Protecting Infrastructure Data via Enhanced Access Control, Blockchain and Differential Privacy

Asma Alnemari, Suchith Arodi, Valentina Rodriguez Sosa, Soni Pandey, Carol Romanowski, Rajendra Raj, Sumita Mishra

Chapter 7

PROTECTING INFRASTRUCTURE DATA VIA ENHANCED ACCESS CONTROL, BLOCKCHAIN AND DIFFERENTIAL PRIVACY

Asma Alnemari, Suchith Arodi, Valentina Rodriguez Sosa, Soni Pandey, Carol Romanowski, Rajendra Raj and Sumita Mishra

Abstract Protecting critical infrastructure data is challenging because it typically includes sensitive information that is often needed by analysts to answer crucial questions about the critical infrastructure. For example, in the healthcare sector, epidemiologists need to analyze personally identifiable information to track the spread of diseases or regional emergency services managers may need to view details of all 911 calls made during a hurricane or terrorist incident. In other situations where personally identifying information is not needed to perform analyses, studies have shown that anonymization approaches such as k-anonymity or l-diversity cannot safeguard the information from inadvertent or malicious exposure. Additionally, recent data breaches involving critical infrastructure information demonstrate that current access control mechanisms, including role-based access control, are neither sufficient to secure the information nor adequate to prevent the ensuing loss of privacy. This chapter presents a novel approach that integrates existing access control mechanisms with blockchain and differential privacy to protect infrastructure data.

Keywords: Data protection, data privacy, access control, blockchain

1. Introduction

Sensitive datasets, such as data generated by critical infrastructure assets, often need to be analyzed to recognize trends, optimize resources and determine proper courses of action [12]. However, critical infrastructure data typically includes a great deal of personally identifiable information (PII) in addition to

other sensitive data pertaining to locations, building access, perimeter security, etc. Based on their data needs, analysts can be categorized into three groups:

- **Primary Analysts:** These users must have complete access to all the critical infrastructure data and related data products to perform their tasks. For example, an emergency manager in a county in the United States may need to see the details of every call in the county's 911 system.
- **Secondary Analysts:** These users may need access to critical infrastructure data that includes some personally identifiable information and/or sensitive information, but the rest of the data can be restricted using aggregation or anonymization. For example, an employee in a different agency who analyzes resource allocation in a county, only needs to see aggregated information from the dataset with most of the personally identifiable information removed. However, the employee may need access to location information that could become personally identifiable information in sparsely-populated areas of the county.
- **Tertiary Analysts:** These users do not need to see any personally identifiable information, but may need access to aggregated or anonymized information. For example, a member of the local news media should not have access to any sensitive information, but may be allowed to see summary data.

The dilemma is to ensure the maximal protection of critical infrastructure data while providing appropriate access to legitimate uses by the three types of data analysts. In all these cases, system access is permitted, but the access must be controlled.

Current access control methods have proven to be inadequate for sensitive datasets. According to tracking by the Privacy Rights Clearinghouse [13], more than 550 data breaches were publicly reported in 2017. In other words, on average, more than 1.5 data breaches occurred daily in the United States. Because these correspond to the events that were recorded and reported, the actual number of data breaches is likely to be considerably higher. In many cases, the breaches were caused by inadequate access control mechanisms that essentially enabled outsiders or malicious insiders to breach them fairly easily.

Access control mechanisms must be enhanced to provide better data security and protection. This chapter argues that access control should be considered to be only the first layer of data protection. The logical next layer is data anonymization – for example, abstracting individual data items as ranges can obscure sensitive values and concept hierarchies can mask specific attributes. However, most techniques such as k-anonymity and l-diversity cannot prevent the exposure of private information when data is queried [9]. Because anonymization is inadequate, a crucial role can be played by differential privacy [6] in providing overall data protection. Differential privacy makes the presence or absence of an individual or single entity indistinguishable, thereby reducing any benefit of adversarial background knowledge about individuals'

data in a dataset. For example, Lin et al. [8] propose an approach that adds random noise to true answers, but even this method is not foolproof. An attacker repeatedly asks the same question and a different answer is provided each time; however, this itself provides a clue that the information is sensitive. Complicating this situation is the fact that real-world data is not independent. This requires the implementation of a comprehensive strategy to hide correlations between attributes [19].

