/chainpoint/chainpoint-binary/.

Chainpoint Proof Generation

The following description approximates how each element interacts throughout the proof generation process.

Figure 4. Chainpoint proof generation process



Hash Submission

Submit a hash to a Chainpoint service. Each submission may contain 1 - 1,000 hashes. Hashes can be hex strings between 40 and 128 characters. This allows submission of common hash types such as SHA-1, SHA-256, and SHA-512. SHA-256 is encouraged. Chainpoint immediately returns a RFC 4122 Version 1 UUID with an embedded NTP timestamp that uniquely identifies each hash.

Hash Processing

The submitted hash is combined with the UUID to create a new hash. This mixing of data acts as a cryptographic nonce and ensures that Chainpoint processes unique hashes even when duplicate hashes are submitted.

Next, the hash is combined with the NIST Beacon data to create another new hash. This makes it possible to prove that the hash was submitted after the NIST Beacon values were published. Hashes are then sent to an aggregation service.

Aggregation

Chainpoint periodically aggregates hashes into number of parallel Merkle trees. This hierarchical aggregation allows for handling massive numbers of hashes. The Merkle root from each tree is periodically sent to the Chainpoint Calendar. A Merkle inclusion proof is generated for each hash and stored. These partial proofs are continually appended with new data throughout the proof generation process.

Calendar Consensus

The Calendar is a blockchain that is kept in consensus between multiple Chainpoint Servers. This ensures that a single global calendar blockchain can be used to verify Chainpoint proofs. Calendar data is organized into blocks. These blocks are stored as records in a distributed cluster of CockroachDB databases[17]. Writes to the calendar are enforced by a leader election using a cluster of Consul[18] servers.

Calendar Blocks

The Chainpoint Calendar periodically aggregates Merkle roots into a new Merkle tree. A new set of Merkle inclusion proofs is generated and appended to the existing partial proofs. The root of this Merkle tree is written to a ***calendar block***.

Anchor Blocks

Calendar blocks are periodically anchored to the Bitcoin and Ethereum blockchain. This is done by publishing a transaction that commits an anchor block hash to a transaction on the blockchain.

Confirmation Blocks

Chainpoint monitors the blockchain. When each ***anchoring transaction*** receives a sufficient number of confirmations, a ***confirmation block*** is added to the Calendar. Each ***confirmation block*** contains the data needed to finalize each Chainpoint proof.

Proof Completion

After a confirmation block is written, Chainpoint appends partial proofs with the final data. Complete Chainpoint proofs are now available for retrieval.

Chainpoint Service Infrastructure

The Chainpoint Service is designed to run as a global network that operates at massive scale. The network involves the interaction of several classes of participants.

Core

Core is a network of partners that run the full Chainpoint Service stack, maintain the global calendar, and anchor data to the blockchain.

