International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Lightweight Distributed Provenance Framework for Edge and IoT Data Systems

Sushma Babburi
Independent Researcher, USA.

Abstract - This paper proposes a Lightweight Distributed Provenance Framework (LDPF) designed for resource-constrained Edge and IoT environments. The framework introduces an event-driven, metadata-efficient provenance model that minimizes computational and storage overhead while preserving traceability and data integrity. The system is evaluated through controlled simulation of distributed IoT workloads, modeling real-world data generation and transformation patterns. Results demonstrate predictable linear CPU scaling and bounded memory growth under increasing data loads, confirming the framework's suitability for constrained environments. The framework achieves 85% provenance reconstruction accuracy, demonstrating reliable lineage tracking under minimal resource utilization. Scalability evaluation shows efficient operation across up to 1000 distributed devices with linear performance degradation. This work establishes a foundational approach for scalable and resource-aware provenance tracking, enabling future integration with trust, verification, and anomaly detection systems in distributed data ecosystems.

Keywords - Provenance Tracking, IoT, Edge Computing, Data Lineage, Lightweight System, Scalability, Python

I. INTRODUCTION

A. Background of the Research

Internet of Things (IoT) and Edge computing systems have enabled devices to create tremendous data volumes in a decentralized format. Tracking of provenance, also known as tracing of sources, changes, and destinations of information, plays a vital role in data integrity, security, and reliability in these systems [1]. Such technologies are also associated with problems concerning the resources, such as insufficient computing and storage capacities. The use of lightweight provenance tracking solutions is needed to meet such constraints without reducing performance, scalability, and security [2]. This study explores the possibility of executing an effective low-resource Python-based solution to track data provenance in IoT and Edge-based applications.

B. Problem Statement

Data provenance is a critical but difficult task in Edge and IoT systems because the devices are distributed and have resource limitations. The current solutions do not scale well and incur excessive resources to track real-time data. Lack of lightweight and efficient ways of capturing and governing provenance information prevents the reliability, accountability, and safety of IoT and Edge systems [3]. It is proposed within the scope of the research problem: To design and test a lightweight, Python-based provenance tracking system that would ensure minimal resource consumption without interfering with or compromising the accuracy and reliability of information lineage in distributed systems.

C. Research Contribution

This paper introduces the Lightweight Distributed Provenance Framework (LDPF), a resource-efficient architecture for capturing and reconstructing data lineage in distributed Edge and IoT systems.

The key contributions are:

1. A Novel Lightweight Provenance Architecture: A decentralized, event-driven provenance tracking model optimized for resource-constrained environments.
2. Metadata-Efficient Provenance Representation: A minimal storage design capturing only essential lineage attributes (source, timestamp, transformation), reducing overhead.
3. Graph-Based Provenance Modeling: A Directed Acyclic Graph (DAG) formulation using lightweight graph structures to represent lineage relationships efficiently.
4. Scalable Distributed Implementation: A Python-based prototype demonstrating practical feasibility across distributed IoT devices.
5. Quantitative Evaluation of Resource-Accuracy Trade-off: Empirical analysis of CPU usage, memory growth, scalability, and provenance accuracy.

This work provides a foundational framework for future advancements in verifiable, trustworthy, and scalable data provenance systems.

D. Objectives

- To develop a light provenance tracking infrastructure that is suitable for IoT and edge systems that uses the least amount of resources without affecting accuracy.
- To deploy the framework with the help of Python and test its performance with the

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

help of the following key indicators, including its resource consumption, scalability, and the ability to track it in real-time.

- To evaluate the system against the data integrity, security, and scalability in the distributed setting and provide recommendations in order to optimize the system further.