This chapter proposes a layered methodology that enhances access control using blockchain and differential privacy to provide strong data protection for critical infrastructure assets and reduce data privacy losses. The proposed framework develops the appropriate access and differential privacy strategies based on user types and dataset characteristics.

2. Motivating Scenarios

This section provides examples that illustrate how the proposed framework would be applied in different domains. Emergency management and healthcare are chosen as the sample domains, although similar scenarios can be developed for other critical infrastructure domains. While the easiest way to safeguard datasets is to completely restrict them, the proposed framework assumes that analyses of the datasets are beneficial as long as the protection of sensitive data is assured.

2.1 Scenario 1: Emergency Services Sector

Emergency response in the United States is typically handled at the municipal level (village, town or city) until an event overwhelms the local resources [15]. At this point, the emergency response is managed at the county level from an emergency operations center. Data about the emergency event is collected by the countywide 911 system and other repositories (e.g., after-action reports). The collected data is analyzed to identify ways in which municipalities can optimize resource allocation, merge or move fire/police stations, or even suggest changes to roadway intersections to minimize accidents. However, some data – especially 911 call data — contains personally identifiable information such as names, addresses, phone numbers, driver's license numbers, medical status and other sensitive data related to individuals and businesses.

This example considers the three user roles mentioned above. The primary analysts are the county emergency manager and municipal department heads. The secondary analysts are county or municipal personnel who analyze broad event patterns that affect resource usage, such as arsons, accidents and emergency medical calls. The analyses do not require and should not contain personally identifiable information, but would have specific event location information and response unit identification data. Finally, the tertiary internal or external users include lower-level municipal employees and university researchers who perform high-level analyses. These users would not have access to personally

identifiable information, specific event locations or response unit identifiers beyond the types of response units (police, fire and emergency medical units).

A more detailed version of this scenario assigns different roles to users depending on their positions. For example, the county emergency manager would have access to all the data regardless of jurisdiction, but a town official may not be granted unrestricted access to data outside the official's municipality. Alternately, an attribute-based control system could accomplish the same purpose.

The benefit to using the proposed framework in this scenario is tighter access control over private data belonging to individuals and sensitive information related to businesses and government entities. Since many government data sources are subject to "freedom of information" type requests, the differential privacy aspect of the framework provides external users with access while protecting critical assets. Safeguarding personally identifiable information is important, but it is just as critical to avoid breaches that might expose the vulnerabilities of business or government installations.

2.2 Scenario 2: Healthcare Sector

In the healthcare sector, information sharing has become crucial to improving healthcare quality and outcomes, as well as lowering costs [17]. The benefits of sharing information must be balanced with security and privacy concerns, especially when healthcare personally identifiable information is involved. The U.S. Health Insurance Portability and Accountability Act (HIPAA) places strict requirements, including access control, for protecting healthcare personally identifiable information [16]. The constant barrage of successful attacks in the healthcare sector and the consequent data breaches reveal that the implemented access control mechanisms are inadequate [13]. Moreover, healthcare organizations incur significant penalties for one-time violations and repeat violations across all HIPAA violation categories [18].

Consider a healthcare scenario similar to the emergency management scenario discussed above. The healthcare scenario has trusted internal users (doctors, nurse practitioners and other medical personnel involved in direct patient care), internal users (medical personnel not involved in direct patient care) and internal/external users such as administrative personnel and researchers. However, the healthcare setting includes aspects that make the scenario more complex than the emergency services scenario.

In the healthcare setting, primary analysts have access to all the information about patients under their care. Unlike the emergency management scenario, the doctor-patient relationship excludes the possibility of a trusted user with unrestricted access to all the data related to patients.

Secondary analysts such as medical technicians would have access to data pertaining to their particular functions for short periods of time. While one would expect the doctor-patient relationship to be ongoing, ancillary medical personnel and even floor nurses would not need to access patient data after the patients are out of their care.