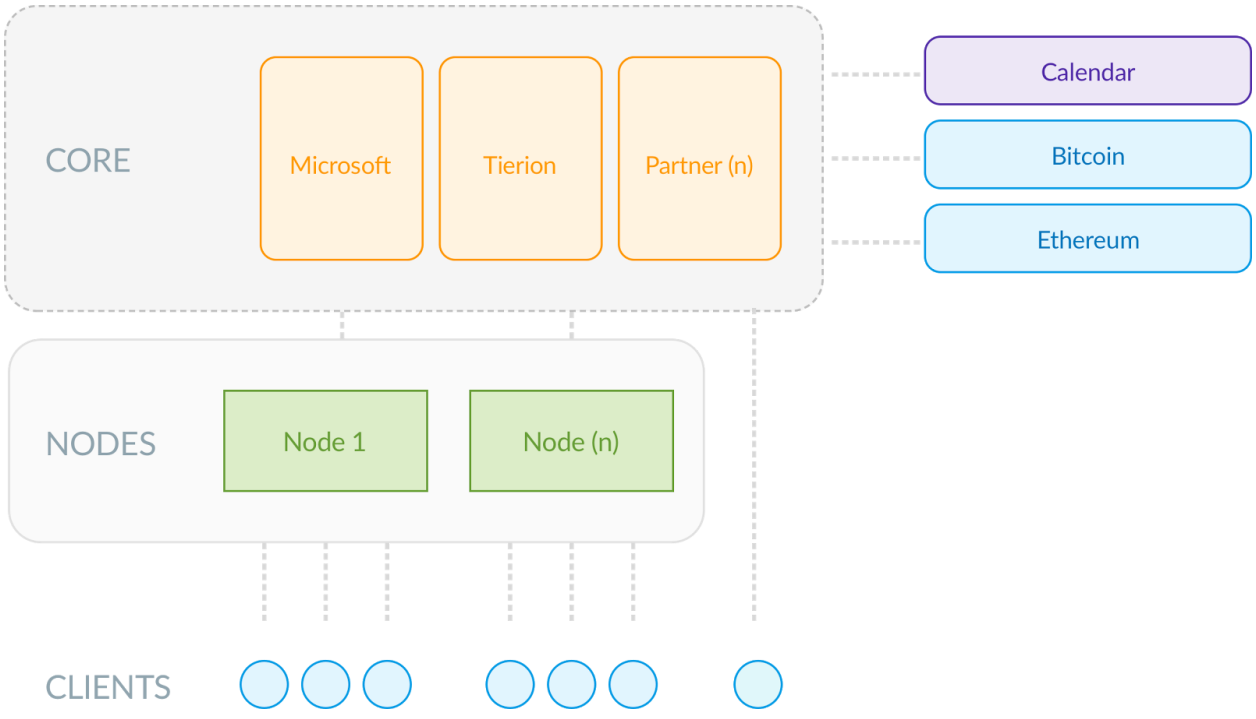
Nodes

Nodes provide additional scaling, mirror the global calendar, and audit Core. Each node that joins the network improves scalability and reliability. Anyone can become part of this distributed network by downloading the software and running a Node.

Chainpoint Clients

Clients can connect to a Node, or directly to Chainpoint Core via an API.

Figure 5. Chainpoint Service architecture diagram



Chainpoint Service Design Goals

Scalable

Chainpoint is designed for virtually unlimited scale. In contrast to other blockchain based systems, throughput increases as nodes are added to the network.

Reliable

Chainpoint is designed to have zero downtime and consistently return proofs in a predictable timeframe. Chainpoint Core is distributed across independent data centers and geographic regions to ensure availability and redundancy. Chainpoint Nodes form a decentralized network to create and verify Chainpoint proofs.

Secure

Anchoring allows Chainpoint to inherit the security properties of multiple blockchains. Modifying the Bitcoin or Ethereum blockchain would cost an attacker millions of dollars and becomes increasingly difficult over time.

Economic Efficiency

Chainpoint is designed to be inexpensive or free for most network participants. Increases in network throughput scales independently of blockchain transaction costs.