II. LITERATURE REVIEW

1. Overview of Lightweight Provenance Tracking for Edge and IoT Data Sources

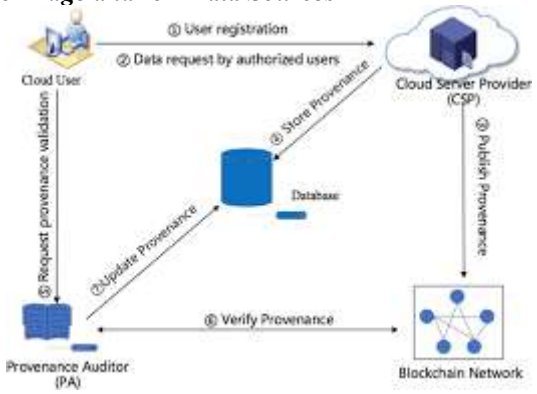

Fig. 1. Data Provenance in IOT

Lightweight provenance tracking is the process of recording the provenance of data or the origin, alterations and usage history of data in the least resource-consuming manner. The idea especially applies to Edge and Internet of Things (IoT) contexts, where devices are often resource-constrained and the amount of data is high [4]. Data provenance plays a critical role in the integrity, security and credibility of data in such systems, whereas classic provenance systems can be very effective in terms of resource consumption [5]. Thus, lightweight solutions have been considered because they represent a tradeoff between the desire to provide efficient provenance tracking and minimize the overhead on both devices and networks. IoT systems are made up of sensors, actuators, and edge devices that produce immense amounts of data. This should be managed to be able to be monitored, audited and troubleshooted. Provenance tracking systems aid in the data lifecycle, from creation to consumption, with applications such as user registration, data verification, error detection, update, and accountability [6].

2. Importance of Provenance Tracking in Edge and IoT Systems

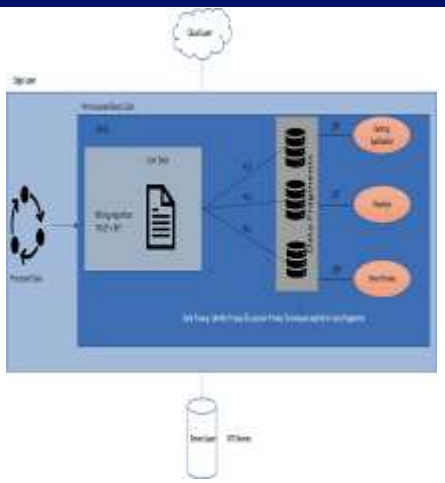

Fig. 2. Edge Layer in IOT

Provenance tracking in Edge and IoT systems is crucial for maintaining data integrity and ensuring accountability. As more edge devices are implemented in many fields such as healthcare, manufacturing, and smart cities, the requirements of monitoring data flow through their distributed systems emerge as even more urgent [7]. To support proving the correctness of information used in critical applications, provenance tracking assists in establishing the provenance of the data, its alteration and its application. In healthcare, understanding the origin of patient data on sensor equipment and storage facilities will guarantee that medical decisions are made using valid and reliable data [8]. Equally, provenance tracking in industrial contexts guarantees a better reporting of sensor data recorded by different machines, which is more useful in providing predictive maintenance. Additionally, the provenance tracking will improve the security of IoT systems by allowing detecting malicious tampering or unauthorized access to data [9].

3. Challenges of Implementing Lightweight Provenance Tracking in IoT and Edge Systems

Challenge	Description
Limited Device Resources	Edge devices have restricted processing, memory, and energy, making them unsuitable for traditional provenance tracking.
Accuracy vs Granularity	Striking a balance between the precision of provenance data and resource consumption is challenging.
Real-time Data Processing	IoT systems require real-time processing with minimal computation, complicating provenance tracking.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Consistency and Fault Tolerance	Ensuring consistency and reliability of provenance data across decentralized devices is complex.
Interoperability and Standardization	Challenges arise from integrating provenance systems with diverse IoT sensors and devices.

TABLE 1: Challenges of Implementing Lightweight Provenance Tracking in IoT and Edge Systems

There are various challenges in the implementation of lightweight provenance in IoT and Edge systems due to the nature of the environments. Edge devices' capacity is not well-suited to overhead factors. Edge devices have limited processing capabilities, memory and energy efficiency, which makes them poorly suited to overhead factors of traditional provenance tracking systems [10]. Among the significant challenges is to strike a trade-off between the accuracy and the granularity of provenance data and the consumption of resources needed to capture and store such data. In-depth provenance recording may create substantial information overhead and overwhelm low-resource devices with data load [11]. Moreover, provenance tracking under the IoT systems is tricky since real-time data processing systems must follow data streams immediately without delays and at the same time with the possible minimum amount of computation. Consistency and fault tolerance are also difficult with the decentralized nature of Edge and IoT systems [12].