Tertiary analysts in medical administration have no need to see detailed health data such as laboratory reports and nursing notes, although they would need to know patient diagnosis and insurance information, thereby having access to personally identifiable information. External analysts such as medical researchers have no need to access personally identifiable information. Given the complexity of the healthcare scenario, attribute-based access control (ABAC) appears to be a better fit than role-based access control (RBAC) [4]. An attribute-based access control approach would also account for the temporal aspects of the healthcare sector.

In short, privacy requirements along with increased information sharing in the healthcare sector provide additional and compelling motivation for the enhanced access control framework proposed in this chapter.

3. Background

This section provides background information needed to understand the proposed framework. It discusses the key concepts of access control, blockchain and differential privacy that set the stage for the rest of this chapter.

3.1 Access Control

Access control models help ensure that only authorized users are allowed to perform previously-approved operations on objects. Numerous access control models have been developed over the years, each with its advantages and disadvantages. Software systems in the critical infrastructure sectors tend to use some variant of role-based access control [3, 14].

Role-based access control is based on five sets of entities: (i) subjects; (ii) objects; (iii) roles; (iv) operations; (v) and permissions; and two relations: (i) subject-to-role assignment; and (ii) permission-to-role assignment.

Central to role-based access control is the concept of a role, which specifies an organizational job function. Each role can also represent a set of responsibilities (or operations) associated with the job function. Instead of granting permissions individually to each subject, permissions are first associated with roles, following which roles are assigned to subjects based on their job functions.

The strengths of role-based access control arise from its simplicity of authorization administration and support for developing secure systems without requiring actual subjects. Because role-based access control is a static model, its access logic relies on a predefined set of associations of permissions to roles, which makes it unsuitable for use in environments and sectors that change dynamically. Also, role-based access control has inadequate protections against information disclosure and modification [14]. While security researchers have recently proposed models such as attribute-based access control to address problems with role-based access control, the new models have yet to gain widespread acceptance; as a result, role-based access control continues to be the dominant model used in critical infrastructure systems [3].

3.2 Blockchain

The decentralized and cryptographically secure characteristics of a blockchain enable it to serve as an immutable public ledger of records that are linked to each other [11]. Each block in the blockchain is a collection of transactions; for example, a block may be a set of financial transactions used for a cryptocurrency.

In a typical blockchain architecture, the blocks are linked to each other via hashing. All the transactions in a block are digitally signed by the involved parties with their private keys, and anyone can verify the owner using the owner's public key. For a cryptocurrency such as Bitcoin, transaction linkage also helps to keep track of the participants' balances. Each transaction is broadcast across the network and can be validated by each node in the network; nodes outside the network do not have permission to broadcast blocks. After the entire network validates a block with the chosen consensus algorithm that establishes agreement, the block is added to the blockchain by all the local nodes. This action results in all the network nodes having the same consistent data in the form of linked blocks – called the blockchain – without any central authority. An external node that wishes to join the network can build the blocks from the starting block to the most recent one with the help of its peers.

Smart contracts are often used in blockchain technology; these elements are executable code where any logic can be applied on all the nodes in the network [5]. In the context of this research, a smart contract contains the user information (roles and attributes) needed by the access control system.

A blockchain provides a decentralized method for enforcing rules and policies at all the network nodes. It also ensures that all the nodes follow and agree on the decisions, and maintains consistency of the data. Traditionally, access control systems have been centralized as opposed to distributed, with a single point of failure affecting and compromising the entire system. In contrast, a blockchain does not have a single point of failure. Blockchain technology has been used to secure data and preserve its privacy [20]. It can also be used to store access permission information.

3.3 Differential Privacy

Differential privacy as proposed by Dwork et al. [6] seeks to make the presence of an individual indistinguishable regardless of the background knowledge that an adversary may have about the dataset containing the individual's data. Hence, applying any analysis on the dataset gives almost the same results as when a record is added or removed from the dataset [1].