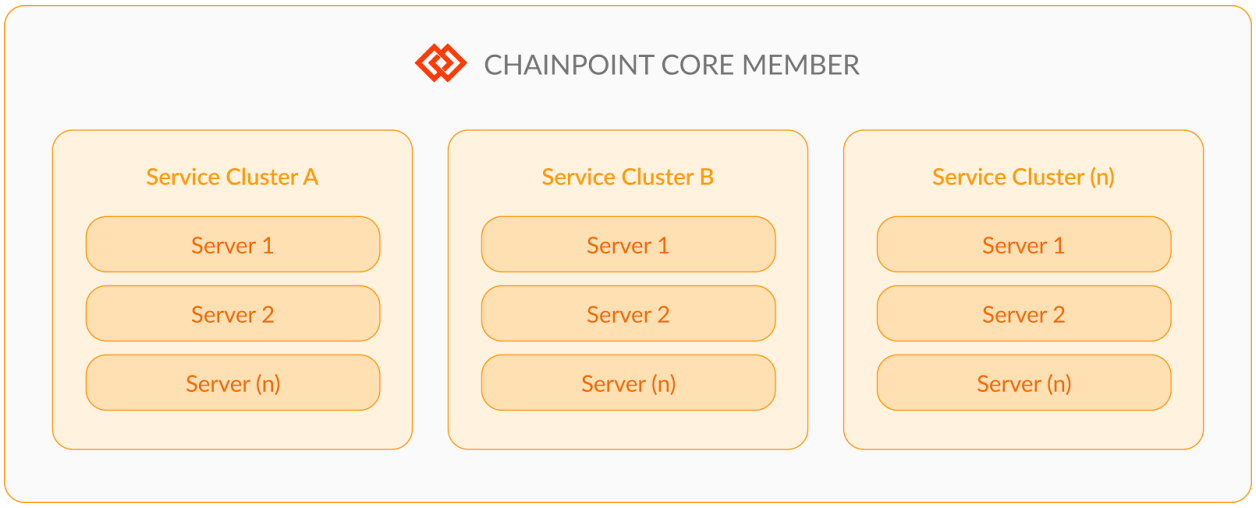
Open

Anyone can join the network by running a Chainpoint Node. Nodes mirror a copy of the calendar data and can independently verify the full chain.

Chainpoint Core

Chainpoint Core is a network of partners that run the full Chainpoint Service stack to create and verify proofs, read and write to the global calendar, and perform anchoring operations. Each Core Member operates one or more clusters of servers. Each cluster is called a service cluster.

Figure 5. Chainpoint Core Member diagram



Core Members have the resources to run scalable systems with high availability and near zero downtime. The first three service clusters will be available at Chainpoint.org. Microsoft is first organization to join Core and will be hosting a service cluster.

Global Calendar

The Chainpoint Calendar is a blockchain that is created by Core and audited by Nodes. The calendar provides several benefits:

Reduced Costs

One Core Partner anchors for everyone on the Chainpoint Network. A single transaction can be used to anchor millions of proofs. This makes Chainpoint inexpensive or free for most network participants.

Faster Response

The full proof generation process can sometimes exceed an hour due to variations in the time it takes to mine Bitcoin blocks. Each Chainpoint proof is anchored to the calendar within seconds of hash submission. Chainpoint then returns a partial proof that is automatically updated throughout the proof generation process. This eliminates the need to wait for Bitcoin and Ethereum transactions to confirm.

Proof Verification

The calendar provides a single source of data for verifying Chainpoint proofs. Anyone with the calendar data can fully verify every Chainpoint proof without having to run a Bitcoin or Ethereum node. You don't have to worry about servers going offline and parts of your proof becoming impossible to verify. Those with advanced security requirements can cross check the calendar data with their own Bitcoin or Ethereum nodes.

Auditability

A global calendar makes it possible for anyone to audit Core and independently verify the validity and integrity of the chain. Each block is signed with a provider specific public key, and the chain is periodically anchored to Bitcoin and Ethereum.

Chainpoint Nodes

Each additional Node increases the total capacity of the network to create and verify proofs. Nodes are able to receive and process hashes, pass hashes up to Core, receive partial proofs from Core, and generate final proofs. Additionally, Nodes mirror a copy of the calendar and continually audit the chain.

Node Operators

Anyone can join the Chainpoint Network by running a Chainpoint Node.

Proof Generation

Each Node provides an HTTP API that is a subset of the Core API. Hashes submitted through this API are aggregated into their own Merkle tree at regular intervals. The Merkle root of that tree is submitted to Core. Each hash sent to Core may be used to generate thousands of proofs per Node. Thus, each Node significantly increases the network's capacity to generate proofs.

Verification

Nodes store a local mirror of the global calendar in real time. This allows Nodes to provide the same proof verification API as Core. Every Chainpoint Node can fully verify every Chainpoint proof.