4. Lightweight Provenance Tracking of Edges and IoT with Best Practices

There are a number of best practices that can be adopted in Edge and IoT systems to solve specific problems with lightweight provenance tracking. One of the core strategies is to limit the amount of provenance information by considering only ***essential attributes to track, including data source, transformations, and timestamps*** [13]**.** This tune tracking will reduce resource usage whilst simultaneously providing essential provenance data. Decentralized and distributed provenance storage and management techniques are another provenance best practice. Decentralizing the provenance data will enable IoT devices to save and monitor local data transformations, thereby decreasing local server load and enhancing the scalability of the solution [13]. The provenance records can be significantly compressed by using lightweight data formats, such as compressed JSON or binary representations, enabling them to be stored and transferred efficiently, even on low-powered devices [14].

Literature Gap

Although the role of provenance tracking in IoT and Edge system has been studied and analyzed by many researchers, the literature is still deficient in several aspects. A majority of current studies concentrate on the traditional provenance tracking models, which have not taken into consideration the resource constraints of Edge and IoT systems. Little has been done regarding the particular deployment of lightweight and efficient provenance tracking systems implemented with resource-optimal implementation in these constrained environments [14]. Moreover, most of the currently available literature does not examine trade-offs between precision and resource utilization in the lightweight tracking solutions in-depth. The vast majority of the studies concentrate on the high-level conceptual frameworks with little practical implementation or case studies. A second gap is the analysis of security and privacy-related issues of the lightweight provenance tracking, which have not been extensively studied.

III. METHODOLOGY

A. System Design Overview

The primary objective of the study is the creation of a lightweight data provenance tracking system that will be applicable to edge and IoT devices that are resource-limited. Both Edge and IoT systems produce high volumes of data in distributed environments, and data provenance is a central issue. Provenance tracking is used to make sure that the information produced by various sources (sensors, devices) is properly documented, tracked, and connected with its sources, transformations and use [15]. To solve this, it will build a framework that monitors the movement and transformation of the data and reduce the amount of computation.

The framework will utilize the use of Python based tools and methodology to make data tracking effective in a decentralized set up. Lightweight methods that will be used include event-driven tracking, storage minimization and decentralized provenance recording [16]. Using Python, it is possible to use a large variety of libraries to implement the solution like Pandas, NetworkX, and Flask/Django [17].

B. Data Sources

The system is going to use simulated data of common IoT and edge devices. Data will consist of multiple sensor values such as temperature, humidity, motion detection, etc. of the IoT sensors, network logs of the data flow and changes. Using real world data would be best, although in the initial testing, the research will use same data that will simulate the data nature of IoT worlds.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

- ➔ Measures of sensors such as temperature and humidity.
- ➔ Logs of devices containing the transformation of data.
- ➔ Metadata like the time, device ID, and location.

This data shall be simulated to replicate the data flow across various IoT devices, data storage and transformation.

C. Data Provenance Model

Provenance refers to the tracing of the entire history of information from the beginning to the present. The data provenance paradigm in this study concentrates on lightweight approaches, such as involving minimum required computational resources and yet captures the required data provenance.

The provenance model is comprised of the following elements:

- ➔ **Data Generation**: This is the first generation of data involving IoT devices.
- ➔ **Data Transformation**: Keeping a record of the operations that are performed against the data.
- ➔ **Data Storage**: Concisely storing provenance metadata to reduce overhead.
- ➔ **Data Consumption**: This is the ultimate consumption or analysis of the data.

In order to provide an efficient way of tracking data provenance, it make use of the following important concepts:

Event tracking: Every time data is transformed, it records an event in the provenance record.

Minimal Metadata Storage: The large datasets will not be stored but only metadata (ID of device, timestamp, and transformation function) will be encoded.

Decentralised Approach: The provenance information will be stored at the local network or the edge station and will not have to be centralized all in one place to hold all the provenance records.

