SCTE·ISBE CABLE-TEC EXPO 2017
DENVER, CO
OCTOBER 17-20

2017 Fall Technical Forum
SCTE·ISBE • NCTA • CABLELABS

A Simple Overview of Blockchains

Why They Are Important to the Cable Industry

A Technical Paper prepared for SCTE/ISBE by

Steve Goeringer
Principal Architect
CableLabs
858 Coal Creek Circle
Louisville, CO 80027
s.goeringer@cablelabs.com

Table of Contents

List of Figures

Introduction

Blockchain technology has rapidly become one of the most discussed and visible emerging technologies. Gartner's 2016 Hype Cycle for Emerging Technologies shows blockchain near the peak of inflated expectations while technology mainstream adoption is still likely to be 5-10 years out. Other technologists and analysts have hyped blockchain even further, claiming it to be the most significant technological innovation since the internet. Recently, many researchers have started to consider whether blockchains can be applied to improving IoT Security or services. What is a blockchain? How is it transformational? This paper provides a quick primer into what blockchains are and why they have the potential to be uniquely valuable to cable network operators. The first part of the paper reviews the basics of how blockchains work. This is followed by a discussion of blockchain features and requirements that are relevant to network operators. The paper concludes by asking two key questions that will aid readers to find their own killer applications.

So… What is a Blockchain?

What is a blockchain? It's hard to find a simple definition that doesn't relate to a distributed database or contain a reference to Bitcoin. Perhaps a simplistic but concise definition is that a blockchain is an immutable, distributed ledger visible to the community implementing and using the blockchain. Immutable means that the information a blockchain contains cannot be changed. Distributed means that the information is replicated amongst many participants (in Bitcoin terms, nodes). Ledger implies that the blockchain records transactions. Visible to the community means that every transaction recorded in the ledger is visible to every participant – user or implementer – of the blockchain.

Another definition, closer to what a computer scientist might appreciate, is that blockchains are a method used to create securely linked lists of transactions. Secure in this context means cryptographically protected (authenticated and signed) and distributed amongst stakeholders. The list of transactions is linked by using a hash of a collection of transactions (a block) in the next collection of transactions. Fault tolerance against failure, including malicious or negligent actions of stake holders, is achieved using a consensus protocol (achieving a security goal referred to as Byzantine fault tolerance) [Lamport][Castro].

However, unless you already had a firm grasp of what a blockchain is, really, these definitions probably didn't really provide much insight. Let's come back to building that insight in a moment.

Some of the hype also talks about distributed ledgers. The phrases blockchain and distributed ledger are often used synonymously. In many ways, the term distributed ledger is more descriptive than blockchains. Perhaps it would be useful to think of blockchains as the technology and distributed ledgers the result of using blockchains (e.g., blockchains create distributed ledgers). However, in common usage the terms are used interchangeably.

Let's talk about why you might want to even care about blockchains or distributed ledgers. There is huge hype about why blockchains are important. Marco Iansiti and Karim Lahkhani describe blockchains as a foundational technology. They also think it will take quite a while for it to achieve its transformational process. [HRB] Alex Tapscott describes blockchain as the "next generation of the internet." He goes further, "The blockchain is the internet of value." [Miller]

The "big deal" is that we've never had a capability to create a distributed public history and make it available to the blockchain participants. Through the power of cryptography, we can create a secure

history of transactions that is much more expensive to change than it is to create, and in fact is practically impossible to change. And we can do so in a way that makes all those transactions visible to all the participants in a given network (or not – visibility is a choice). This makes blockchains uniquely valuable.

Why is an unchangeable, public, distributed ledger valuable? Some applications that have been discussed for blockchain include digital currencies, recording of real estate transactions, and registration of marriage licenses. Digital currencies can be used in countries experiencing hyper-inflation. Blockchains can be used to record public transactions and prevent corrupt officials from changing transactions illicitly after the fact. These are just a couple of examples of blockchain applications.

How Do Blockchains Work?

There are many blockchain implementations – Bitcoin, Ethereum, Hyperledger (a Linux Foundation Project), Multichain, BigChainDB. A comparison of several is provided by a companion paper [Hintzman]. They all differ in important details. However, the general notion is illustrated in Figure 1.