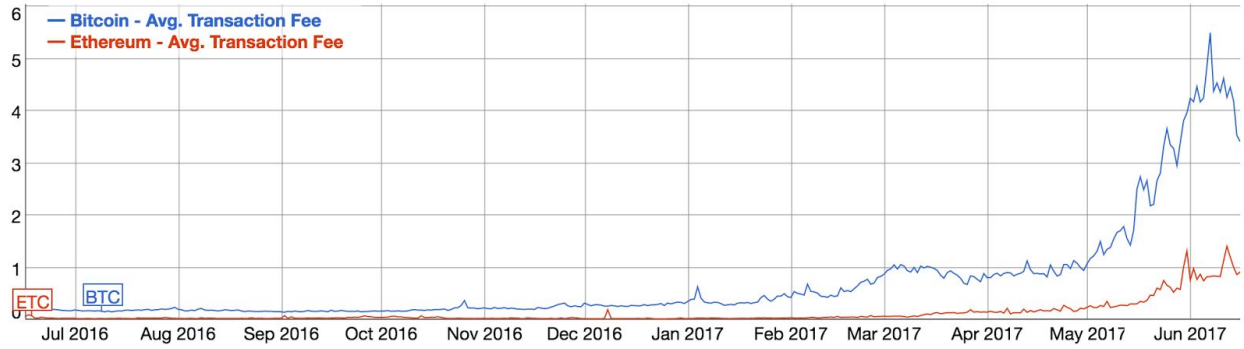
Real-time Calendar Audit

Nodes mirror the calendar data in real-time. Nodes validate that each block in the chain is internally consistent, and signed with the public key of the Core Partner. Periodically, Nodes verify that the entire chain is valid all the way back to the genesis block and report these results to the network.

Economics

When Tierion was first released in June 2015, the cost of anchoring data into the Bitcoin blockchain was approximately $0.03 USD. In June of 2017, that cost has increased over 100x to $3.40 USD. Ethereum transaction fees are following a similar pattern.

Figure 5. Bitcoin and Ethereum transaction fees in $USD.



Rising transaction fees have made it too expensive for individual developers and most businesses to anchor data. Based on current market prices, anchoring one transaction every ten minutes to the Bitcoin and Ethereum blockchains costs $181,332 per year. These costs are projected to continue to rise.

The Tierion Network makes anchoring data economically viable for all. The Chainpoint Service scales to anchor a virtually unlimited amount of data with a minimal footprint on the blockchain.

Tierion Network Token

The Tierion Network Token (TNT) incentivizes network participants to operate and secure the network infrastructure. Chainpoint is the first service available on the Tierion Network. We have plans for future services that will be built on top of Chainpoint and will announce these services in the future.

Core Members

Core Members incur significant costs to operate server clusters. TNT provides a method to recoup these costs. Core Partners earn TNT for anchoring data into the Bitcoin and Ethereum blockchains. Periodically, Core's consensus algorithm elects a leader that can create an anchor block, which requires them to spend BTC or ETH to publish a transaction. The Core Member that creates the anchor block receives a block reward, as well as the tokens paid to Core for that anchor block.

Nodes

Nodes earn TNT by mirroring a copy of the calendar and publishing an API endpoint for proof creation and verification. Nodes are periodically audited to prove they have a current copy of the calendar that can be used to verify a Chainpoint proof. Nodes that pass the audit have a chance to win the reward for that period.

Periodically, nodes spend TNT to send data to Core for anchoring. Each node has a local mechanism for constructing Merkle trees and generating proofs. By sending a Merkle root upstream, each node can create thousands of Chainpoint proofs using a single anchoring transaction.

Nodes may charge for generating and verifying proofs. Node operators can also build services and charge at a price that's independent of the value of TNT.

