Towards Blockchain-Based Collaborative Intrusion Detection Systems

Nikolaos Alexopoulos ✉, Emmanouil Vasilomanolakis ✉,
Natália Réka Ivánkó, and Max Mühlhäuser

Telecooperation Group
Technische Universität Darmstadt
Darmstadt, Germany
Email: [alexopoulos,vasilomano,max]@tk.tu-darmstadt.de,
[nataliareka.ivanko]@stud.tu-darmstadt.de

Abstract. In an attempt to cope with the increased number of cyber-attacks, research in Intrusion Detection Systems (IDSs) is moving towards more collaborative mechanisms. Collaborative IDSs (CIDSs) are such an approach; they combine the knowledge of a plethora of monitors to generate a holistic picture of the monitored network. Despite the research done in this field, CIDSs still face a number of fundamental challenges, especially regarding maintaining trust among the collaborating parties. Recent advances in distributed ledger technologies, e.g. various implementations of blockchain protocols, are a good fit to the problem of enhancing trust in collaborative environments. This paper touches the intersection of CIDSs and blockchains. Particularly, it introduces the idea of utilizing blockchain technologies as a mechanism for improving CIDSs. We argue that certain properties of blockchains can be of significant benefit for CIDSs; namely for the improvement of trust between monitors, and for providing accountability and consensus. For this, we study the related work and highlight the research gaps and challenges towards such a task. Finally, we propose a generic architecture for the incorporation of blockchains into the field of CIDSs and an analysis of the design decisions that need to be made to implement such an architecture.

1 Introduction

Nowadays, cyber-attacks are increasing in both their numbers and sophistication. In particular, recent attacks such as the case of the so-called Wannacry malware [15], highlight the need for protection, especially in the realm of critical infrastructures. To cope with such challenges, research in cyber-security has focused on more collaborative approaches that are broadly referred to as CIDSs [33]. CIDSs attempt to create an improved and holistic picture of the monitored networks via, as their name implies, collaboration among participants.

Nevertheless, a number of research challenges remain unsolved with regard to CIDSs. First, one of the most important aspects of such systems is the trust establishment (and management) of the participants. That is, the techniques

that the system employs to ensure that the collaborative monitoring units trust each other, as well as methods for quantifying the quality of the exchanged alert data. In addition, a fair and public approach that provides accountability among the monitors of the CIDS is also a topic that has not been touched until now. Lastly, methods for providing consensus in a distributed environment and specifically in the CIDS scenario are yet to be explored.

Recently, there has been a spike in the interest around blockchains in the context of several industry applications, including critical infrastructures. Apart from, the now established, use of blockchains among financial institutions [12], identity schemes secured by blockchain technology have been discussed, with many startups offering such solutions [28]. Furthermore, critical services, such as healthcare providers [29, 18], national and state authorities [36], land registries [20], and the energy sector [17, 27], are considering incorporating distributed ledgers (i.e. blockchains) in their workflows. They take advantage of the immutability and consensus properties of these designs, in order to secure their respective systems.

In this paper, we introduce the idea of utilizing blockchain technology as a mechanism for improving CIDSs. In particular, we argue that certain functionality and properties of blockchains can be of significant benefit for collaborative intrusion detection; namely for the improvement of trust between monitors, for providing accountability and as a consensus mechanism. For this, we carefully study the related work and highlight the research gaps and challenges towards such a task. Finally, we propose a generic architecture for the incorporation of blockchains into the field of CIDSs.

The rest of this paper is organized as follows. First, Section 2 provides some background knowledge with regard to both CIDSs and blockchains. Section 3 discusses the related work in the intersection of CIDSs and blockchain technologies. Section 4 proposes a number of requirements for CIDSs with regard to the need for trust and fairness during collaboration. Sections 5 and 6 discuss our proposed CIDS architecture and the various design considerations, respectively. Finally, Section 7 concludes this paper and summarizes our future directions.

2 Background

In this section, we present necessary background information with regard to blockchains and CIDSs.