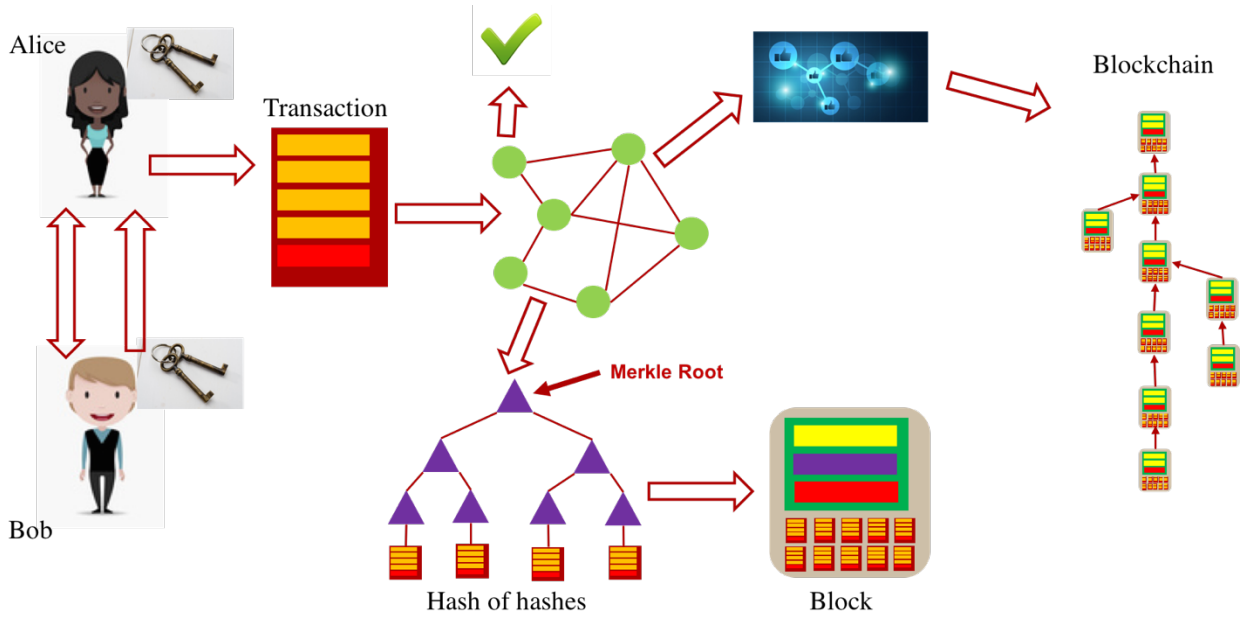

Figure 1 – How blockchains work

Alice and Bob want to record a transaction – perhaps Alice is buying something from Bob using a digital currency, or perhaps there is some event they are trying to permanently record. This explanation is use case neutral. Both Alice and Bob have created or been assigned public-private key pairs that will be used to attest this transaction between them. Generally, the transaction will be from Bob's public key to Alice's public key. (Readers are referred to Wikipedia's entry on public key infrastructure for further details. [PKI]) After they negotiate the terms of their transaction, Bob provided his public key to Alice. Alice creates and sends a transaction to Bob using his public key and signs the transaction using her private key. (In reality, Alice uses an application to do all this. Digital currencies call this application a wallet.)

Alice submits that transaction to one or more nodes in a blockchain. The nodes comprise a network and Alice's transaction may be submitted to one, many, or all nodes. For purpose of this discussion, the

elements that participate in the blockchain network will be referred to as nodes. Each node that receives Alice's transaction will validate the transaction according to some criteria (for example, authentication using public key infrastructure (PKI) or comparison of information in the transaction to a policy or list). Nodes will add valid transactions to stack or queue of transactions.

At some point, the collections of transactions in the queue get processed at each of the nodes. Usually, this is triggered by a time interval but other criteria are possible. First, the transactions are hashed (this might have been done when the transaction was generated by Alice). Hashing is a mathematical function (often referred to as a trap door) that computes a value that cannot be easily reversed [Diffie]. Trap door functions ensure it is hard to determine the original input given just the output of the hash function. These hashes are then aggregated and hashed again, producing a hash of hashes (such as a Merkle root, which use a tree structure as shown in Figure 1) [Merkle]. The transactions, the hash of hashes, a link to the immediately previously produced block (usually the hash of that previous block), and other information are encoded into a block. Proof-of-work may be performed on this block [Jakobsson]. (Proof of work requires application of computer resources to solve a problem, usually a mathematical computation, as an economic measure to discourage system misuse such as denial of service.) And, of course, this block is hashed.