A fixed supply of TNT will be created during a token sale using the ERC20[19] standard. The token sale and our partner commitments guarantee that for the first year, users will be able to send limited amount of data to Core at zero cost.

Conclusion

As the world relies more and more on digital data, it becomes increasingly important to identify the source of information and verify its authenticity. This is especially pressing for companies in regulated industries, such as healthcare, insurance, and financial services. These organizations are trusted with safeguarding huge volumes of critical data. Proving the authenticity of information within these industries is currently slow, cumbersome, and expensive.

The Tierion Network will fundamentally improve how the world secures and shares data. By providing a “Proof Engine” for the Internet, everyone will be able to prove with absolute certainty, when data was created or if it has been modified from its original state. Additionally, Tierion provides an ultra-secure trust anchor that doesn’t rely on trusted authorities.

The vision for the Tierion Network stretches beyond the initial Chainpoint offering. This is just the first step in a longer journey. We’re building a global network for data verification that operates at Internet scale. Incentives are built into the ecosystem to ensure the network’s reliability, growth, and development. By running a node, developers will improve the network’s scalability and earn tokens which grant access to a growing array of services.

We see the Tierion Network as a solution to the problem of trust on the Internet. The seemingly simple innovation of using hypertext to link HTML documents led to the creation of the World Wide Web. Similarly, linking data to the blockchain will create a global standard for verifiable data. The implications of this seemingly simple innovation could be groundbreaking.

There is a fundamental gap in the Internet’s trust infrastructure. The root of trust for all systems relies on trusted authorities. Tierion closes this gap and makes it possible to create a better Internet where proof replaces trust as the foundation for security.