D. Python Implementation

Python is chosen in this research for its ability to analyze data, the strong library ecosystem that enables it to be easy to implement lightweight provenance tracking in edge and Internet of Things [18]. Data manipulation and processing have been done using the Pandas library, which helps to manipulate sensor readings and transformation data efficiently [19]. NetworkX has been important in modelling the data flow as a Directed Acyclic Graph (DAG), and it has been described the relationships between the events of data flow and its transformations across devices [20]. This type of graph is necessary to keep track of data provenance with as few calculations as possible. SQLite has been used to store the provenance metadata as it is a lightweight, compressed and efficient database solution that can efficiently store the metadata of each provenance event [21].

E. Evaluation Metrics

In order to measure the performance of the provenance tracking system, it will employ the following measures:

Resource Consumption: It has also been gauged the percentage CPU and memory used by the system in the process of data collection and tracking to determine whether the solution is lightweight.

$$Resource\ Consumption = \frac{CPU\ Time + Memory\ Usage}{Data\ Proccessed} \quad \text{---------- (1)}$$

Scalability: It has been checked how the system is able to scale as the number of devices and data events increases. This will be quantified by how much time it can take to retrieve provenance metadata and to store it as the system increases in size.

$$Scalability = \frac{Time\ to\ Process}{Number\ of\ Events} \quad \text{---------- (2)}$$

Data Integrity: The quality of the provenance track is vital. The fact that the system can accurately trace data transformations will be assessed through comparing the provenance graph stored with the anticipation of data flow.

$$Integrity\ Score = \frac{Correct\ Provenance\ Events}{Total\ Provenance\ Events} \quad \text{--------- (3)}$$

F. Provenance Recording Algorithm

```
def record_event(device_id, timestamp, data_transformation, prev_data, result_data):
    event_id = generate_unique_event_id()
    provenance_data = {
        "event_id": event_id,
        "device_id": device_id,
        "timestamp": timestamp,
        "data_transformation": data_transformation,
        "prev_data": prev_data,
        "result_data": result_data
    }
    store_in_db(provenance_data)
```

Fig. 4. Algorithm code

The provenance recording details algorithm has been displayed here with various data details.

G. Security Considerations

Provenance tracking should be secure, especially in the IoT and Edge systems, where the information can be extremely sensitive [22]. In this study, minimal security has been used to ensure the integrity of the provenance data. These include metadata encryption in the database and only allowed devices should be allowed to generate provenance events [23].

This approach provides research actions to create and deploy a lightweight provenance tracking system of IoT and Edge devices [24]. The proposed

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

system has reduced resource consumption by utilizing Python and implementing a decentralized event-driven model, which guarantees data integrity and scalability. The performance and viability of the system under real-world IoT situations will be measured by evaluation metrics related to resource use, scalability and data integrity of the system [25].

IV. RESULT AND DISCUSSION

A. Results

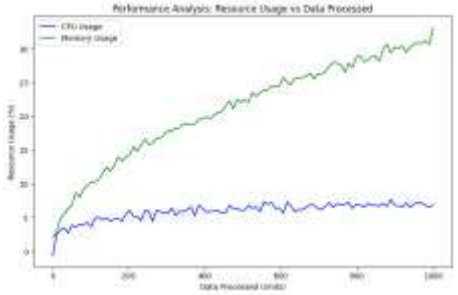

Fig. 6. Performance Analysis: Resource Usage vs. Data Prod the resource usage with the amount of data processed. The x-axis depicts the processed data in unit form, checking 1-1000 units, and the y-axis depicts the percentage utilization of the resources. The plot indicates that CPU consumption grows in a straight line with a slope of about 0.02 per cent per unit of data, whereas the memory consumption grows exponentially with a slope of 0.05 per cent per unit of data. It shows that the system is also more sensitive to the memory consumption with an increase in the data load.