The next step has the greatest variation amongst the different blockchain implementations. One or more blocks from all the nodes in the network need to be added to the blockchain (distributed ledger). All the blockchain network nodes that successfully create a block in time (systems that use proof of work have uncertainty) have a chance of having their block added. A consensus protocol and process is applied to select the block (or blocks) [Fischer]. This might be through voting, or by proof of work, or proof of stake, or some other scheme. Proof of stake consensus has block selection conducted using an a priori deterministic selection of blocks based on the stake (ownership or possession) a given submitter has in the blockchain. [POS] The goal of the consensus process is to make it hard for one of more nodes to compromise the overall, long term integrity of the blockchain. In some systems, the consensus process may allow more than one blockchain to exist at a time while consensus is still "debated". Eventually, however, the network should converge to a single chain.

And then the process starts over. On current blockchains, iteration is assumed to occur for eternity.

What Do Blockchains Achieve?

What is the result of the blockchain as described above? There are several achievements worth noting. Alice, Bob, and everybody else can see that a transaction occurred between their identities. If those identities are anonymous, then the transaction is visible but nobody knows the actual actors involved (see Figure 2). It's important to keep in mind that anonymity is a choice. Also, information contained in the blockchain can be encrypted or actually just be a hash of information that is stored elsewhere known to at least Alice and Bob.

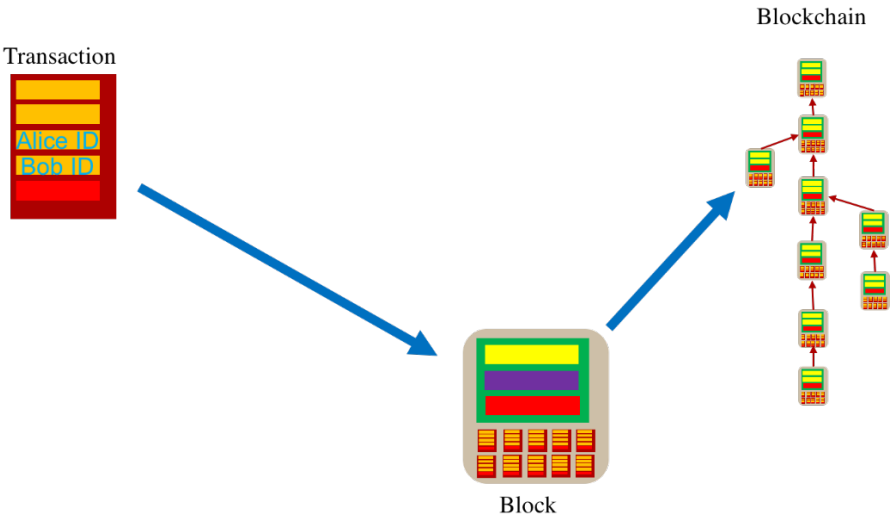

Figure 2 – Blockchain visibility

We've also achieved layers of integrity as shown in Figure 3. The integrity of that transaction within the blockchain is verifiable by Alice's digital signature and the Merkle root in the block in which it saved. This is probably the real big deal about blockchains – we've created a verifiable history that is computationally challenging to change.