Let q be an arbitrary query with domain M and range P ($q : M \rightarrow P$) and let D and D' be two neighboring datasets that differ in one record. Furthermore, let f_q be a randomized function used to answer the query q. Then, f_q provides ε-differential privacy if for any $s \subseteq Range(f_q)$:

$$\Pr[f_q(D) \in s] = e^{\varepsilon} Pr[f_q(D') \in s]$$

Adding noise to the true answers is a common way to satisfy differential privacy. Consider a query q on a dataset D. If r is the true answer of query q, then the answer to the query that satisfies differential privacy is $r + y$, where y is random noise.

Several approaches have been proposed for generating noise. The most common approach is to draw the noise from a Laplace distribution with mean 0 and scale $\Delta f/\varepsilon$, where Δf is the maximum difference between $f_q(D)$ and $f_q(D')$ and ε is a parameter that controls privacy (as ε becomes smaller, the privacy level increases, but the accuracy decreases) [6].

Counting queries require an aggregating function to retrieve a specific value (count) of records that satisfy certain conditions [2]. Answers to these queries could exacerbate individuals' loss of privacy [7]. Because interactive settings provide better privacy than non-interactive settings, user access to data can be limited dynamically.

An unlimited number of sequential queries could still result in sensitive information being leaked, especially when the queries operate over related attributes. However, this issue can be resolved by setting up a workload of queries ahead of time and submitting them as a batch to adjust the level of added noise based on the given queries. Partitioning mechanisms permit sensitive areas of the vector of counts to have larger amounts of noise than other areas. This helps ensure more accurate answers when the workload has insensitive queries. The mechanism thus considers the sensitivity of the given set of queries, but is otherwise data independent [2].

4. Design and Implementation

This section describes the design and implementation of the proposed framework for enhancing access control using blockchain and differential privacy.

4.1 System Architecture

The system architecture assumes a role-based access control model with three major roles, primary, secondary and tertiary, corresponding to the three types of analysts discussed above. Other access control models are also possible, but role-based access control is sufficient for the purposes of this work. To address the goal of protecting sensitive information, the framework uses layered access as shown in Figure 1. Each layer receives input queries from the previous layer (higher in the figure), and invokes the appropriate access policy depending on the analyst's role.

The system comprises the following layers:

- **Client Layer:** The client layer accepts queries from the different types of analysts and passes the queries along with user credentials to the access control layer.

- **Access Control Blockchain Layer:** The access control blockchain layer is responsible for granting access to the requested data. The layer is

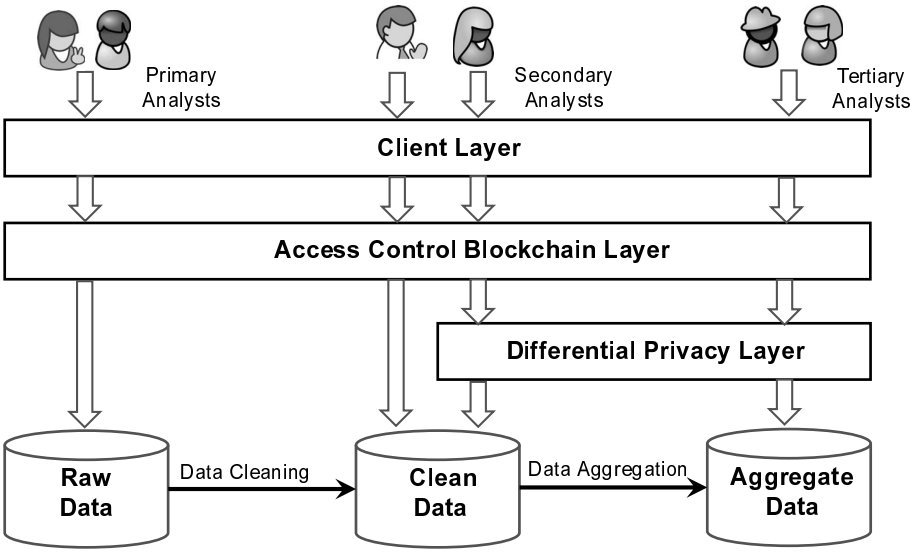

Figure 1. System architecture.

implemented using blockchain technology, where each user/node initiates a transaction on the blockchain network. The transaction is initiated after a smart contract is executed by the client layer. Based on the inputs provided to the smart contract, the user is provided with appropriate access permissions to complete the transaction. The smart contract runs on all the nodes that attempt to gain access to the data tables. The block is then broadcast across the blockchain network. All the network nodes validate the block, come to an agreement based on the chosen consensus algorithm and add the block to the blockchain.

