Securing Data Provenance in Internet of Things (IoT) Systems

Nathalie Baracaldo[1(✉)], Luis Angel D. Bathen[1], Roqeeb O. Ozugha[2], Robert Engel[1], Samir Tata[1], and Heiko Ludwig[1]

[1] Almaden Research Center, IBM Research, San Jose, CA, USA
{baracald,bathen,engelrob,stata,hludwig}@us.ibm.com
[2] Dakota State University, 820 N Washington Ave., Madison, SD, USA
roqeeb.ozugha@trojans.dsu.edu

Abstract. The Internet of Things (IoT) promises to yield a plethora of new innovative applications based on highly interconnected devices. In order to enable IoT applications for critical and/or sensitive use cases, it is important to (i) foster their dependability by assuring and verifying the integrity and correctness of data processed in such applications, and (ii) adequately account for privacy and confidentiality concerns. For addressing these requirements, IoT systems can be equipped with *data provenance* mechanisms for maintaining information on the lineage and ownership of data. However, in order to provide secure and dependable IoT systems, provenance data needs to be sufficiently protected against tampering and unauthorized access. In this paper, we present a novel framework for cryptographic provenance data protection and access control based on blockchain technology and confidentiality policies.

Keywords: IoT · Provenance · Security · Blockchain · Keyless signature · Access control

1 Introduction

The Internet of Things (IoT) [8] has received significant attention in industry and in the academic community in recent years. Gartner forecasts that 6.4 billion connected things will be in use worldwide in 2016 [6]. They also predict that by 2020, more than 25% of identified attacks in enterprises will involve IoT [6], yet less than 10% of organizations' budget is dedicated to security.

IoT environments create an opportunity to collect information, run analytics and make important decisions that range from modifying the dosage of medicines for elderly patients to distributing budget for different projects in smart cities. Given the distributed nature of IoT systems, it is important to ensure that data used for analytics is actually generated by the expected entities. Failing to do so creates an opportunity for adversaries to manipulate decision making processes. For example, an adversary may fabricate data to ensure a smart city invests money in a targeted location or project.

K. Drira et al. (Eds.): ICSOC 2016 Workshops, LNCS 10380, pp. 92–98, 2017.
https://doi.org/10.1007/978-3-319-68136-8_9

In this context, maintaining the history of data creation, modification and transfer, a.k.a. *provenance*, has become an increasingly important requirement in IoT environments. *Data provenance* deals with the recording, management and retrieval of information about the origin and history of data [5]. In the smart city use case, provenance data may include information about the devices that collected information such as model and serial number, their location as well as the timestamp when observations are collected.

By recording and verifying the history of data, provenance data provides the ability to assure the integrity and correctness of systems, but also enables *auditing* and *digital forensics* and can help enforce *privacy* and *data sovereignty*. For the latter, the origin and lineage of data is instrumental for enabling fine-grained access control for sensitive information collected in IoT applications (e.g., health-related data).

In this paper, we present a framework to maintain and protect IoT provenance data. Our framework is designed for IoT environments and provides the following functionality: (i) Ensure provenance data integrity is protected using a fully distributed lightweight keyless blockchain component. (ii) Ensure confidentiality of provenance data by providing custom access control to multiple stakeholders when necessary. Our solution allows the enforcement of fine-grained access control policies over provenance data. Because in IoT environments data and its provenance data flow is not easy to control, our solution makes use of cryptographic techniques to ensure confidentiality. (iii) Finally, our architecture is designed to allow high availability of provenance data.

In the following, we present the state of the art and then we introduce the proposed framework.

2 State of the Art

Multiple researchers have recognized the need to provide protection against forgery, fabrication and leakage of private information in provenance systems. Braun *et al.* present multiple scenarios in which data and its associated provenance information require different protection [2]. Muniswamy-Redd *et al.* highlight the necessity of protecting provenance data generated in cloud environments against forgery, fabrication and leakage of private information [12].

Cryptographic techniques have been proposed to protect forgery and confidentiality of provenance data [1,9,14,17]. Hasan *et al.* propose the use of broadcast encryption to ensure the confidentiality and integrity of provenance records and their related provenance chains [9]. The model defined in [14] utilizes a mutual agreement signature-based approach to provide confidentiality, integrity and availability of links between provenance data records. The model captures digital acyclic graph provenance and information sharing between users. Other approaches [1,17] use private-public key infrastructure to prevent forgery and confidentiality leakage of provenance data.

The above mentioned approaches uniquely focus on the cryptographic primitives and do not provide an architecture suitable for IoT environments. Additionally, they do not explicitly integrate access control policies to protect provenance

data, making it difficult to manage changes in access control policies. Moreover, because provenance data is not stored in a distributed fashion, adversaries may be able to repudiate their latest creation or modification of data. To address these drawbacks, in this paper we integrate access control policies with cryptographic enforcement to protect the confidentiality and privacy of provenance data in a easy to manage fashion, and provide an architecture suitable for IoT environments.