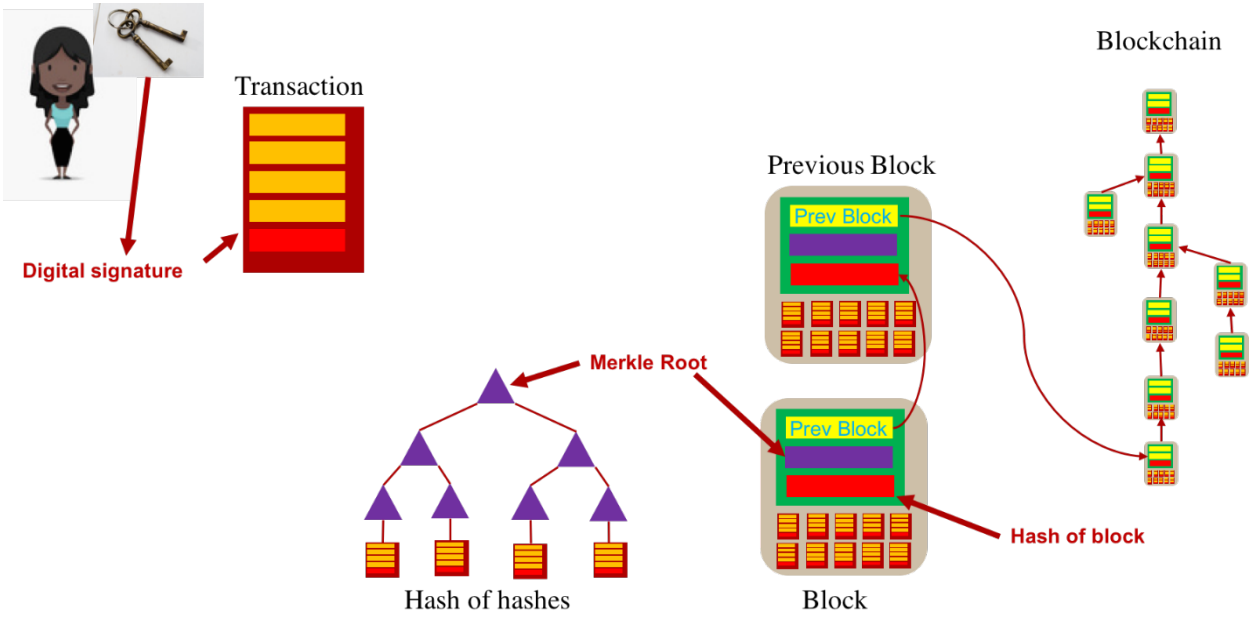

Figure 3 – Blockchains provide multiple layers of integrity

There are more features that further protect the integrity of the transaction history. Several nodes achieved consensus (at least 51%) on the validity of the blockchain after the new block was added before their proposed blockchain was accepted. This protects against malicious nodes and against malfunctioning nodes (in other words, Byzantine fault tolerance). As long as the consensus pool is large enough, the

nodes don't to have to trust each other – they just assume more nodes are trustworthy than aren't. The resulting distributed ledger is replicated (usually in whole, but in part is possible) across many nodes assuring availability and also making it hard to change the distributed ledger. Finally, the integrity of the chain itself is verifiable by checking the chain of signatures from the genesis of the ledger all the way to the current block. See Figure 4.

The result is a linked list of transactions that are visible to blockchain participants, verifiable, and unchangeable. The transaction occurred and the ledger entry for that transaction on the blockchain cannot be changed. Consequently, there is no longer any need for participants in a blockchain to need to trust each other with regards to the nature of that transaction – the existence of, and the contents of the transaction can be treated as fact. A major misconception about blockchains is that they provide a basis of trust. A better perspective is that blockchains eliminate the need for trust.

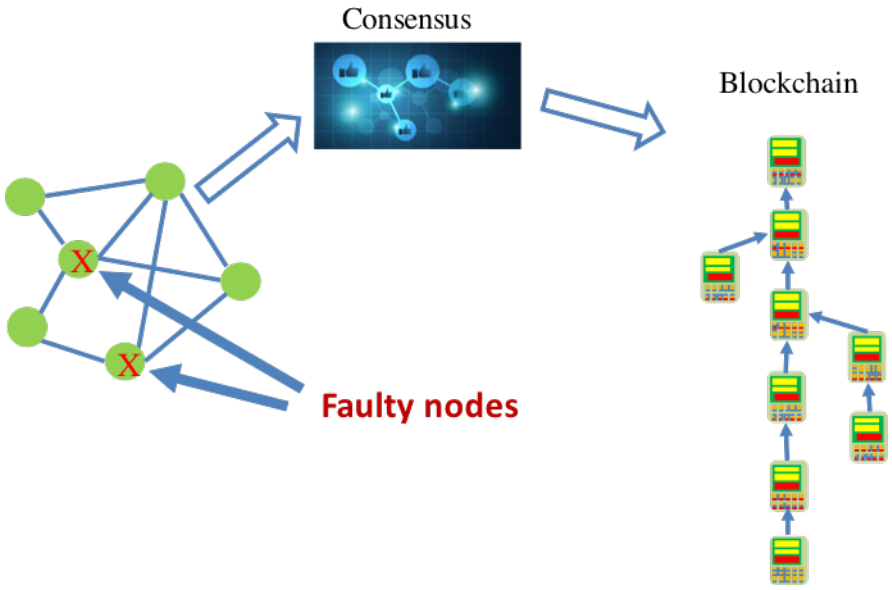

Figure 4 – Blockchain consensus

Smart Contracts

One of the values not discussed above is that the transaction submitted by Alice may include executable code, or a script. This concept is loosely described as a "smart contract" [Stark]. This is an area still in its infancy – it still must be proven that small code snippets imbedded in blockchain transactions can really be secured against misuse. Assuming they are secure, smart contracts could be transformational by creating programmable currencies and transactions that execute automatically according to the conditions included in the contract. Conditions of the contract are programmed using a constrained and highly secure programming language. Only an authorized part (verified usually by possessing a private key) can execute the code.

