Blockchain based Resource Management System

Saurav Chakraborty
University of South Florida, sauravc@mail.usf.edu

Kaushik Dutta
University of South Florida, duttak@usf.edu

Donald J. Berndt
University of South Florida, dberndt@usf.edu

Recently blockchain has emerged as a distributed ledger based platform, which has gained repute in the domain of cryptocurrency technology. The non-modifiable aspect of blockchain makes it a candidate to store the absolute truth that cannot be disputed. Additionally the distributed nature of blockchain allows all participants of blockchain to have equal control on the system without any single party control. In this paper we develop a blockchain based resource management system, where resources are controlled by its owners and the right to access resources can be controlled at its finest granularity. We devise a general protocol system, which provides a framework for using Blockchain as a platform for managing access control of resources. As an example, we take online education certificates, where students receive various certificates from multiple educational institutes. The student can control the access to these certificates to employers. In this research we present the architecture of this system using blockchain that will ensure the authenticity of the resources and also allow the fine grained access control to these certificates resources.

Key words: Blockchain, Resource Management, Access Control

1. Introduction

In recent years the cryptocurrency technology bitcoin (Nakamoto 2009) has become popular. The global market of bitcoin has grown from less than 200 USD during its inception in year 2008 to 12.5 billion USD in year 2014 (Kaminski 2014). With the increased popularity of bitcoins, the underlying technology, blockchain, has also got the attention of industry leaders. The blockchain is a peer to peer technology that can be used to store the

"truth", that can't be modified, tampered and updated in future (Kishigami et al. 2015). The use of blockchain in the financial domain to manage financial ledgers and transactions is well-known (Swan 2015). Recently we also saw innovative use of blockchain in other non-financial fields such as insurance (Chain 2014) , healthcare (Yue et al. 2016) and patent management (Shanahan 2016). The use-case in all these have been to manage the ground truth. However, the capability of blockchain technology can go beyond just managing the ground-truth. In this paper, we present an application of block-chain to manage digital resources.

Digital resources refer to any resource that is the product of some activity and is owned by a person or an organization online. The digital resource can be shared with other entities in a system (such as another person or another organization) and can be used by them. Examples of the digital resource include but are not limited to social media profile, personal certificates (such as graduation certificate), credit card information and financial documents. These are owned by an organization or a person and can be shared with other parties to be used in business transactions. For example, the social media profile of an individual is shared and used by marketing companies for their marketing campaign. The personal certificates are used to gain access to higher education and job applications. Financial documents (such as bank statements) are used for business transactions (such as mortgage lending). However, in present days these documents are shared across the organizations in a very crude manner such as encrypted email or file upload over a secure channel. The issue with such sharing is once the document is given to an organization by an owner, further sharing of the document can't be controlled by the owner anymore. The document goes out of control of the original owner and is shared across multiple parties without the knowledge of the original owner. For example, an owner provides the pay

statement to a mortgage company for the purpose of the mortgage application. However, once the pay statement is at the mortgage company, the mortgage company can share the statement with other partners financial institution (such as a credit company)for future marketing and credit assessment purpose. The owner will not know such use of the document (pay-slip) and will not have any control over any further use. The owner relies on the good-faith of the mortgage company and existing laws, which in most cases is not enough in protecting the personal information.

In this research, ***we design a set of operations and systems to manage and control access to such digital resources using blockchain technology.*** With our proposed system, the owner maintains control over the access of the resource. Even though the resource is shared by the owner with another entity, the resource can't be further shared by that organization to another third-party. If a third-party needs the access to the resource, the third-party needs to get the permission from the original owner. The owner and only the owner fully controls who can access these documents. As described before there exists multiple use-cases for such a system. We pick once such use case, educational sector, and describe how blockchain can help us achieve educational certificate management across multiple organizations.