2.1 Collaborative Intrusion Detection

There has been a lot of work in the area of CIDS over the last years [33]. The majority of it has been focusing on novel architectures and collaborative detection techniques. In more detail, CIDSs can be classified with regard to the network placement of their monitors to: *centralized*, *hierarchical* and *distributed*. Due to various fundamental disadvantages of centralized and hierarchical CIDSs (e.g., scalability and the existence of a Single Point of Failure (SPoF)) [33], this

paper focuses mainly on distributed systems. In this class, a number of such systems have been proposed in the literature, e.g. [38, 34, 22]. However, most of these proposals deal with the construction of sophisticated architectures, hence not addressing other challenging topics. For instance, validating and managing trust between monitors (see Section 3.1), or creating consensus when exchanging alert data, have not been tackled in sufficient extent in the state of the art.

2.2 Blockchains

Despite recent media, corporate, and research coverage, there is no standard definition for blockchain technology - or simply blockchains - yet. It can be described as a distributed data structure, which is shared and replicated between the participants of a peer-to-peer network. The data structure itself is built from a back-linked list of blocks, where each block is identified by its cryptographic hash and also contains the hash of the previous block. This property establishes a cryptographic link between blocks, creating a so-called "blockchain" [1] that all participants can examine, yet without being able to tamper with. Due to this fact, blockchains are considered an implementation of a shared *secure distributed ledger*, where the participants can read from - most of the time without any constraints - and write to, when only specific constraints are met.

Regarding the control of these permissions, current blockchain implementations fall into three categories: *public*, *consortium* and *private* [26]. In the case of *public* blockchains, such as Bitcoin [25] and Ethereum [37], everyone can read and maintain the ledger, i.e. there is no membership mechanism in place.

Meanwhile, in *consortium* blockchains, such as Hyperledger [1] [8], a pre-defined consortium of peers is responsible for maintaining the chain. In *private* blockchains, such as Monax [2], a single entity controls the system, i.e. there is no consensus process.

The process of updating the blockchain takes place via a protocol, which achieves consensus, i.e. gives guarantees that all participants agree on a uniform view of the ledger that contains only valid transactions, ensuring the integrity and consistency of the ledger [3]. This protocol may vary a lot and depends on both the type of the blockchain implementation and the threat model. To offer guaranteed security properties, public blockchains design the consensus part to be either computationally hard (Proof-of-Work) or based on the possession of a scarce resource within the system (Proof-Of-Stake). On the other hand, consortium and private blockchains apply some kind of Byzantine [19] or benign fault tolerant algorithms, such as PBFT [10] or SIEVE [9], to cope with malicious nodes. For an overview of Bitcoin and other cryptocurrencies, we refer the reader to [7].

[1] https://www.hyperledger.org
[2] https://monax.io

3 Related Work

This section discusses the related work in the field of CIDSs, emphasizing in trust management, as well as in the field of blockchains.

3.1 Building Trust in CIDSs

Beyond the fundamental (architectural-level) research on CIDSs, some work has been done lately with regards to trust management. In more detail, researchers have proposed trust management mechanisms to cope with both the *insider attack* problem[3] as well as to enhance the overall quality of the collaboration [31].

Specifically, trust management in CIDSs can be distinguished based on the overall goal of the respective (trust) mechanism. In this context, computational trust is most commonly applied to quantify the trust levels between monitoring nodes. That is, if a monitor is compromised or starts disseminating false information, its trust score will decrease and eventually some response action will take place (e.g., blacklisting). Another approach is to attempt to quantify the quality of the alert data (rather than the source of it). In such a scenario, the trust model attempts to measure the quality of the alerts themselves or assign a reputation score to certain parameters of an alert (e.g., to the IP address of an adversary) [4].

The majority of the work proposed in this area makes use of computational trust mechanisms, based on various mathematical models, to measure the trustworthiness of the monitors [16, 14]. In particular, the basic concept is that a monitor can utilize its own old experiences, with respect to its communication with other monitors, and via the usage of certain computational trust methods (e.g., Bayesian statistics) can infer (with a certain probability and confidence) the amount of trust it can place on others. Nevertheless, to the best of our knowledge, no work has been done towards providing CIDSs with strong accountability and consensus properties.

3.2 Blockchains as a means of collaboration

There has been a recent explosion of interest around blockchains and several industry applications have been proposed in the last five years. Each application is developed for a special use case, thus requires a different blockchain implementation, which has to provide custom, unique characteristics. For example, there are specific implementations, which were developed to enhance privacy by enabling different parties to use the system, meanwhile keeping the stored data completely private [39]. Other implementations use cryptographic identity schemes, which offer full anonymity and unlinkability between the transactions [28]. Due to these

[3] This refers to the case where a monitor, which is part of the CIDS, turns malicious and attempts to attack or misguide other monitors of the system.

versatile properties, various industries have started to investigate the potential of this technology.