So, in our blockchain example, the contract gets encoded by Alice. She includes it in her transaction (which is signed by her, so we know the contract is valid). See Figure 5. Then execution of the transaction can be conditional. Some examples:

- The transaction is not valid unless the recipient can respond correctly to a cryptographic challenge which is verified by the smart contract.
- The smart contract contains a counter that is decremented each time the contract is accessed and when the counter is zero, the transaction becomes unusable.

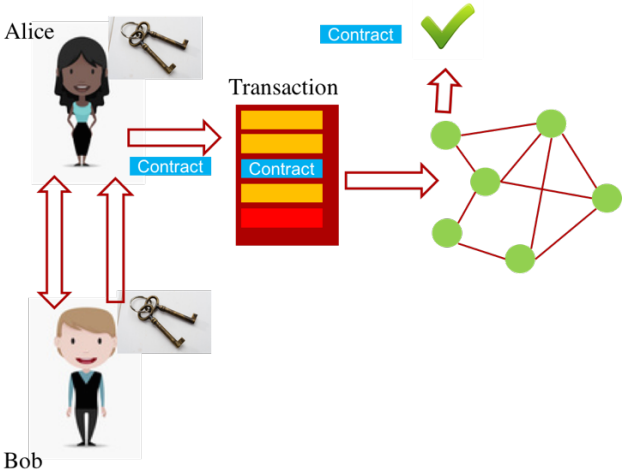

Figure 5 – Smart contracts using blockchain

How Can Cable Use Blockchains?

Now that we know what blockchains are, let's consider how they apply towards the cable industry. Blockchains can be used as platforms for orchestrating ecosystem-secure transactions. They convey transparency and visibility. They are immutable. And they are transaction-oriented. Given those strengths, it would seem that they might be widely applicable.

However, there are some factors that should be evident from the example above to consider first. For blockchains to be useful, they rely on relatively strong cryptography which is computationally intensive. In network engineering terms, this means transaction processing uses lots of power and may be relatively slow. Furthermore, they are distributed which means that information is stored very redundantly – perhaps at thousands of nodes. In fact, in most blockchain implementations, every node contains every transaction. And the size of that storage increases linearly over time per node, and geometrically across the network (because every node contains all the data). And finally, behind the consensus algorithm is the need for a community of actors to work together to implement the blockchain. This ultimately means governance of the code base and network participation terms. In summary, blockchains are resource intensive in terms of compute, storage, and networking and the stakeholders that implement the chain must be willing to work together.

Given that background, the transformational potential of blockchains can be hard to realize. There have been many use cases postulated that ultimately use blockchains as a secure database. Blockchains are very inefficient for data storage and data retrieval and all the public chains actually have databases used to present information from the blockchain (rather than running queries against the blockchain itself). Many of these use cases, however, appear to have worked very well. However, the benefits may have had more

to do with application programming interfaces that were optimized for the specific needs of these particular use cases.

How then, can we identify good use cases for the cable industry where blockchains may work well or use cases that might be transformation to our industry? The goal is to find opportunities that dramatically impact cable, and identify areas where we can reduce friction, speed time to market, and remove the need for trust. Considering the following two questions are helpful:

- *Can we use blockchains as platforms for digital transformation of the cable user experience?*
- *Can blockchains enable dynamically social user experiences for cable subscribers that mirror the sharing economy?*

Use cases that satisfy these questions are still under investigation. Three areas seem potentially very attractive: improving trust in content distribution, streamlining complex service delivery in the medical industry by leveraging cable, and providing improved secure digital content production and distribution. Evaluation of specific use cases should apply formal methods to determine blockchain applicability [Scriber].

A Blockchain for the Cable Industry

It may be beneficial to leverage public or other industry chains for some cable industry use cases. However, the cable industry might be well served by one or more industry-specific blockchains. Such blockchains can be designed to meet specific security, performance, and scalability requirements appropriate to regulated products that serve markets of millions of subscribers. Moreover, governance can be left wholly in the control of the cable industry stakeholders without compromise with the best interests of other sectors or parties. Finally, a cable industry blockchain may prove to be more economic in the longer term. Leveraging existing chains may involve transactions fees, integration and consulting costs, and features that provide little value to cable use cases while dramatically increasing processing and storage requirements.

This should be very practical because of the following: Given how many dozens of blockchains already exist, it is clear there is no specific technical or economic hurdle that prevents creation of an industry blockchain. The critical design decision issues appear to be how governance will be executed and maintained and, under that governance, who will implement the blockchains.