In Section 2 we provide information related to the blockchain as a technology and research that is pertinent to the domain. In Section 3 we describe a set of basic operations using blockchain for resource management purpose. In Section 4, we describe an example application. Subsequently, we use this example application in Section 5 to demonstrate how the digital resource management can be achieved by combining blockchain and encryption technology using the operations described in Section 3. Lastly in Section 6 we summarize our contribution.

2. Related Work

The idea of a cryptographically secured chain of blocks was first described in 1991 by Haber (Haber and Stornetta 1990). Szabo (Szabo 1997) also worked on a decentralized digital currency that was referred to as "bit gold". Pilkington (Pilkington 2015) provides a broad and thorough overview of the technology. However, the idea of a blockchain was first conceptualized in the famous white paper issued by Satoshi Nakamoto in 2008 (Nakamoto 2009). It was then implemented the following year giving rise to the first successful worldwide cryptocurrency called *bitcoin*. More recently, blockchain has become a more prominent technology and has seen usage beyond the financial domain (Kakavand and Kost De Sevres 2016, Crosby et al. 2016).

One of the major reasons for the success of bitcoin can be attributed to the security aspect associated with the cryptocurrency. Kosba (Kosba et al. 2016), Heilman (Heilman et al. 2016) and Kogias (Kogias et al. 2016) show that the underlying blockchain technology of bitcoin allows distrustful partners to collaborate without the help of a mutually trusted third party. Swan (Swan 2015) and Crosby (Crosby et al. 2016) showed that this decentralized peer-to-peer network can have far-reaching applications and need not be limited to the financial applications that were utilizing it. Back (Back et al. 2014) enabled bitcoins and other ledger assets to be transferred between multiple blockchains. Ali (Ali et al. 2016) and Kishigami (Kishigami et al. 2015) showcase how blockchain can be used as a platform for different kinds of distributed digital content management.

Needham (Needham and Schroeder 1978) showed how encryption could be used for authentication networks and Datta (Datta et al. 2010) demonstrated how encryption could help maintain privacy in social media. We take these studies and extend them by using blockchain as a tool to augment the privacy of participants in this system and ensure access

privileges are maintained. Zyskind (Zyskind et al. 2015) described and conceptualized a decentralized personal data management system which again makes use of a decentralized network of peers to ensure users own and control their personal data. They discussed the protocols required to give the users control over their data. Our present research falls in similar line. Zyskind (Zyskind et al. 2015) did not describe the implementation of the system using a specific blockchain technology, that makes it difficult to implement their approach in a real life blockchain based system. Whereas in this paper we propose the management of digital resources using a specific blockchain technology (multichain **??**), that makes our approach immediately implementable and applicable.

3. A Generic Set of Protocols for Resource Management using Blockchain

In this section, we discuss a set of resources and operations which are used to manage resources using blockchain. First and foremost to demonstrate these processes, we identify various entities involved in such a system.

- *Hosts* The entity which will host the resource in the blockchain. This user participates in the blockchain by contributing one or more nodes in the blockchain.
- *Resource Owner* The user, whose actions will lead to the creation of a resource and who seeks to control access to this resource.
- *Third User* The user, who seeks access to the resource created and intends to verify the ownership of the said resource.

An user of the system needs to first create a profile of himself. He registers and provides his personal information to create the profile. This information is hashed to generate his digital signature, and public and private keys, which the user will be needing in the future.

For managing the information in the blockchain, we make use of a mechanism called *Streams*. A Stream is a dedicated ledger for certain kind of transactions. A Stream, for

example, will be dedicated to managing the shared key associated with the signature of any particular user. In the resource management system, we make use of four streams S_1, S_2, S_3 and S_4.

