Review of "Blockchain-based Data Provenance for the Internet of Things" By Marten Sigwart [1]

Gagan Chowdary Chatu*,
CS5378 Advanced Computer Security
Texas State University, San Marcos, Texas, USA
Email: *akl103@txstate.edu

The Internet of Things (IoT) transformed everyday life with devices like GPS, Smart Appliances etc. The data collected by these devices can find application in increasing number of use cases. To use data generated by these devices, data needs to be trustworthy. Data provenance system could help in making sure the data is trustworthy. It helps in tracking the data attributes like who created, who modified for a particular data point. Traditionally in distributed environments, participants must trust in each other or a third party to securely store data. Trusting in a centralized system can be avoided by using blockchain technology because it is tamper-proof and distributed in nature. IoT has different sets of use cases and application areas. So, the solution should accommodate for the diversity [2]. Existing research focuses on a specific use case [3] [4]. This paper proposes a generic framework that can be applied to various use cases.

Advantages of a generic framework are the easier adoption of provenance concepts by new use cases and interoperability of application that use the framework. The contribution of the paper is as follows:

- Functional and non-functional requirements of the data provenance framework.
- Conceptualize and implement an IoT data provenance framework using smart contracts.
- Evaluation of the framework regarding defined requirements with a proof of concept using Ethereum smart contracts.

Functional and non-functional requirements of generic framework are defined as follows:

1) *Provenance Abstraction:* The framework should provide a generic way of capturing data, querying and storing data functionalities which can later be adopted by specific requirements.
2) *High-level and low-level Provenance:* Low-level data points are the data generated by sensors whereas high-level data points do not have a single source of origin. They represent abstract concepts.
3) *Completeness:* A provenance record is complete if all relevant data about the data point is collected.
4) *Creation of lineage:* Provenance record for a data point is created from the last provenance record of the same data point. This enables to create a lineage of records.
5) *Derivation:* A provenance record entails references to the provenance records of the data points that led to its creation.
6) *Provenance of modification of data points:* This framework allows us to track history of modifications, on data points.
7) *Parallel Provenance:* Multiple provenance records of the same data point can exist at the same time.
8) *Integrity*: Provenance records must be protected from modification by adversaries.
9) *Availability:* Data can be accessed when needed.
10) *Privacy:* Sensitive data in provenance records must be protected from unauthorized access.
11) *Scalability:* Provenance system should have low overhead as some IoT devices are resource constrained.

This framework uses layered architecture and has three layers as visualized in Fig 1. Each layer represents a level of abstraction and responsibilities within the framework. They are storage layer, generic provenance layer, specific provenance layer.

1) ***Storage Layer:*** This layer is responsible for low-level storage of provenance records. It contains a definition of a provenance record. It also performs operations like create, retrieve, update, delete on provenance records. Delete operation cannot be performed on a blockchain as it is immutable, instead record is invalidated. Once a record is invalidated it cannot be used as an input of the next provenance record.
2) ***Generic Provenance Layer:*** This layer provides generic functionality that is universally applicable to many use cases. Responsibilities of this layer are as follows:
   - *Ownership of data points:* Even though blockchain ensures integrity, once records entered the system,mechanisms need to be in place to ensure that the records that enter the system are correct. As a first step, we aim to prevent the creation of provenance records by arbitrary clients, i.e., if client A generates some data point *dp0*, we want to make sure that only client A is able to create provenance records for

dp0. Thus, the concept of ownership is defined. Only the owner of the data point may produce more records of it. If any other client tries to create a provenance record system generates error.

- *Associating provenance records with data points:* This layer further links data points with provenance records. The framework provides information about associated provenance records of specific data points. It is achieved by providing a mapping of a data point with a set of associated provenance records. *addr(dp)* – {*addr(prov1(dp)), addr(prov2(dp), . . .* }. here, prov1, prov1 are parallel provenance records for the same data point.

3) ***Specific Provenance Layer:*** This layer utilizes two other layers and allows customization of provenance model according to their needs. Access control happens on two levels. First, A specific contract defines which parts of the generic provenance layer's API are exposed. Some requirements may not need all the functionalities provided by generic provenance layer. Specific provenance layer can hide those functions by removing them from public access. Second, access control is relevant for controlling the ownership of data points. This layer is responsible for ownership of data points.

The functional and nonfunctional requirements of this framework are tested with a proof-of-concept (POC) implementation using Ethereum smart contracts. This POC is tested on the public Ethereum test network Rinkeby and Ropsten. They are specifically chosen for their close resemblance to main Ethereum network.

The presented framework fulfilled all functional requirements and most non-functional requirements like integrity and availability but has some limitations on scalability and privacy. There are multiple blockchain concepts trying to address these limitations. In future, the author would evaluate those solutions and address scalability and privacy problems in the context of data provenance framework.

REFERENCES

[1] M. Sigwart, M. Borkowski, M. Peise, S. Schulte, and S. Tai, "Blockchain-based data provenance for the internet of things," in *Proceedings of the 9th International Conference on the Internet of Things*, ser. IoT '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi-org.libproxy.txstate.edu/10.1145/3365871.3365886

[2] H. O. et al, "Data provenance model for internet of things (iot) systems," *Service-Oriented Computing–ICSOC 2016 Workshops: Revised Selected Papers*, pp. 85–91, 2017.

[3] G. C. P. et al, "Blockchain-assisted information distribution for the internet of things," *2017 IEEE International Conference on Information Reuse and Integration*, pp. 75–78, 2017.

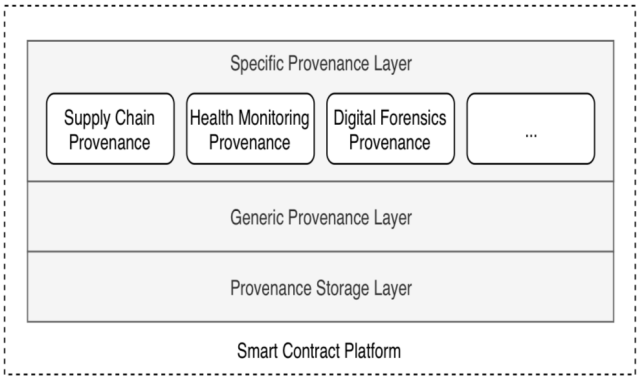

Fig. 1: Architecture of Data Provenance System [1]

[4] N. B. et al, "Securing data provenance in internet of things (iot) systems," *Service-Oriented Computing–ICSOC 2016 Workshops: Revised Selected Papers*, pp. 92–98, 2017.