The smart contract code cannot be modified by any of the users and the logic is always executed after a user attempts to access data. The access control system leverages blockchain technology and smart contracts in granting secure access, returning the key used to execute the queries. The main advantage of using smart contracts is that any complex access permission logic can be coded easily.

- **Differential Privacy Layer:** The differential privacy layer implements differential privacy techniques to provide further protection to sensitive information. The access control layer requires secondary and tertiary analysts to provide all their queries as a workload and then invokes the differential privacy layer. Based on the workload of queries, the actual results are modified to ensure individuals' privacy and operational privacy as discussed in Section 4.3.

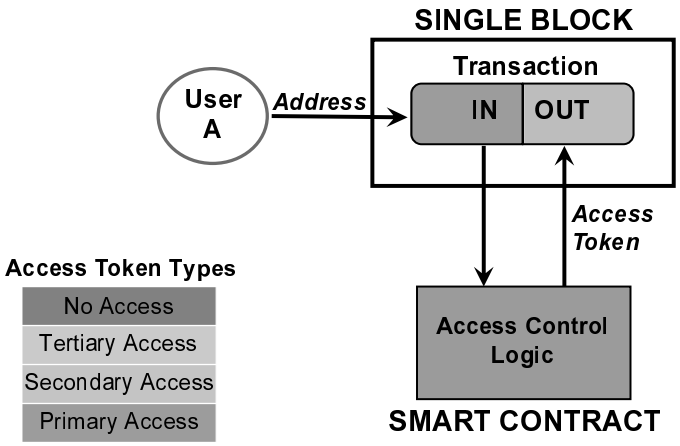

Figure 2. Single block example of smart contract interaction with access control.

4.2 Queries by Different Types of Analysts

In the case of primary analysts, the initial access layer works in a pass-through mode. Specifically, it simply passes the query straight to the raw database without any processing. This path is shown in the left-hand side of Figure 1.

In the case of secondary analysts, differential privacy techniques may be invoked depending on the nature of the query. Specifically, whether the query includes sensitive attributes or combinations of such attributes. If no sensitive attributes are present, then the query is passed through, as in the case of primary analysts. This path is shown in the middle of Figure 1.

In the case of tertiary analysts, the differential privacy layer is always used. The query path is shown in the right-hand side of Figure 1.

4.3 Implementation Details

Based on the architecture shown in Figure 1, role-based access control was implemented on the Ethereum blockchain [5] with the analyst roles stored in a smart contract. Analysts interact with the system via a public address to issue queries. The client layer receives an analyst's public address and then executes a call to the smart contract. The smart contract returns the analyst's role if access is granted; otherwise, access is denied.

Figure 2 illustrates the access control mechanism using a single transaction in a block. The analyst's address is input for the transaction and is used by the access control logic in the smart contract to look-up and return an appropriate access token for the analyst. The access token (primary access, secondary access, tertiary access or no access) returned by the smart contract is the transaction output, which is stored with the issuing analyst's address in a block.

Assuming that the access control layer approves, the client layer request is either sent directly to the data repository (for primary analysts) or is passed through the differential privacy module (for secondary and tertiary analysts). Of course, users who are not legitimate analysts are denied access.

As stated above, whenever the differential privacy module is invoked, the system requires analysts to present all the queries in a single batch or workload. This module employs the workload partitioning mechanism described in earlier work [2]. The mechanism takes the provided set of queries as a workload, along with the attribute values expressed as a vector of counts. The vector is partitioned into buckets based on the ranges of the given queries. The total count of each bucket is then anonymized by adding an amount of noise drawn from a Laplace distribution. After the count of each bucket is anonymized, it is split uniformly between the vector positions, producing a different private vector for answering the queries. The results, which are then returned to the user via the client layer, provide the desired additional privacy.

5. Preliminary Analysis

The proposed generic prototype can be used to implement a variety of access control models provided that the access control logic of the models can be programmed in the smart contract. Blockchain technology and differential privacy provide added protection for sensitive data.