Dependency: A User u registers and creates a profile for himself;
Result: signature of the user is generated along with the private and public key of the user

1 A user u creates profile after registration;
2 The signature sig_u of the user u is generated from hash of his profile data;
3 Public and private key of the user u is generated from the signature, the private key pk_u is sent to the user;
4 The signature sig_u, along with the public key of the user sk_u is published to a Stream $S1$;

Algorithm 1: Operations for Signature and Key Generation

Once a user u is registered, his signature and the shared key is released to the public via Stream S_1, but only he has knowledge of his private key. Now once this user has a profile, he can perform the required task which can lead to the creation of the resource. For example, in the use case discussed later in this paper, the user (student) needs to complete a course and pass the required examinations to get the course completion certificate generated. In this case, the course completion certificate is the resource.

Dependency 1: User u registers and creates a profile for himself;
Dependency 2: User u takes action leading to resource (i.e. document) being generated/created;
Result: Signature of the resource is generated and shared with the owner

1 Signature of the resource is created by hashing the resource content d_{uc};
2 The shared key and private key, ($pk_{d_{uc}}$ and $sk_{d_{uc}}$), are generated from the signature of the resource d_{uc};
3 The signature of the resource d_{uc} is encrypted with the shared key of the resource and is published to S_2 along with the signature of the resource owner sig_u;
4 The private key of the document $pk_{d_{uc}}$ is encrypted with the public key of the resource owner, and encrypted private key of the document is published to stream S_3 along with the id of the resource owner ;

Algorithm 2: Operations for Resource Creation and Sharing with Owner

Now once the resource has been created and the original owner has access to it, he can share it with the other registered users. Suppose the owner wants to give another user full access to the resource, then he will be able to do so by making use of streams and this person will be able to get access to the said resource.

Dependency 1: Users u_1 and u_2 register and create a profile for themselves;
Dependency 2: User u_1's action leads to resource being generated/created;
Result: User u_2 is able to get access to the resource d_{uc} owner by the user u_1

1 User u_1 encrypts the combination of private key of the resource and his own signature with the shared key of the user u_2, this we will refer to as the access key;
2 This access key along with the signature of the user u_2 is published to the Stream S_4 by user u1;
3 The user u_2 using his own private key $pk_(u_2)$ is able to retrieve this access key by decrypting the access key published on Stream S_4;
4 The private key of the resource $pk_{d_{uc}}$, is retrieved from the access ke by user u_2. The private key of the resource $pk_{d_{uc}}$ is then used to retrieve the resource stored in Stream S_2 ;

Algorithm 3: Operations for Resource Sharing with another user

If the resource owner does not want to provide full access but only limited access (such as for certain time), then he can do so as well. In this case, the resource owner can take the original resource and use various hashing algorithms to generate a hash which is time-variant and can only be viewed for a period of time. Additionally, the resource can be hosted so that it cannot be downloaded but it can only be viewed after decryption. When the resource owner needs to provide only limited access to the resource, the resource owner can employ these mechanisms to ensure the user can not share the document to a third party.

4. Application of Blockchain in Managing Digital Resources - An Example Use-case

In this section, we design a course and certification management system as a use case for resource management using blockchain. In present days, people are earning online educational certificates along with traditional certificates and degrees from a broad range of accredited and non-accredited educational and industrial organizations. For example, a student (say Joe) may complete a course at Udemy.com, an online course at Harvard Business School, a certificate from Oracle and a degree from the University of Texas. When Joe applies for a job, he mentions the completion of these certificates to his employer. To

verify the completion of these courses and certificates, the employer must contact the individual organizations for the details, along with Joes permission to access his record. This is cumbersome. Additionally, many new organizations such as Udemy and Coursera may not have an established way to verify the record. These organizations focus on the delivery of content to a large and widely dispersed audience, and less on providing verified certificates of completion. LinkedIn is becoming a de facto platform to share such documentation and professional portfolios. The LinkedIn portfolio of Joe may contain these certificates and other courses completed by Joe. However, a prospective employer has no easy way to verify the authenticity of these certificates. This motivated us to propose a blockchain-based system to address the problem as mentioned above without relying on a single authority or platform. Additionally, in the present world, when a set of educational certificates of Joe are provided to an organization (say a recruiting company), the recruiting company can forward these certificates to potential employers without proper permission of the Joe. Through our blockchain based system, we want to give total control of these certificates to Joe. If a recruiting company needs to access certificates of Joe, Joe has to provide the access right to the recruiting company. However, the recruiting company can't forward the access-right to another organization (prospective employer). In our system, potential employers requiring the access to Joe's certificates will need the access right from Joe. To describe our system, we first identify the entities in the system and describe the use cases.