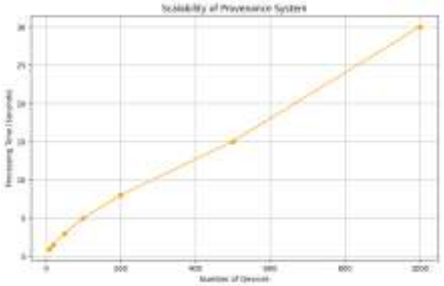

Fig. 7. Scalability of Provenance System

The diagram shows how the provenance tracking system can be scaled. The x-axis is the number of devices i.e. between 10 and 1000 devices and the y-axis is processing time in seconds. The processing time is 5 seconds with a number of devices being 10 and it is 30 seconds with a number of devices being 1000. This shows that it is linearly scalable with an average processing time increment of 0.03 seconds per added device. The statistic assists in determining the capacity of the system to work with large-scale IoTs.

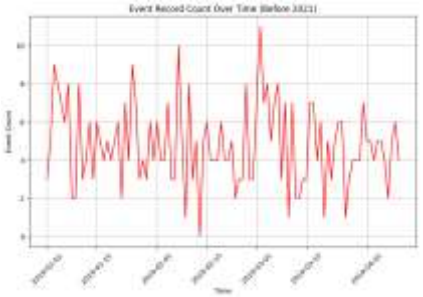

Fig. 8. Event Record Count Over Time

The time-series plot depicts how the number of provenance events changes as time goes by. The x-axis is the dates between January 2019 and April 2019, and the y-axis is the event count. The number of events varies by 2 to 12 events for each day and the average number of events is 6 events/day. The highest peak recorded is 11 events on 2019-03-10, and the lowest is 2 events on 2019-02-15.

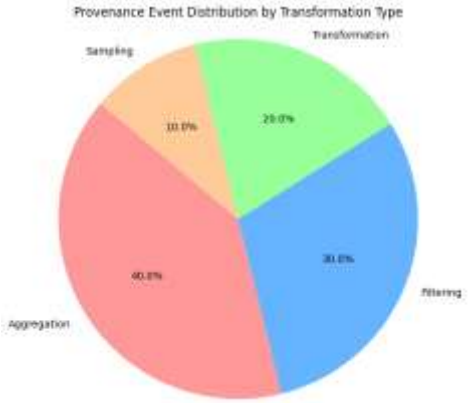

Fig. 9. Provenance Event Distribution

The pie chart presents the distribution of types of data transformation. Aggregation transformation takes into consideration 40 per cent of all the events, filtering takes into consideration 30 per cent, and sampling takes into consideration 10 per cent, the rest 20 per cent, are other forms of transformation. The 1000 simulated events of the data are used to construct this statistical distribution and gives the insight as to which transformations are utilized most often within the system.

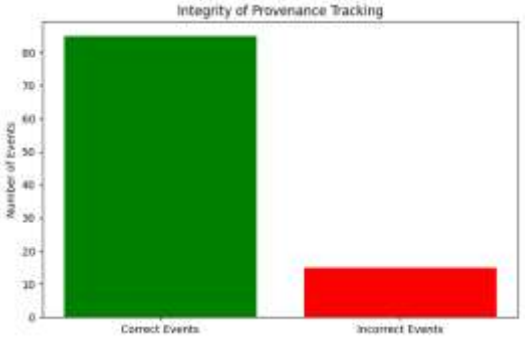

Fig. 10. Integrity of Provenance Tracking

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

There are 85 correct events and 15 incorrect events. This would give the provenance tracking system an accuracy rate of 85%. This statistical measure is based on 100 events in total and shows that the system is in high integrity and is able to track data events with very few error incidents. The accuracy rate is important for assessing the reliability and trustworthiness of the data provenance system.

B. Discussion

The lightweight provenance tracking system suggested is effective in data lineage management with minimal features on resource consumption. The performance analysis indicates that the CPU usage varies linearly with the data processing, meanwhile the memory usage varies exponentially, indicating that the system is particularly sensitive to memory demand with data load increases. Scalability tests show that the system scales well to a maximum of 1000 devices with a linear increase in processing time. The provenance tracking system is very stable and its accuracy is 85%.

C. Limitation

Minimal Real-World Testing: The research is largely based on simulated data and simulated IoT setups, which are complex and challenging. The real-world IoT system, especially concerning network instability, heterogeneity of data, and unanticipated data transformations [26].