However, the extra protection comes at a cost – in this case, additional overhead from the system components. First, the efficiency of the system is influenced by the complexity of the access control logic for the selected access control model. Depending on the application, the access control logic chosen and implemented can vary from simple to complex, and the execution time overhead varies accordingly.

Second, by requiring each node to process a transaction, blockchains can slow the system and are, therefore, unlikely to be scalable [10]. Additionally, underlying distributed blockchain network parameters such as the network load, consensus mechanism, processing power of the nodes, number of nodes and other distributed network parameters also affect system performance. Figure 3 shows the possible impact of access policies and blockchain overhead on the processing time.

Third, using differential privacy may affect system performance because of the processes that must be performed until the final answers are returned to a user. However, differential privacy may not be universally invoked for all users.

Other concerns regarding the security and privacy of the proposed framework include:

- The access control logic in the smart contract cannot be modified after it is deployed. For this reason, the smart contract code must be foolproof with no bugs and other programming flaws. If there are any issues, an adversary may be able to view the smart contract code and exploit flaws in its code.

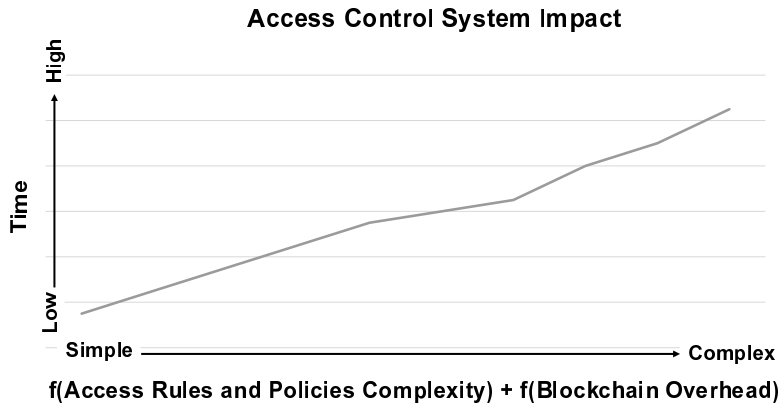

Figure 3. Effects of access control policies and blockchain on processing time.

- The data stored in the blockchain is visible to all the nodes in the network and any node can view the access permission of another node, leading to a potential privacy issue. However, because an analyst's address is stored with his/her access permissions, the system creates new addresses whenever a user query is sent to the system, preventing blockchain users from breaching analyst privacy. Current research is investigating the possible impact of scale on this approach.

- The framework only stores access permission details in the blockchain, not the real data. An application that uses the framework must ensure that the user who wishes to gain access interacts with the access control system to obtain the access permission; also, it should ensure that no adversary can circumvent the access control system. The blockchain ensures that unauthorized users cannot initiate transactions or change data in the ledger.

- The heart of the blockchain is the consensus mechanism. If more than half of the network nodes are not trustworthy, then there is a chance that adversaries may be able to take over the system. However, the possibility of this occurring is remote.

6. Conclusions

Effective information sharing, decision making and allocation are critical precursors to effective response, especially under conditions of widespread stress and overwhelming need. Even in such precarious times, it is important to protect individual, collective and, perhaps, operational privacy, and to secure critical infrastructure assets. Many current information sharing systems depend on outmoded controls that provide little certainty, and exhibit undesirable trade-offs between access control and responsiveness.

The framework described in this chapter addresses these fundamental concerns while supporting optimal decision making in evolving environments. However, a thorough exploration of the layered approach involving systematic testing and parameter optimization remains to be performed. Since questions still remain about system scalability and potential vulnerabilities, future research will focus on prototype testing under a range of parameter settings using a dataset containing twelve years of 911 call data from Monroe County, New York. The raw dataset contains personally identifiable information and sensitive critical asset information, which makes it possible to test the differential privacy module as well as the access control and blockchain layers.

Acknowledgements

Asma Alnemari acknowledges the support of the Ministry of Higher Education of the Kingdom of Saudi Arabia. This research was partially supported by the National Science Foundation under Grant No. DGE-1433736.