For more information, visit https://Tierion.com or email info@tierion.com

Appendix A: Full Chainpoint 3.0 Proof Sample

This sample is for demonstration purposes only.


```
á
  íî[gfl]plíÆ í`llhkÆ××oüa\Çgj_×[`Yafhgafl×nüíÄ
  ílqh]íÆ í#`YafhgaflíÄ
  í`Yk`íÆ íþû\YúYZ[úÿùāZ^üĀYûùý^ü\Ăÿÿýþýú^Y\āā\Z\Ăúùúý[\ü]þ^ùþýûZĂā[ùùZĀāĀ[íÄ
  í`Yk`ëa\íÆ íāāþüZúĂùÈÿùÿúÈúú]ĀÈĂüûûÈýþüþýāýĀ]ÿûĂíÄ
  í`Yk`ëkmZeall]\ëYlíÆ íûùúĀÈùĀÈùý4ùûÆüÿÆùĀ:íÄ
  íZjYf[`]kíÆ ß
    á
      ídYZ]díÆ í[YdëYf[`gjëZjYf[`íÄ
      íghkíÆ ß
        á
          ídíÆ íāāþüZúĂùÈÿùÿúÈúú]ĀÈĂüûûÈýþüþýāýĀ]ÿûĂí
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
        á
          ídíÆ
íúýĂĂúüþĀÿùÆāāĂùüÿ[Y[ÿ^ý\Ă\Z^[úüÿĂü\YûYþþā^ÿþ^[ûýÿā\ûÿ]^ü^üĂüýZā\þ]úĂ]āÿ\ĀÿúÿĀĂü]ý\ûý\]Z[[\^ûÿĀ
ý\úĀþ^ÿÿĀûý[\[^þúĂāýùÿ[úĂÿĀ\āüYüþZĂYùù\ü^úû[Zí
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
        á
          ídíÆ íúýùÃÆúýĂĂúüþĀĂúĀûĀÆúÆYÇ[`YafhgaflÇgj_Æ[YdÆúýùĀí
        âÄ
        á
          íjíÆ í]ÿú]]ZZ]ûĂĀ]\ûĀÿ\ûùùùþ\[\úýÿāùþ[úāýÿYýZĀùĂùÿÿ[ýùYúĀÿĂùü^āýýÿýY[]í
        âÄ
        á
          íghíÆ ík`YÈûþÿí
        âÄ
```

```
        á
          íYf[`gjkíÆ ß
            á
              ílqh]íÆ í[YdíÄ
              íYf[`gjëa\íÆ íúýùÃíÄ
              ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úýùÃ×`Yk`í à
            â
          à
        â
    àÄ
    íZjYf[`]kíÆ ßá
        ídYZ]díÆ í]l`ëYf[`gjëZjYf[`íÄ
        íghkíÆ ßá
            ídíÆ íÃû\]ÿ^]þ[ã^üþùþZþãýüÿ[ãÿZãÃ\û[\^Ã^[YüYû]\Zýãþ^ÿÿýû]úãûýÃ[]\^û\ÿãí
          âÄ
          á
            íghíÆ ík`YÈûþÿí
          âÄ
          á
            ídíÆ íùÃùÃÿ\Z[ýÃZÃÃùÃÃúÃ]úüZÃÃþ]Z^ûãÃY[]þùZãÃùýýù^úùýûýY^ããýþ^ZÃÃÿú]Yþí
          âÄ
          á
            íghíÆ ík`YÈûþÿí
          âÄ
          á
            ídíÆ íûãü\ûþþãYýYüZÃ[þÿúüÿÃãý^Ãûúú\]ýûÿüÃZþã[ýúþÃÿ]\ã\ÃÿûÃZ^þýûü[ÃÃÿÿ[í
          âÄ
          á
            íghíÆ ík`YÈûþÿí
          âÄ
          á
            íYf[`gjkíÆ ßá
              ílqh]íÆ í]l`íÄ
              íYf[`gjëa\íÆ í\ü]Ã][ãý[ü\Z]ãÿ^Ã\ÃYã]YÿãY]ý\]\ÿ[ùZùúûZ]þúÃ^ýüüYùÃ^úþZ\ÿúû^ZýÃYÃíÄ
              ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×úýÿÃ×\YlYí à
            âà
          â
        à
```

```
      âÄ
      á
        ídYZ]díÆ íZl[ëYf[`gjëZjYf[`íÄ
        íghkíÆ ßá
            ídíÆ í[Zù\ZZ]\Zþ][þüÿü]üÃZ]Ă^[ýü^þÿ^üûú]úþĀû[^[^üùý\ûÿ^[ÿĀ[Zÿ]Yû]ýĂ^Y^í
          âÄ
          á
            íghíÆ ík`YÈûþÿí
          âÄ
          á
            íjíÆ í[Zù\ZZ]\Zþ][þüÿü]üÃZ]Ă^[ýü^þÿ^üûú]úþĀû[^[^üùý\ûÿ^[ÿĀ[Zÿ]Yû]ýĂ^Y^í