Can a single company benefit internally from a blockchain? Perhaps. If trust management between company elements is challenging, the visibility and immutability of a blockchain may prove useful. However, using a blockchain may appear easier in implementation than traditional database or transaction-logging mechanisms. As mentioned previously, this may largely be due to more modern, or more simply engineered application programming interfaces that are particularly easy to apply to the intended use cases. It must be remembered that maintenance of a blockchain requires significant processing and storage resources. Moreover, the rigid linked-list structure of a blockchain is not well suited for efficient searching or indexing. It seems that the best use cases must have the scope of an entire ecosystem for blockchains to serve their intended purpose in an efficient manner.

Conclusion

This paper has provided a quick, practical overview of the basic concepts of how blockchains work and some suggested use-cases that may be of interest to cable operators. Given that background, the paper has suggested that there may even be transformational use cases that are yet to be discovered, that are very applicable to the cable industry. However, the benefit of blockchains to those use cases must be carefully considered because blockchains can be inefficient and/or expensive. Where blockchains seem to fit well for cable services and applications, we may find that public or otherwise commercially-enabled chains may not be the best implementation path for cable operators. Rather, the cable industry should consider implementation of one or more dedicated blockchains.

Abbreviations

ISBE	International Society of Broadband Experts
SCTE	Society of Cable Telecommunications Engineers

Bibliography & References

[HRB] The Truth About Blockchain. Marco Iansiti and Karim Lakhani. Harvard Business Review. January-February 2017. Online. Downloaded July 11, 2017. https://hbr.org/2017/01/the-truth-about-blockchain.

[Miller] Alex Tapscott on why the future of finance will be on blockchain. Podcast. Zack Miller. Tearsheet. July 22, 2016. Online. Downloaded July 11, 2017. http://www.tearsheet.co/digital-currency/podcast-alex-tapscott-on-why-the-future-of-finance-will-be-on-blockchain.

[Stark] Making Sense of Blockchain Smart Contracts. Coindesk. June 2016. Online. Downloaded July 12, 2017. http://www.coindesk.com/making-sense-smart-contracts/.

[Lamport] The Byzantine Generals Problem. Leslie Lamport, Robert Shostak, Marshal Pease. ACM Transactions on Programming Languages and Systems. July 1982. Online. Downloaded July 12, 2017. http://dl.acm.org/citation.cfm?doid=357172.357176&CFID=784828639&CFTOKEN=84236530.

[Castro] Practical Byzantine Fault Tolerance. Miguel Castro and Barbara Liskov. Proceedings of the Third Symposium on Operating Systems Design and Implementation. February 1999. Online. Downloaded July 12, 2017. http://pmg.csail.mit.edu/papers/osdi99.pdf.

[Hintzman] Comparing Blockchain Implementations. Zane Hintzman. Cable-Tec Expo 2017. October 2017.

[PKI] Public key infrastructure. Wikipedia. Online. Downloaded July 17, 2017. https://en.wikipedia.org/wiki/Public_key_infrastructure.

[Diffie] New Directions in Cryptography. Whitfield Diffie and Martin Hellman. IEEE Transactions in Information Theory. November 1976. Online. Downloaded July 12, 2017. http://www-ee.stanford.edu/~hellman/publications/24.pdf.

[Merkle] Comments in 2012 about the 1979 paper: A Certified Digital Signature. Ralph Merkle. Online. Downloaded July 12, 2017. http://www.merkle.com/papers/Certified1979.pdf.

[POS] Proof of Stake. Bitcoin wiki. Online. Downloaded July 17, 2017. https://en.bitcoin.it/wiki/Proof_of_Stake.

[Jakobsson] Proofs of Work and Bread Pudding Protocols. Markus Jacobsson and Ari Juels. Communications and Multimedia Security. Kluwer Academic Publishers. 1999. Extended Abstract Online. Downloaded July 12, 2017. https://www.emc.com/emc-plus/rsa-labs/ps/breadpudding.ps.

[Fischer] The Consensus Problem in Unreliable Distributed Systems (A Brief Survey). Michael Fischer. International Conference on Foundations of Computation Theory. August 1983. Online. Downloaded July 12, 2017. http://zoo.cs.yale.edu/classes/cs426/2012/bib/fischer83consensus.pdf.

[Scriber] A Framework for Determining Blockchain Applicability: Blockchain Architectural Fit. Brian Scriber. IEEE Software Magazine. 2017. Publication pending at time this article was written.