In the energy sector, blockchain can facilitate a peer-to-peer market, where machines buy and sell energy automatically, according to predefined criteria [23, 24]. For example, prosumers with solar panels can record their output in the blockchain and sell it to other parties via smart contracts. Moreover, Azaria et al. [2] propose a novel, decentralized record management system to handle and store medical data by using blockchains to ensure data integrity and to enforce access control policies.

These applications have demonstrated that the combination of IoT and blockchains can lead to rewarding results [11]. All of them benefit from the fact that this technology allows peers to communicate with each other in a verifiable manner without trusting each other and without any trusted intermediary; while being able to preserve their anonymity and guarantee the integrity of their data.

4 Requirements

As a first step of any system design attempt, it is very important to clearly articulate the requirements of the goal system. Thus, in accordance with the related work in [33, 31], we specify the requirements for an effective and trustworthy CIDS:

- **Accountability:** Participating parties should be held accountable for their actions.
- **Integrity:** The integrity of the alert data is very important for detecting attacks over time as well as for post-mortem analysis (e.g., during forensic analysis).
- **Resilience:** The system should not have SPoFs and should not depend on small groups of participants.
- **Consensus:** The system should be able to reach consensus on the quality of individual alert data and on the trustworthiness of each participant.
- **Scalability:** The system should be able to scale to a large number of participants/monitors and also handle churn.
- **Minimum Overhead:** The communication and computation overhead should be kept as low as possible.
- **Privacy:** Participants should be able to reserve their right to privacy and selectively disclose alert data as they wish. However, at the same time, the accountability and integrity requirements should still hold.

In this section, we specified seven requirements for a successful CIDS design. These requirements will guide our design choices and argumentation, for the rest of this paper. However, the requirements presented above are not orthogonal regarding the design decisions they favor. That is, there are inherent trade-offs between them (e.g. accountability vs. privacy). These trade-offs are further explored in Section 6.

5 A blockchain-based architecture for CIDSs

To satisfy the requirements of Section 4, we propose the utilization of a secure distributed ledger, as e.g. implemented by blockchain technology, to secure the exchange of alerts between the collaborating nodes.

As a simple example, raw alert data generated by the monitors are stored as transactions in a blockchain, replicated among the participating nodes of the network. For an insight into the options regarding the nature of the actual data stored in the blockchain (e.g. alert hashes, bloom filters) see Section 6. The nodes involved, run a consensus protocol to guarantee the validity of the transactions before adding them in a block. This process guarantees that only well-formed alerts are included in the blockchain, that alert data transactions are tamper-resistant, and that each participating entity has a global view of the alerts.

This way, the participants are held *accountable* for their actions, as the latter are transparent to the network. Furthermore, the *integrity* of the data is guaranteed and the system has no *SPoF*, as it can tolerate as many byzantine failures as the underlying consensus protocol. The communication *overhead* of the construct can be managed e.g. by storing hashes of the alert data in the blockchain instead of the raw data. This way, a node would be able to verify the integrity of the alerts it receives by comparing their hash value with the corresponding hashes that are stored on the chain. There are a multitude of design considerations, like the one mentioned above towards the realization of such a system. In this section, we focus on the proposed generic architecture, while in section 6, we explore the design space of a possible implementation.

The proposed architecture for a blockchain-based distributed CIDS can be seen in Figure 1. The participating nodes in the blockchain network are either monitor units, analysis units, or perform both tasks simultaneously, which is the most general case. Communication between the nodes takes place in two logical layers, namely the *Alert Exchange layer* and the *Consensus layer*.

In the *Alert Exchange layer*, the implemented CIDS performs the alert data dissemination process. Specifically, the participating nodes exchange or collect alert data with respect to their role as monitors or analysis units. The exact communication mechanism in use is determined by the needs of the CIDS. For example, a flooding or gossiping [13] protocol can be used to disseminate data in the Alert Exchange layer of a distributed CIDS, while on-demand data exchange is also an option.