- *Course providers* host the course management, enroll students and keep the student records in a database referred to as the student record database (SRD).
- *Students* enrolls in a course, fulfills the course requirements and earns a certificate of completion. The certificate is written into the SRD by the instructor.
- *Instructors* Once a student has completed all the requirements of a course, the instructor can provide the certificate to the student and update the SRD.

- *Employers* With the permission granted by a student, an employer can access the student record from the SRD and verify the authenticity of completed courses.

If a single entity manages all the courses, i.e. there exists only one course provider, the course provider could manage the SRD containing the students' identities and certificates. However, with the presence of multiple course providers (such as Udemy, Coursera, Harvard and Oracle), the centralized control of the SRD becomes a challenge. It will be difficult for the individual course providers to agree upon a single SRD system and conform to it. More importantly, do we even want a single authority (or small group) to control this information?

To solve the above issues, we propose a blockchain-based approach, where the multiple course providers participate in a blockchain network to store and manage student records in a peer-to-peer fashion. Each participant creates and controls the information under their purview, but no single authority can run over the privacy of others.

In the next section, we describe the high-level design of a blockchain-based system for managing this simplified educational ecosystem. Blockchain can be considered a type of peer-to-peer database with no single administrator. All the participants in a blockchain participate on equal terms, with authority over their own information, which makes the blockchain an attractive technology for developing the above application.

4.1. Blockchain-Based Application Architecture

In Figure 1 we depict an architecture of the proposed course management system using blockchain. In the proposed course management system, there is no single entity managing the authenticity of the certificates. Rather it is stored and maintained in a peer-to-peer fashion by all course providers using the blockchain infrastructure. The blockchain cluster is composed of several blockchain nodes (or servers). Each course provider participates in

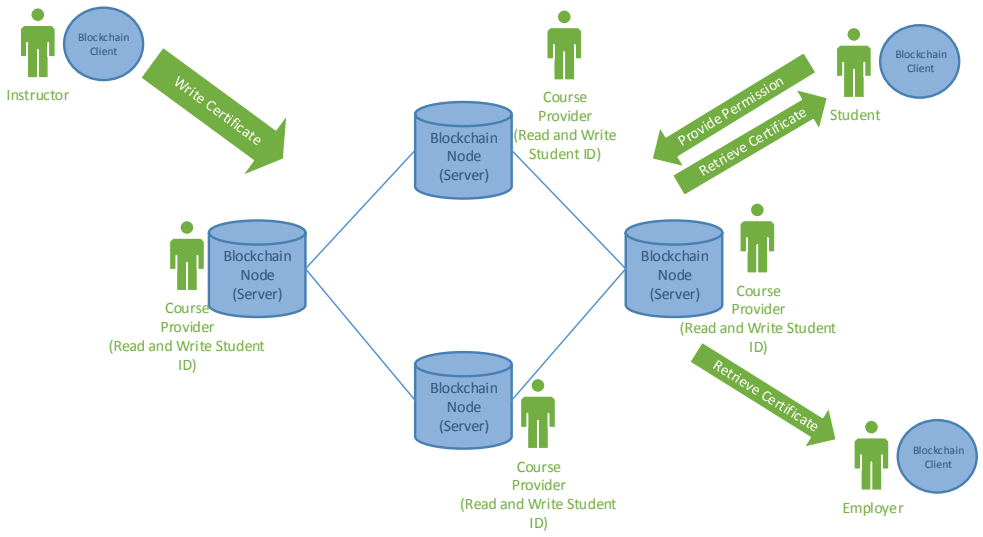

Figure 1 **Blockchain Architecture for the Course Management System**

the system by contributing one or more blockchain nodes. The data about the completion of certificates by various students are stored as individual transactions in the blockchain. These transactions can be viewed by the employer and the student.