          âÄ
          á
            íghíÆ ík`YÈûþÿí
          âÄ
          á
            ídíÆ
íùúùùùùùùùú\ĂýYĀ^ĂûýÝúüÿYĂþ[]ZĀùZĀ[ÿĀþāZûYÿþ^Ā[[YûZù^Yúýý[Z]Ā[üĂ^ûúĀYùúùùùùùùÿYýĀüùýýùûûù
þùýYýþĀú[ýûÿü[āü\þúüĂĂ]^úýûýùYüZZùÿY^ĀúþĂ^Zÿ\ZZÿ\Zúāû]Ā]ĀĂùú]\^āùûûùûĂýû]ĂāYûù\ûĂþĀþüúþþ[þûýĂYþ
āý^ÿûÿúYÿZüú[]ÿùüĀûùZĀ[üĀ]ù]ĀúZYĀýûùĀùúûúùüþZÿĂùúúýÿĀĂ\ýý\ĀþZĀþYYúĀù]üýþĂÿ[ĂýĀĀā^þāĂZ[ZĂùÿüZù]
ý]ûĂü^[Y[\ú\^^^^^^^^ùûùùùùùùùùùùùùùûÿYûùí
          âÄ
          á
            íjíÆ íĀZù\ü]ùùùùùùùùùúĂĀÿYĂúýĀùùü[[þĂúþ^ÿ[ûü^\þúûZüā\Y]]][^\\]ĀYþāĀ]ĂāāY[ùùùùùùùùí
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            ídíÆ í[YĂĀāúúû[YúZZ\[Y^Y[ûüúZüĂYûü\[ý\YĀāÿ]^^āúýĀ[ý]ĀûZĂāùĀĀāþY^]]ýāZZí
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            íjíÆ íü]ûü]āúÿùùüĂþĂýYüüāĂý^ÿþÿý]úZúüýāZZ\ĀYùùāā\ýû[ýY[ZĀü]]Y]\þĂ[ùùĂ\í
          âÄ
          á
```

```
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íû]Ā\û[ùüYĂþùĀY]ûÿþ][^þZþüþÿāāþYþüüĂüYûùûĂ\ûýúüĂýĂĂĀûÿþYúYûþY]^[ÿí
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íúāY[ü]Āüýü^ùúÿāĂù[þúù]Ăü^Ăüþûÿúúÿ\Ă]ü^þÿþýüÿýûĂāüù^Y^ùĂüý^ý^ā]ýí
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íü^ĀĂZZĀZýüþZùþüûúÿþú\Y]^\üĀý[\[ÿāú\[ùÿ^YYÿþ]üĀý]üāüüĀZāā[Yùýÿ\]Yí
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íúþúÿ^ùùù\]ÿ[^^þ[ā[ÿü]]^ùāú]Z[][ûY\û^\[^Āùüý\Zúÿùýþ\ùûýYĂùüýú]ùĀ\í
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íĀĀ[ÿþýZü\úÿùþ^Āā]\ùĂú[Z\ýûù[ĂüĂ[ü^]^^Ā\þĀ\[üù[úĀú^YýþYþYü[āú^\Ā\í
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
á
  ídíÆ íYûĂûĀāù[[Āýāÿ ĂĀ[ZýĂĂ^\[[ā[ZāĂ\āüþÿùĂ^úú]þùûûāú\^]ü^ÿÿĂùZú[[ûü\[Zí
âÄ
á
  íghíÆ ík`YÈûþÿÈpûí
âÄ
```

```
          á
            ídíÆ í[ZýĂÃùZĂYãĂüÿZZ[úüĀ\\]Zÿ\[YZýÿûùãĂĀZùĂĂYýþù][\[þ^ü]ãÿ]^ýZüYĀúüþ[í
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            ídíÆ íÃãû^^ýYü\ÿù^ãĀýZüúYZþþ\ZþãY\ĂûúĂ^ú]Yýû^ÿããüĂþ]Ăûù\ù]Zýû^þĂúÿãĂĂĀí
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            ídíÆ í^ûĂù]]Yúýýþúü\ÿúããþùûããāýýĀĀþúþùYúĀ]ZĂ\^ÿZāùã[Y[Yû\Āýã]]ý[\ü[ãùùí
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            ídíÆ íùĀã\üüÿÿüĀY[]\[ĀĂĀ\][üü]]]ùYÿĂü^ĀYüüþZ[ÿþüYZ]Züúÿúýþù]ûĂýþþúþZĂ]í
          âÄ
          á
            íghíÆ ík`YÈûþÿÈpûí
          âÄ
          á
            íYf[`gjkíÆ ßá
              ílqh]íÆ íZl[íÄ
              íYf[`gjëa\íÆ íýüýĀùûíÄ
              ímjakíÆ ß í`llhÆ××YÇ[`YafhgaflÇgj_×[Yd]f\Yj×ûÿýû×\YlYí à
            âà
          â
        à
      â
    à
  âà
â
```