Second, there is a *Consensus layer*, where a subset (not necessarily proper) of peers, e.g. only the analysis units, run a consensus protocol, and agree on which transactions should be included in the ledger. The most basic scenario is one in which all members of the CIDS participate in the consensus protocol, and all alert data is stored in the blockchain. The connection between the two layers, the result of the Consensus layer, and the properties of the underlying blockchain construct, together enforce strict accountability of the participants and guarantee data integrity.

Furthermore, if required, it is possible to keep specific alert data confidential among a subset of peers. For instance, there might be a scenario, where specific

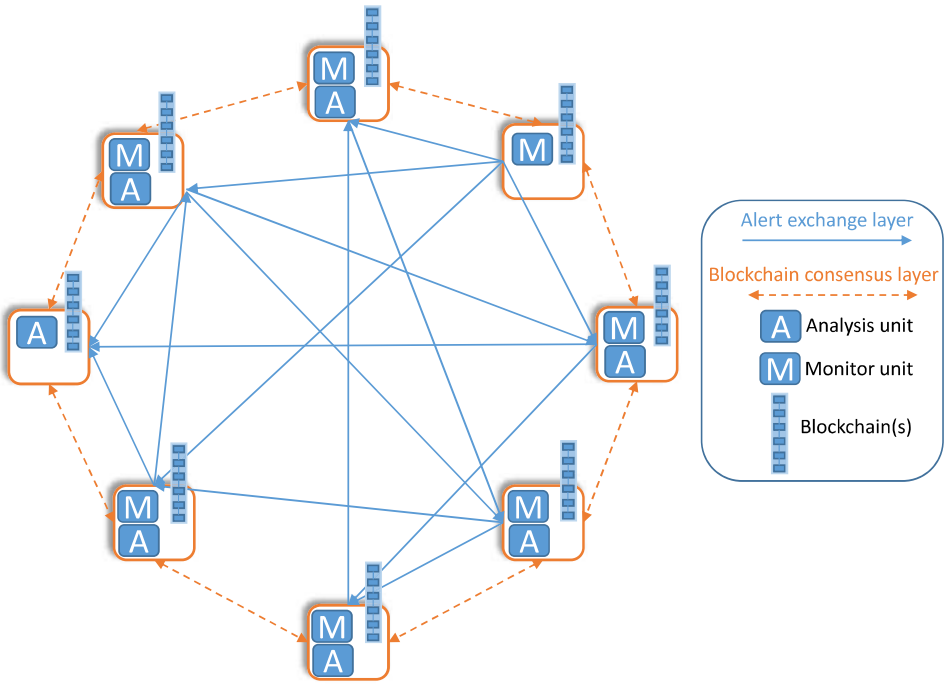

Fig. 1. Generic architecture of a blockchain-based CIDS.

alerts contain confidential information, which should not be revealed to anyone outside a specific corporation. In this case, the nodes who should have access to these specific alerts, can participate in a separate collaborative network and create a separate blockchain for it. The specific alert data can be encrypted in both layers as well as in the distributed ledger, and the keys will be available only to the participating nodes. This way, peers that do not belong to this collaborative group, and do not hold the specific secret key, cannot view alert data exchanged and processed among the certified participants. As a result, peers are able to collaborate in multiple groups without revealing any confidential alert information to external nodes, while still staying on the same CIDS network.

6 Design considerations

In Section 5, we proposed a general architecture for a trustworthy CIDS. However, there are a number of design and implementation choices regarding the realization of this architecture. Using blockchains as the basis of the design, inherently offers a degree of accountability, integrity and resilience to our system. Nevertheless, note that the choice of the type of blockchain and consensus

algorithm to use, affects the degree to which these requirements are satisfied. Other characteristics, such as the scalability of the network, the communication overhead and the privacy of the participants, might depend on the distributed ledger type (w.r.t. the implemented consensus algorithm) or the data format via which the alerts are exchanged. Note that each combination of these alternatives provides unique characteristics to a CIDS, which enables us to design the system architecture tailored to the real world application.

In this section, we recount the aforesaid alternatives while focusing on the trade-offs between them, with respect to the requirements laid out in Section 4.