While there are many different blockchain technologies (Pilkington 2015) available, we have used Multichain (Christidis and Devetsikiotis 2016) to implement our example system. In the next subsection, we describe the Multichain and how it is used to implement the course management system.

5. System Design

5.1. Multichain

Multichain is a open source private blockchain platform that can be installed as a standalone system across multiple nodes. The Multichain software is installed as a server daemon and can be installed as a cluster of blockchain nodes. Application software can communicate with a Multichain server using the JSON-RPC protocol. In the next section, we describe how multichain is used in our application to develop the ownership and access management of educational certificates.

5.2. Interaction Protocol

In our system, the course providers are the hosts of Multichain nodes. The authenticity of a course certificate is implemented using several streams to hold and manage access to the data. We employ four Multichain streams to store and control access privileges.

1. Stream S_1 contains the public keys of all students in the system.

2. Stream S_2 contains the encrypted course completion data. Once the student completes a course, the course provider encrypts the course completion certificate with the help of a randomly generated key using the symmetric encryption technology **??** and stores the encrypted certificate in this stream along with the student ID.

3. Stream S_3 contains information on how users can gain access to their certificates stored in S_2. In stream S_3, the key for the certificate stored in S_2, is encrypted with the public key of the certificate owner (the student). Retrieving the key for the certificate from here will help the user in recovering the certificate document from Stream S_2

4. Stream S_4 contains the transaction of $user_i$ giving employer e access to document d_{uc}. This stream keeps a track of list of employers to whom the access right of certificates are given by the student.

In the next few subsections, we desribe how Multichain can be used to manage the various activity in the system using the four streams described above.

5.3. Student Signature Generation

In this step, an user u (Student) registers with a course provider. The course provider takes the profile data of an user u as input and hashes it to generate an id (sig_u); the unique student ID (sig_u) can be generated by hashing the student profile (Line 1, Algorithm 4). This student u can now be identified by the sig_u throughout the system. With the help of asymmetric key generation we generate private key (pk_u) and shared key (sk_u) for student

Item	Description
u	Student or user
pk_u	private key of user u
sk_u	public/shared key of user u
sig_u	signature/id of user u
s	Course Provider
S_i	Stream i
c	Course
d_{uc}	Certificate document created when user u completes course c
$pk_{d_{uc}}$	Private Key of Certificate document d_{uc}
$sk_{d_{uc}}$	Public Key of Certificate document d_{uc}
$sig_{d_{uc}}$	Signature/ID of Certificate document d_{uc}
c_{value}	Encryption of the document with the document's public key
c_{key}	Encryption of certificate private key by user's shared key
c_{access}	Encryption of student id and private key of certificate d_{uc} with the shared key of the employee

Table 1 Table of Symbols

u (Line 2, Algorithm 4). The shared key (sk_u) and unique student ID (sig_u) are written in the stream, S_1 (Line 4, Algorithm 4). The private key (pk_u) is provided to the user u ((Line 3, Algorithm 4)). Once a student registers with a course provider, the student ID can be reused to register for different courses throughout the application. Also hashing function based on student's profile ensure one user will have one id.