Scalability Limitations: The study demonstrates the ability to scale with a small number of devices up to 1000, but could fail to demonstrate how the scale impacts extreme large-scale IoT faces thousands or millions of devices, with possible results of performance bottlenecks that may not be seen in this controlled scenario.

V. FUTURE RESEARCH AND CONCLUSION

A. Future Research

Future studies can involve testing the system using real-world IoT data to evaluate the system at different network conditions and different data sources. Moreover, the performance of machine learning and greater accuracy and scalability in large-scale IoT within provenance tracking due to the exploration of current advanced techniques will be studied [27].

Formal Provenance Representation Model

Let:

- D_i= data event generated at device i
- E_i= provenance event associated with D_i
- $G = (V, E)$= Directed Acyclic Graph representing provenance

Each data event is mapped as:

$$D_i \rightarrow E_i \rightarrow G$$

Provenance integrity is defined as:

$$Integrity = \frac{\text{Correctly Reconstructed Events}}{\text{Total Events}}$$

Resource efficiency is defined as:

$$Efficiency = \frac{\text{Provenance Accuracy}}{\text{CPU Usage} + \text{Memory Usage}}$$

This formulation enables quantification of the trade-off between resource utilization and provenance accuracy in distributed environments.

B. Conclusion

This paper introduces a Lightweight Distributed Provenance Framework (LDPF) that enables efficient and scalable lineage tracking in resource-constrained Edge and IoT environments. The system had done well in simulation and was efficient in terms of CPU and memory consumption and had a scaling capability to 1000 devices. There was a high provenance integrity, and its accuracy was 85%. Nevertheless, performance of the system can be affected by real-life situations when it is employed with bigger networks, different types of data and impossible to predict network conditions. The paper mentions options to continue to improve the work, such as using real-time data and working on machine learning methods to gain better tracking and detecting anomalies.

VI. REFERENCES

[1] Siddiqui, M.S., Syed, T.A., Nadeem, A., Nawaz, W. and Albouq, S.S., 2020. BlockTrack-L: A lightweight blockchain-based provenance message tracking in IoT. *International Journal of Advanced Computer Science and Applications*, *11*(4).

[2] Kamal, M., 2018. Light-weight security and data provenance for multi-hop Internet of Things. *IEEE Access*, *6*, pp.34439-34448.

[3] Aman, M.N., Basheer, M.H. and Sikdar, B., 2020. A lightweight protocol for secure data provenance in the Internet of Things using wireless fingerprints. *IEEE Systems Journal*, *15*(2), pp.2948-2958.

[4] Dai, D., Chen, Y., Carns, P., Jenkins, J. and Ross, R., 2017, September. Lightweight provenance service for high-performance computing. In *2017 26th International Conference on Parallel Architectures and Compilation Techniques (PACT)* (pp. 117-129). IEEE.

[5] Lomotey, R.K., Pry, J.C. and Chai, C., 2018. Traceability and visual analytics for the Internet-of-Things (IoT) architecture. *World Wide Web*, *21*(1), pp.7-32.

[6] Hu, R., Yan, Z., Ding, W. and Yang, L.T., 2020. A survey on data provenance in IoT. *World Wide Web*, *23*(2), pp.1441-1463.

[7] Suhail, S., Hong, C.S., Lodhi, M.A., Zafar, F., Khan, A. and Bashir, F., 2018, January. Data trustworthiness in IoT. In *2018 International Conference on Information Networking (ICOIN)* (pp. 414-419). IEEE.

[8] Rahman, M.A., Hossain, M.S., Islam, M.S., Alrajeh, N.A. and Muhammad, G., 2020. Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach. *Ieee Access*, *8*, p.205071.

[9] Elsaleh, T., Enshaeifar, S., Rezvani, R., Acton, S.T., Janeiko, V. and Bermudez-Edo, M., 2020. IoT-Stream: A lightweight ontology for internet of things data streams and its use with data analytics and event detection services. *Sensors*, *20*(4), p.953.

[10] Siddiqui, M.S., Rahman, A. and Nadeem, A., 2019. Secure data provenance in IoT network using bloom filters. *Procedia Computer Science*, *163*, pp.190-197.