Governance of distributed ledger: In general, both public (permissionless) and corporate (permissioned) blockchain designs provide authenticity, integrity and resilience to the system, via guaranteeing a global, partially ordered view of alert transactions. However, they have their own advantages and disadvantages. Public blockchains provide an uncontrolled network, which everybody can freely join, and where every peer can read from and update the distributed ledger. General-purpose CIDSs, especially cyber incident monitors and network telescopes, e.g., [30, 32], can benefit considerably from these advantages. The stored alert data are integrity-protected and available to everybody, while the participants remain accountable for their actions. These properties can provide high quality data for the scientific community to examine attacks, create statistics, or gather data to train another CIDS. Nevertheless, a major shortcoming is the possible transaction cost that a peer has to pay in order for her alert to be included in a public Proof-of-Work blockchain, e.g. in Bitcoin a peer can store 80 bytes of data using the "OP_RETURN" [5] transaction script, with a transaction fee of a couple of USD.

In consortium blockchains, access permissions are more tightly controlled and rights to modify or even read the blockchain state are restricted to a specific set of users. For example, the consensus is controlled by a pre-selected set of nodes. In this case, the validators are known and any risk of a lot of malicious peers joining the system (e.g. due to a sybil attack) and destroying the accuracy of the CIDS is mitigated. Additionally, if a peer starts to behave maliciously, e.g., starts sending fake alerts, the organization can easily change the rules of the blockchain and revert the fake transactions. This makes consortium blockchains a better choice for institutions and groups of institutions who do not want to reveal the alert data publicly and want to keep the system's participants under control.

Consensus: Apart from the issue of who is able to view and add alert data (in the form of transactions) to the blockchain, the selection of the consensus algorithm and of the peers that take part in it, is of great importance. Especially in corporate blockchain designs, there is the possibility to choose a subset of peers, i.e., super-peers, that will be responsible for running the consensus algorithm; hence, offering integrity guarantees to the system. Apart from that, the consensus algorithm that is selected will greatly affect the security guarantees of the system. We have already presented several options, including Proof-of-Work,

Proof-of-Stake and traditional byzantine fault-tolerant designs in Section 2. The choice of the consensus algorithm defines the adversary model of the system, i.e., the ratio of honest and malicious peers in the CIDS. Furthermore, each approach comes with a different scalability potential, e.g., practical byzantine fault tolerant (PBFT) designs will generally be less scalable (in terms of the peer population) than Proof-of-Work or Proof-of-Stake based ones [35].

Data on/off the ledger: Another question that rises is related to the alert data and their granularity during the sharing process, in each of the two communication layers (*Alert Exchange* and *Blockchain Consensus*). For this, there are various strategies that can be considered in a CIDS; each one having its own advantages and disadvantages. For instance, exchanging raw alert data can provide the deepest level of granularity. However, this comes with a major communication overhead. In addition, considering the large amount of data that a local IDS generates, such an approach would not scale. Another approach is to instead only share compact representations of alert data. For instance, in [21, 34] researchers have proposed the exchange of bloom filters [6] containing such a mapping of alerts. Such approaches fulfill a number of requirements, namely the minimal overhead, privacy and integrity. Nevertheless, when only exchanging aggregated/compact versions of the alerts the accuracy of the system might be decreased.

Our proposal is to use a compact representation, e.g. bloom filters, for the communication in the *Consensus layer*, in order to decrease the overhead and size of the blockchain construct. On the other hand, in the *Alert Exchange* layer, where data exchange can potentially take place on demand, exchanging raw data would offer the highest level of accuracy to the system.

Data encryption: The norm in both public and corporate blockchain networks is that the transactions can be observed by all participating peers, a fact which could give away information that should not be revealed, e.g., sensitive corporate information. Therefore, in some CIDS usage scenarios, it is important to provide a mechanism that protects the privacy of the participating parties with respect to alert data and confidential information (exchanged in the *consensus layer*). One solution can be encrypting the alert data by using symmetric key cryptography and making the keys available only to the participants who should have the right to read them[4]. This allows every peer to stay on the same network, but be able to decrypt and examine only the alert data which they are certified to access. However, this would produce overhead in the form of key management and distribution. Nevertheless, as described in the previous paragraph, another approach is to only exchange compact representations of the alert data (e.g., bloom filters or hashes). In such a case the problem of over-sharing is mitigated.

In general, there are numerous considerations when designing a blockchain-based CIDS. In this section, we gave an overview of the most challenging ones

[4] An asymmetric approach, e.g., with a Public Key Infrastructure (PKI), is also possible, however a lot of overhead would be expected in the key distribution and maintenance process.

and presented the trade-offs between them. There is no generic optimal solution to making decisions about the considerations presented above. The specific requirements of a CIDS, such as the expected number of peers, the expected alert volume, the required privacy level, etc., will guide these decisions.