3 Terminology and Requirements

In this section we present the requirements that led to the design of our system, but first, we present the terminology used throughout the paper. We assume a provenance model that describes the lineage of *data points*. A data point is a *uniquely identifiable* and *addressable* value in the context of the IoT system. A data point is specifically different from basic readings or other data flowing in the system in that it is addressable. The provenance information describes the context of the creation or modification of data points, including information about involved agents (e.g., a device containing several sensors), execution context, time, and location information.

We identify the following requirements for secure provenance in IoT environments:

1. Adversaries should not be able to tamper, fabricate provenance data or link valid provenance data to a different data point.
2. Provenance data should be highly available to entities that need to verify it.
3. The architecture should adapt to meet resource constraints in different environments.
4. The framework should allow the specification of policies that define the stakeholders or entities that may access certain provenance data. The goal here is to provide fine-grained confidentiality protection of provenance data by limiting the access to pre-defined stakeholders, e.g., auditors.

Based on these requirements, we now introduce the proposed architecture.

4 Secure Provenance Framework

In this section, we present the proposed framework depicted in Fig. 1. We make the following assumptions:

1. Devices and sensors are registered and authenticated with their assigned gateway(s). Similarly, gateways and other analytic services use best practices to communicate (SSL) and store data (data-at-rest encryption).
2. Gateways and agents that may modify or create information have assigned a cryptographic asymmetric key pair.
3. Provenance data may have different confidentiality protection requirements. These requirements are specified using an attribute-based access control policy, e.g., [7], which defines who may access different types of provenance data.

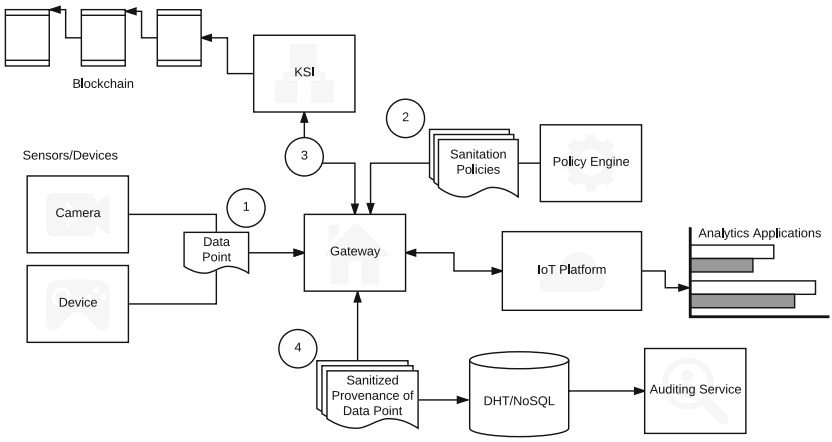

Fig. 1. Secure Provenance System Architecture

Our framework is composed of the following modules: a *Policy Engine*, *Keyless Signature Infrastructure Module* (*KSI*), an *IoT Platform Module* and an *Auditing Service*.

The *Policy Engine* is used to maintain the policies that define how to protect provenance data. These policies are enforced through attribute-based encryption (ABE) [16] to ensure only authorized users can gain access to protected provenance data. In this scheme, protected provenance data is stored encrypted and each ciphertext is associated with a combination of attributes and private keys of authorized users. Users may decrypt a ciphertext (in this case provenance data) only when their private keys and their associated set of attributes "match" the ciphertext.

The *KSI* Module is used to provide provenance integrity, and the *IoT Platform Module* serves as the management point for our applications. This follows very closely the implementation of IBM's Bluemix IoT Platform-as-a-Service model [10]. We provide an *Auditing Service*, which interacts with the storage backend, and can build provenance graphs for data points by linking the data points to their provenance data.

Figure 1 shows how provenance data flows through our architecture:

1. Sensors (e.g., cameras) or Devices (e.g., phones) may generate a data point, which is initially sent to the Gateway. In order to provide scalability, we assume sensors and devices keep a list of peer Gateway nodes.
2. Gateways are the point of entry into our provenance framework, so it is of utmost important to maintain their availability. Gateways are managed in a decentralized manner akin to the way traditional P2P systems operate (e.g., BitTorrent). Gateways constantly publish their IPs and services to peers so that any one node that needs to subscribe may interact with it. Sensors may send their data to any of their peer Gateways. Once data is received by a Gateway, it polls the Policy Engine and requests the *Sanitation Policy* mapped to

the specific *type of data* it needs to process (and caches it). The Sanitation Policy contains the set of cryptographic keys that the Gateway should use, if any, to protect provenance data. For instance, if geographic information needs to be protected, an attribute-based policy that specifies what entities in terms of attributes they hold may access geographic information for a given type of data is stored in the Policy Engine. The Policy Engine is in charge of generating and maintaining cryptographic keys used for each attribute-based policy. Thus, when the Sanitation Policy is retrieved, the Gateway will encrypt the geolocation information with the right set of keys. We call this *sanitized provenance data*.