References

[1] R. Agrawal and R. Srikant, Privacy-preserving data mining, *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp 439–450, 2000.

[2] A. Alnemari, C. Romanowski and R. Raj, An adaptive differential privacy algorithm for range queries over healthcare data, *Proceedings of the IEEE International Conference on Healthcare Informatics*, pp. 397–402, 2017.

[3] S. Alshehri, S. Mishra and R. Raj, Using access control to mitigate insider threats to healthcare systems, *Proceedings of the IEEE International Conference on Healthcare Informatics*, pp. 55–60, 2016.

[4] S. Alshehri and R. Raj, Secure access control for health information sharing systems, *Proceedings of the IEEE International Conference on Healthcare Informatics*, pp. 277–286, 2013.

[5] V. Buterin, A Next-Generation Smart Contract and Decentralized Application Platform (`www.github.com/ethereum/wiki/wiki/White-Paper`), 2014.

[6] C. Dwork, F. McSherry, K. Nissim and A. Smith, Calibrating noise to sensitivity in private data, in *Theory of Cryptography*, S. Halevi and T. Rabin (Eds.), Springer, Berlin Heidelberg, Germany, pp. 265–284, 2006.

[7] C. Dwork and A. Roth, The algorithmic foundations of differential privacy, *Foundations and Trends in Theoretical Computer Science*, vol. 9(3-4), pp. 211–407, 2014.

[8] C. Lin, Z. Song, H. Song, Y. Zhou, Y. Wang and G. Wu, Differential privacy preserving in big data analytics for connected health, *Journal of Medical Systems*, vol. 40(4), 2016.

[9] A. Machanavajjhala, J. Gehrke, D. Kifer and M. Venkitasubramaniam, *L*-diversity: Privacy beyond *k*-anonymity, *Proceedings of the Twenty-Second International Conference on Data Engineering*, pp. 24–36, 2006.

[10] L. Mearian, Ethereum explores a fix for blockchain's performance problem, *Computerworld*, January 5, 2018.

[11] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System (`bitcoin.org/bitcoin.pdf`), 2008.

[12] President's National Security Telecommunications Advisory Committee, NSTAC Report to the President on Big Data Analytics, Washington, DC, 2016.

[13] Privacy Rights Clearinghouse, Chronology of Data Breaches: Security Breaches 2005 – Present, San Diego, California (`www.privacyrights.org/data-breaches`), 2018.

[14] R. Raj, S. Mishra, C. Romanowski, J. Schneider and S. Alshehri, Modeling threats: Insider attacks on critical infrastructure assets, poster presented at the *IEEE International Symposium on Technologies for Homeland Security*, 2017.

[15] C. Romanowski, R. Raj, J. Schneider, S. Mishra, V. Shivshankar, S. Ayengar and F. Cueva, Regional response to large-scale emergency events: Building on historical data, *International Journal of Critical Infrastructure Protection*, vol. 11, pp. 12–21, 2015.

[16] U.S. Department of Health and Human Services, Standards for Privacy of Individually Identifiable Health Information; Final Rule, *Federal Register*, vol. 67(157), pp. 53182–53273, August 14, 2002.

[17] U.S. Department of Health and Human Services, HITECH Act Enforcement Interim Final Rule, Washington, DC (`www.hhs.gov/ocr/privacy/hipaa/administrative/enforcementrule/hitechenforcementifr.html`), 2017.

[18] M. Winger, HIPAA increases financial penalties for repeat violations to address increasing healthcare data breaches, Zephyr Networks, Laguna Hills, California (`www.zephyrnetworks.com/hipaa-healthcare-data-breaches-financial-penalties`), February 10, 2013.

[19] T. Zhu, P. Xiong, G. Li and W. Zhou, Correlated differential privacy: Hiding information in a non-IID data set, *IEEE Transactions on Information Forensics and Security*, vol. 10(2), pp. 229–242, 2015.

[20] G. Zyskind, O. Nathan and A. Pentland, Decentralizing privacy: Using blockchain to protect personal data, *Proceedings of the IEEE Security and Privacy Workshops*, pp. 180–184, 2015.