7 Conclusion and Future Work

To cope with the increase and the sophistication of cyber-attacks in critical infrastructures, there is a demand for collaboration between the defenders. In order to practically improve the area of collaborative intrusion detection, there is a need for constructing a trusted and accountable environment for the participating monitors. In this paper, we presented a generic architecture for the creation of such a CIDS, and explained how blockchain technology can be incorporated. In addition, we identified the open research challenges and discussed the various directions of this research area.

With regard to our future work, we plan to demonstrate the feasibility of a blockchain-based CIDS via implementing a proof of concept. In more detail, we are currently implementing such a system based on the architecture presented in this paper and by taking into account the design considerations of Section 6. We plan to evaluate our approach by examining whether the introduction of a blockchain adds overhead to the system, and if yes how much. Another challenge is the incorporation of computational trust mechanisms in our design, with the goal of increasing the trustworthiness of the overall system even further. In conclusion, we believe that *by-design* secure collaborative platforms, such as the ones offered by the various implementations of secure distributed ledgers, can increase the performance and durability of CIDSs, and in turn offer better protection to critical infrastructures.

Acknowledgments. This work has received funding from the European Union's Horizon 2020 Research and Innovation Program, PROTECTIVE, under Grant Agreement No 700071. This work has also been funded by the DFG within the RTG 2050 "Privacy and Trust for Mobile Users" and within the CRC 1119 CROSSING.

References

1. Antonopoulos, A.M.: Mastering Bitcoin: unlocking digital cryptocurrencies. O'Reilly Media, Inc. (2014)
2. Azaria, A., Ekblaw, A., Vieira, T., Lippman, A.: Medrec: Using blockchain for medical data access and permission management. In: Open and Big Data (OBD), International Conference on. pp. 25–30. IEEE (2016)
3. Baliga, A.: Understanding Blockchain Consensus Models. Tech. rep., Persistent Systems Ltd. (2017)
4. Bartoš, V., Kořenek, J.: Evaluating reputation of internet entities. In: IFIP International Conference on Autonomous Infrastructure, Management and Security. pp. 132–136. Springer (2016)

5. bitcoinwiki: OP_RETURN (2017), https://en.bitcoin.it/wiki/OP_RETURN
6. Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. Communications of the ACM 13(7), 422–426 (jul 1970)
7. Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J.A., Felten, E.W.: Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In: Security and Privacy (SP), 2015 IEEE Symposium on. pp. 104–121. IEEE (2015)
8. Cachin, C.: Architecture of the hyperledger blockchain fabric. In: Workshop on Distributed Cryptocurrencies and Consensus Ledgers (2016)
9. Cachin, C., Schubert, S., Vukolić, M.: Non-determinism in byzantine fault-tolerant replication. arXiv preprint arXiv:1603.07351 (2016)
10. Castro, M., Liskov, B., et al.: Practical byzantine fault tolerance. In: OSDI. vol. 99, pp. 173–186 (1999)
11. Christidis, K., Devetsikiotis, M.: Blockchains and smart contracts for the internet of things. IEEE Access 4, 2292–2303 (2016)
12. Coindesk: Seven asian banks investigating bitcoin and blockchain tech. Available: http://www.coindesk.com/7-asian-banks-investigating-bitcoin-and-blockchain-tech/
13. Demers, A., Greene, D., Hauser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D., Terry, D.: Epidemic algorithms for replicated database maintenance. In: Proceedings of the sixth annual ACM Symposium on Principles of distributed computing. pp. 1–12. ACM (1987)
14. Duma, C., Karresand, M., Shahmehri, N., Caronni, G.: A Trust-Aware, P2P-Based Overlay for Intrusion Detection. In: International Conference on Database and Expert Systems Applications (DEXA'06). pp. 692–697. IEEE (2006)
15. Ehrenfeld, J.M.: Wannacry, cybersecurity and health information technology: A time to act. Journal of Medical Systems 41(7), 104 (2017)
16. Fung, C.J., Zhang, J., Aib, I., Boutaba, R.: Dirichlet-based trust management for effective collaborative intrusion detection networks. IEEE Transactions on Network and Service Management 8(2), 79–91 (2011)
17. Grid, T.: Available: http://transactivegrid.net/
18. Halamka, J.D., Lippman, A., Ekblaw, A.: The potential for blockchain to transform electronic health records (2017), https://hbr.org/2017/03/the-potential-for-blockchain-to-transform-electronic-health-records
19. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Transactions on Programming Languages and Systems (TOPLAS) 4(3), 382–401 (1982)
20. Lantmäteriet, Landshypotek Bank, SBAB, Telia company, ChromaWay, Kairos Future: The land registry in the blockchain - testbed. Tech. rep. (2017)
21. Locasto, M.E., Parekh, J.J., Keromytis, A.D., Stolfo, S.J.: Towards Collaborative Security and P2P Intrusion Detection. In: IEEE Workshop on Information Assurance and Security. pp. 333 – 339. IEEE (2005)
22. Locasto, M.E., Parekh, J.J., Stolfo, S., Misra, V.: Collaborative distributed intrusion detection. Tech. rep., Columbia University (2004)
23. Mihaylov, M., Jurado, S., Avellana, N., Van Moffaert, K., de Abril, I.M., Nowe, A.: Nrgcoin: Virtual currency for trading of renewable energy in smart grids. In: European Energy Market (EEM), 11th International Conference on the. pp. 1–6. IEEE (2014)
24. Mihaylov, M., Jurado, S., Van Moffaert, K., Avellana, N., Nowé, A.: Nrg-x-change-a novel mechanism for trading of renewable energy in smart grids. In: SMARTGREENS. pp. 101–106 (2014)
25. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008)