3. We provide data integrity by means of a *KSI* [3,4] as it is highly scalable and has been mathematically proven to protect against several tampering attacks. We use a *blockchain* [11,13] as a second layer in order to protect and publish the top root of the *KSI* tree. Data points and their respective provenance data are linked through unique identifiers (*UUIDs*) derived from the hash of their contents in a similar fashion as transaction IDs [13]. Once the metadata has been sanitized, it is sent to one (or several) KSI peer(s) to sign and enter the provenance data into the ledger. This step ensures the integrity protection of the provenance data, thereby preventing any sort of tampering from happening. As with KSI systems, validation of data integrity is a straight forward and inexpensive task.
4. Finally, the sanitized provenance data along with the KSI signature is persisted onto a DHT (in the event of a peer-to-peer storage model) or a NoSQL DB hosted in a cloud provider. The DHT model allows for gateways to participate in a peer-to-peer distributed storage model, where they may all share some of their storage. This allows us to provide high availability by distributing the data across multiple peers. We use erasure coding to minimize the amount of storage consumed by the DHT as in [15]. Similarly, we could leverage the high availability of traditional storage backends (e.g., NoSQL DBs, DHTs, etc.) promised by cloud providers (over 5–9s reliability service level agreements) [15]. We use the provenance data's *UUIDs* as storage keys when sent to our storage backend.

Finally, we note that any entity that uses as input or transforms data produced by the IoT environment, such as analytic services that aggregate information, can also maintain provenance information of their observations using our architecture.

5 Conclusions

In this paper, we presented a framework for protecting provenance data in IoT environments that addresses the requirements of (i) tamper prevention, (ii) high availability and (iii) access control for provenance data while ensuring that (iv) even constrained devices can be part of the system. Using a fully distributed, lightweight and keyless signature infrastructure in conjunction with attribute-based encryption and blockchain, the framework allows for the enforcement of

fine-grained access control policies while assuring and enforcing the integrity of the provenance data. To the best of our knowledge, our proposed framework is the first to provide a lightweight, scalable architecture for protecting provenance data that enforces confidentiality of provenance data at the point of transmission in IoT systems.

Our ongoing research efforts focus on implementing and evaluating the aforementioned framework as well as combining it with mechanisms for secure deployment and verification of code in IoT environments (i.e., for verifying the integrity of producers of provenance data).

References

1. Gadelha, J., et al.: Kairos: an architecture for securing authorship and temporal information of provenance data in grid-enabled workflow management systems. In: eScience 2008 (2008)
2. Braun, U., Shinnar, A., Seltzer, M.I.: Securing provenance. In: HotSec (2008)
3. Buldas, A., Kroonmaa, A., Laanoja, R.: Keyless signatures' infrastructure: how to build global distributed hash-trees. In: Riis Nielson, H., Gollmann, D. (eds.) NordSec 2013. LNCS, vol. 8208, pp. 313–320. Springer, Heidelberg (2013). doi:10.1007/978-3-642-41488-6_21
4. Buldas, A., Truu, A., Laanoja, R., Gerhards, R.: Efficient record-level keyless signatures for audit logs. In: Bernsmed, K., Fischer-Hübner, S. (eds.) NordSec 2014. LNCS, vol. 8788, pp. 149–164. Springer, Cham (2014). doi:10.1007/978-3-319-11599-3_9
5. Buneman, P., Khanna, S., Wang-Chiew, T.: Why and where: a characterization of data provenance. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 316–330. Springer, Heidelberg (2001). doi:10.1007/3-540-44503-X_20
6. Gartner: Gartner says worldwide IoT security spending to reach $348 million in 2016 (2016). http://www.gartner.com/newsroom/id/3291817
7. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89–98. ACM (2006)
8. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gen. Comp. Sys. **29**(7), 1645–1660 (2013)
9. Hasan, R., Sion, R., Winslett, M.: The case of the fake picasso: preventing history forgery with secure provenance. FAST **9**, 1–14 (2009)
10. IBM: IBM bluemix (2016). https://console.ng.bluemix.net
11. Linux Foundation: The Hyperledger Project (2016). https://www.hyperledger.org
12. Muniswamy-Reddy, K.K., Seltzer, M.: Provenance as first class cloud data. ACM SIGOPS Oper. Syst. Rev. **43**(4), 11–16 (2010)
13. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008). https://bitcoin.org/bitcoin.pdf
14. Rangwala, M., Liang, Z., Peng, W., Zou, X., Li, F.: A mutual agreement signature scheme for secure data provenance. Environments **13**, 14
15. Rodrigues, R., Liskov, B.: High availability in DHTs: erasure coding vs. replication. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 226–239. Springer, Heidelberg (2005). doi:10.1007/11558989_21

16. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005). doi:10.1007/11426639_27
17. Wang, X., Zeng, K., Govindan, K., Mohapatra, P.: Chaining for securing data provenance in distributed information networks. In: MILCOM 2012, pp. 1–6 (2012)