Input: profile data of user u;
Result: sig_u, sk_u, pk_u

1 $sig_u \leftarrow hash(u)$;
2 $(pk_u, sk_u) \leftarrow gen\_key(sig_u()$;
3 return pk_u to user u ;
4 publish (sig_u, sk_u) to S_1;

Algorithm 4: Signature Generation from Profile Creation

5.4. Certificate Creation and Writing

A student u completes a course c with an instructor and the instructor generates the course completion certificate (d_{uc}). Once this certificate is created, it gets an unique id ($sig_{d_{uc}}$), a public key ($sk_{d_{uc}}$) and a private key ($pk_{d_{uc}}$) (Line 2, Algorithm 5). The course provider encrypts the course completion certificate (d_{uc}) with the public key of the certificate ($sk_{d_{uc}}$) (Line 3, Algorithm 5). This encrypted document (c_{value}) is then stored in S_2 (Line 4, Algorithm 5), against the student ID, sig_u.

The private key of the certificate ($pk_{d_{uc}}$) is encrypted with the public key of the student who got the certificate (Line 5, Algorithm 5). This encrypted key is referred to as c_{key} and is written into stream S_3 (Line 6, Algorithm 5). Thus, in future the student can access the private key of the document ($pk_{d_{uc}}$) using his own private key (pk_u)from S_3 and subsequently, which he can use to access the certificate (d_{uc}) from stream S_2.

Input: Course completion of c by student u ;
Result: $(d_{uc}, sk_{d_{uc}}, pk_{d_{uc}})$

1 $sig_{d_{uc}} \leftarrow hash(d_{uc})$;
2 $(pk_{d_{uc}}, sk_{d_{uc}}) \leftarrow gen\_key(sig_{d_{uc}})()$;
3 $c_{value} \leftarrow encrypt(sk_{d_{uc}}(d_{uc}))$;
4 publish (sig_{u_i}, c_{value}) to S_2;
5 $c_{key} \leftarrow encrypt(sk_u(pk_{d_{uc}}))$;
6 publish (sig_{u_i}, c_{key}) to S_3;
7 return (d_{uc}, c_{key}, c_{value});

Algorithm 5: Certificate Creation

5.5. Access to Certificate by an Employer

The employer, e registers with the system through any course provider. This follows Algorithm 4 to generate employer (sig_e), private key (pk_e) and shared key (sk_e) of the employer e (Line 1, Algorithm 6). The employer requests permission to access certificates for a student, u, the student provides the access permission in the following way: The student first retrieves the private key of his certificate ($pk_{d_{uc}}$) from S_2 (as mentioned in Use Case 2). The student then encrypts this ($pk_{d_{uc}}$) and his student id (sig_u), with the public key of the employer (sk_e) (Line 2, Algorithm 6) and publishes it to stream S_4 along with the employers ID (Line 3, Algorithm 6). The employer then accesses the data using his own private key (pk_e). The employer now has access to the student's user id (sig_u) and the private key of the certificate ($pk_{d_{uc}}$). The employer can then retrieve the document (d_{uc})

from S_2 using the private key of the certificate ($pk_{d_{uc}}$) associated with the user id (sig_u) of the student (Line 3, Algorithm 6).

Input: Employer e requests access to certificate d_{uc} ;
Result: (sig_e, c_{access})

1 $(sig_e, sk_e, pk_e) \leftarrow$ Call Algorithm 4 for e ;
2 $c_{access} \leftarrow \text{encrypt}(sk_e(sig_u, pk_{d_{uc}}))$;
3 publish (sig_e, c_{access}) to S_4;
4 $d_{uc} \leftarrow \text{decrypt}(pk_{d_{uc}}(c_{value}))$;

Algorithm 6: Certificate Access Grants

With the above approach, the student is able to give certificates of completion to various employers. For example, a student has completed a course on Java, a course on the Oracle DBMS and a course on wine tasting. He can give access to some of these certificates, say the Java and Oracle courses, to a prospective employer (avoiding the unnecessary information on wine tasting). The employer, upon retrieving these certificates from the blockchain, is sure about its authenticity since the certificates cannot be modified once written into the blockchain.