26. Okada, H., Yamasaki, S., Bracamonte, V.: Proposed classification of blockchains based on authority and incentive dimensions. In: Advanced Communication Technology (ICACT), 2017 19th International Conference on. pp. 593–597. IEEE (2017)
27. Rutkin, A.: Blockchain-based microgrid gives power to consumers in new york. New Scientist. https:// www. newscientist. com/ article (2016)
28. Shrier, D., Wu, W., Pentland, A.: Blockchain & infrastructure (identity, data security). Tech. rep., Tech. rep. URL: http://cdn. resources. getsmarter. ac/wp-content/uploads/2016/05/MIT_Blockchain_Infrastructure_ Report_Part_Three_May_2016. pdf (2016)
29. Suberg, W.: Factom's latest partnership takes on us healthcare (2015), https://cointelegraph.com/news/factoms-latest-partnership-takes-on-us-healthcare
30. Ullrich, J.: Dshield internet storm center. https://www.dshield.org/ (2000)
31. Vasilomanolakis, E., Habib, S.M., Malik, R.S., Milaszewicz, P., Mühlhäuser, M.: Towards trust-aware collaborative intrusion detection: challenges and solutions. In: International Conference on Trust Management (IFIPTM). Springer (2017)
32. Vasilomanolakis, E., Karuppayah, S., Kikiras, P., Mühlhäuser, M.: A honeypot-driven cyber incident monitor: lessons learned and steps ahead. In: International Conference on Security of Information and Networks. pp. 158–164. ACM (2015)
33. Vasilomanolakis, E., Karuppayah, S., Mühlhäuser, M., Fischer, M.: Taxonomy and Survey of Collaborative Intrusion Detection. ACM Computing Surveys 47(4), 33 (2015)
34. Vasilomanolakis, E., Krügl, M., Cordero, C.G., Mühlhäuser, M., Fischer, M.: Skipmon: A locality-aware collaborative intrusion detection system. In: Computing and Communications Conference (IPCCC), IEEE 34th International Performance. pp. 1–8. IEEE (2015)
35. Vukolić, M.: The quest for scalable blockchain fabric: Proof-of-work vs. bft replication. In: International Workshop on Open Problems in Network Security. pp. 112–125. Springer (2015)
36. Walport, M.: Distributed ledger technology: Beyond blockchain. UK Government Office for Science (2016)
37. Wood, G.: Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper 151 (2014)
38. Zhou, C.V., Karunasekera, S., Leckie, C.: A peer-to-peer collaborative intrusion detection system. In: International Conference on Networks. pp. 118–123. IEEE (2005)
39. Zyskind, G., Nathan, O., Pentland, A.: Enigma: Decentralized computation platform with guaranteed privacy. arXiv preprint arXiv:1506.03471 (2015)