Endnotes

1. W. Vaughan. Chainpoint: a standard blockchain proof protocol, (2016)
https://medium.com/@WayneVaughan/chainpoint-a-standard-blockchain-proof-protocol-79def1c37189

2. JSON For Linking Data, JSON-LD (2017)
https://json-ld.org/

3. “Chainpoint Community Group”, W3C, (2016)
https://www.w3.org/community/chainpoint/

4. Web consortium weighs in on blockchain standards, Fedscoop, (2017)
https://www.fedscoop.com/web-consortium-starts-work-on-blockchain-standards/

5. S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, (2009)
https://bitcoin.org/bitcoin.pdf.

6. M. Aráoz. What is Proof of existence?, (2014)
https://web.archive.org/web/20130711050552/http://www.proofofexistence.com:80/about

7. “Transaction Rate”, Blockchain.info, (2017)
https://blockchain.info/charts/transactions-per-second?daysAverageString=7&timespan=all

8. “Block timestamp”, Bitcoin Wiki, (2017)
https://en.bitcoin.it/wiki/Block_timestamp

9. “Microsoft and Tierion collaborate on attestations & Blockchain proofs”, Microsoft, (2017)
https://azure.microsoft.com/en-us/blog/blockchain-identity-proofs/

10. Trust Anchor, Wikipedia (2017)
https://en.wikipedia.org/wiki/Trust_anchor

11. NTP: The Network Time Protocol, NTP.org (2017)
http://www.ntp.org

12. A Universally Unique IDentifier (UUID) URN Namespace, IETF.org (2017)
https://www.ietf.org/rfc/rfc4122.txt

13. Chainpoint – Innovations in Blockchain Timestamp Proofs, Tierion (2017)
https://tierion.com/blog/chainpoint-innovations-in-blockchain-timestamp-proofs/

14. How to Time-Stamp a Digital Document, anf.es (2017)
https://www.anf.es/pdf/Haber_Stornetta.pdf

15. NIST Randomness Beacon, NIST (2017)
https://www.nist.gov/programs-projects/nist-randomness-beacon

16. Hash Calendar, Wikipedia, (2017)
https://en.wikipedia.org/wiki/Hash_calendar

17. CockroachDB, Cockroach Labs (2017)
https://www.cockroachlabs.com

18. Consul, Hashicorp (2017)
https://www.consul.io

19. F. Vogelsteller, "ERC 20 token standard", (2015)
https://github.com/ethereum/EIPs/issues/20

Acknowledgements

The development of Chainpoint has been a collaboration of many. The authors would like to recognize the following individuals for their contributions.

Zaki Manianↄ—Co-Founder, Skuchain
Zaki helped architect and code Chainpoint 3.0.

Eder Santanaↄ— Developer & AI Researcher
Eder researched competing technologies, helped develop Chainpoint 3.0, and was the primary developer of Verifai.

Shawn Wilkinsonↄ—ↄFounder & CEO, Storj
Shawn made early contributions to the first version of Chainpoint.

Manu Spornyↄ—ↄFounder and CEO, Digital Bazaar; Creator of JSON-LD
Manu assisted our team with the use of JSON-LD in Chainpoint 2.0. He also helped facilitate discussions of the Chainpoint protocol at several industry standards events.

Christopher Allenↄ—ↄPrincipal Architect, Blockstream; Co-author of SSL/TLS
Christopher runs the excellent Rebooting Web of Trust events where Chainpoint 2.0 received peer review. Christopher contributed to the Chainpoint 2.0 protocol and has been a supporter of industry standards around Chainpoint and similar technologies.

Ryan Sheaↄ—ↄCo-Founder & CEO of Blockstack
Ryan reviewed and contributed to the Chainpoint 2.0 protocol.

Jude Nelsonↄ—ↄEngineering Partner, Blockstack
Jude reviewed and contributed to the Chainpoint 2.0 protocol.

Paul Sztorcↄ—ↄVP of Economics, Bloq
Paul reviewed and contributed to the Chainpoint 2.0 protocol