6. Conclusion

In this paper we demonstrated how Blockchain can be used to build applications that ensure data ownership and access management in a robust manner. Through the example of the course management system we were able to ensure that owners have full control over their data and can decide who is able to access the said data. This paper thus provides an example of how blockchain can be used to address the major data ownership issues that are prevalent today. Applications built using Blockchain as a platform therefore, can mitigate the influence of third party organizations.

References

Ali M, Nelson J, Shea R, Freedman MJ (2016) Blockstack: A global naming and storage system secured by blockchains. *2016 USENIX Annual Technical Conference (USENIX ATC 16)*, 181–194 (USENIX Association).

Back A, Corallo M, Dashjr L, Friedenbach M, Maxwell G, Miller A, Poelstra A, Timón J, Wuille P (2014) Enabling blockchain innovations with pegged sidechains. *URL: http://www. opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains* .

Chain A (2014) Lifetime: How blockchain technology might transform personal insurance by michael mainelli and chiara von gunten. *Z/Yen Group, Long Finance December* .

Christidis K, Devetsikiotis M (2016) Blockchains and smart contracts for the internet of things. *IEEE Access* 4:2292–2303.

Crosby M, Pattanayak P, Verma S, Kalyanaraman V (2016) Blockchain technology: Beyond bitcoin. *Applied Innovation* 2:6–10.

Datta A, Buchegger S, Vu LH, Strufe T, Rzadca K (2010) Decentralized online social networks. *Handbook of Social Network Technologies and Applications*, 349–378 (Springer).

Haber S, Stornetta WS (1990) How to time-stamp a digital document. *Conference on the Theory and Application of Cryptography*, 437–455 (Springer).

Heilman E, Baldimtsi F, Goldberg S (2016) Blindly signed contracts: Anonymous on-blockchain and off-blockchain bitcoin transactions. *International Conference on Financial Cryptography and Data Security*, 43–60 (Springer).

Kakavand H, Kost De Sevres N (2016) The blockchain revolution: An analysis of regulation and technology related to distributed ledger technologies .

Kaminski J (2014) Nowcasting the bitcoin market with twitter signals. *arXiv preprint arXiv:1406.7577* .

Kishigami J, Fujimura S, Watanabe H, Nakadaira A, Akutsu A (2015) The blockchain-based digital content distribution system. *Big Data and Cloud Computing (BDCloud), 2015 IEEE Fifth International Conference on*, 187–190 (IEEE).

Kogias EK, Jovanovic P, Gailly N, Khoffi I, Gasser L, Ford B (2016) Enhancing bitcoin security and performance with strong consistency via collective signing. *25th USENIX Security Symposium (USENIX Security 16)*, 279–296 (USENIX Association).

Kosba A, Miller A, Shi E, Wen Z, Papamanthou C (2016) Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. *Security and Privacy (SP), 2016 IEEE Symposium on*, 839–858 (IEEE).

Nakamoto S (2009) Bitcoin: A peer-to-peer electronic cash system .

Needham RM, Schroeder MD (1978) Using encryption for authentication in large networks of computers. *Communications of the ACM* 21(12):993–999.

Peters GW, Panayi E (2016) Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. *Banking Beyond Banks and Money*, 239–278 (Springer).

Pilkington M (2015) Blockchain technology: principles and applications. *Browser Download This Paper* .

Shanahan NA (2016) Overcoming information asymmetry in patent pledge records .

Swan M (2015) *Blockchain: Blueprint for a new economy* (" O'Reilly Media, Inc.").

Szabo N (1997) Formalizing and securing relationships on public networks. *First Monday* 2(9).

Wright A, De Filippi P (2015) Decentralized blockchain technology and the rise of lex cryptographia .

Yue X, Wang H, Jin D, Li M, Jiang W (2016) Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. *Journal of medical systems* 40(10):218.

Zyskind G, Nathan O, et al. (2015) Decentralizing privacy: Using blockchain to protect personal data. *Security and Privacy Workshops (SPW), 2015 IEEE*, 180–184 (IEEE).