[11] Lautert, F., Pigatto, D.F. and Gomes, L., 2020, July. A fog architecture for privacy-preserving data provenance using blockchains. In *2020 IEEE Symposium on Computers and Communications (ISCC)* (pp. 1-6). IEEE.

[12] Olufowobi, H., Engel, R., Baracaldo, N., Bathen, L.A.D., Tata, S. and Ludwig, H., 2016, October. Data provenance model for internet of things (iot) systems. In *International Conference on Service-Oriented Computing* (pp. 85-91). Cham: Springer International Publishing.

[13] Nikouei, S.Y., Chen, Y., Song, S., Choi, B.Y. and Faughnan, T.R., 2019. Toward intelligent surveillance as an edge network service (isense) using lightweight detection and tracking algorithms. *IEEE Transactions on Services Computing*, *14*(6), pp.1624-1637.

[14] Nwafor, E., Campbell, A., Hill, D. and Bloom, G., 2017, August. Towards a provenance collection framework for internet of things devices. In *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)* (pp. 1-6). IEEE.

[15] Siddiqui, M.S., Rahman, A., Nadeem, A. and Alzahrani, A.M., 2019. Secure data provenance in internet of things based networks by outsourcing attribute based signatures and using bloom filters. *International Journal of Advanced Computer Science and Applications*, *10*(5).

[16] Von Leon, D., Miori, L., Sanin, J., El Ioini, N., Helmer, S. and Pahl, C., 2018. A lightweight container middleware for edge cloud architectures. In *Fog and edge computing: principles and paradigms* (pp. 145-170). Wiley.

[17] El Ioini, N. and Pahl, C., 2018, October. Trustworthy orchestration of container based edge computing using permissioned blockchain. In *2018 Fifth International Conference on Internet of Things: Systems, Management and Security* (pp. 147-154). IEEE.

[18] Pahl, C., El Ioini, N., Helmer, S. and Lee, B., 2018, April. An architecture pattern for trusted orchestration in IoT edge clouds. In *2018 third international conference on fog and mobile edge computing (FMEC)* (pp. 63-70). IEEE.

[19] Ranjan, R., Rana, O., Nepal, S., Yousif, M., James, P., Wen, Z., Barr, S., Watson, P., Jayaraman, P.P., Georgakopoulos, D. and Villari, M., 2018. The next grand challenges: Integrating the internet of things and data science. *IEEE Cloud Computing*, *5*(3), pp.12-26.

[20] Von Leon, D., Miori, L., Sanin, J., El Ioini, N., Helmer, S. and Pahl, C., 2018. A performance exploration of architectural options for a middleware for decentralised lightweight edge cloud architectures. In *IoTBDS 2018: Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security; Funchal, Madeira, Portugal, 19-21 March 2018*. SciTePress.

[21] Merenda, M., Porcaro, C. and Iero, D., 2020. Edge machine learning for ai-enabled iot devices: A review. *Sensors*, *20*(9), p.2533.

[22] Appelbaum, D., 2016. Securing big data provenance for auditors: The big data provenance black box as reliable evidence. *Journal of emerging technologies in accounting*, *13*(1), pp.17-36.

[23] Scolati, R., Fronza, I., El Ioini, N., Samir, A., Barzegar, H.R. and Pahl, C., 2019, May. A containerized edge cloud architecture for data stream processing. In *International Conference on Cloud Computing and Services Science* (pp. 150-176). Cham: Springer International Publishing.

[24] Peterka, T., Bard, D., Bennett, J., Bethel, E., Oldfield, R., Pouchard, L., Sweeney, C. and Wolf, M., 2019. *ASCR workshop on in situ data management: Enabling scientific discovery from diverse data sources*. US Department of Energy (USDOE), Washington, DC (United States). Office of Science.

[25] Nakkar, M., Altawy, R. and Youssef, A., 2020. Lightweight broadcast authentication protocol for edge-based applications. *IEEE Internet of Things Journal*, *7*(12), pp.11766-11777.

[26] Sekaran, R., Patan, R., Raveendran, A., Al-Turjman, F., Ramachandran, M. and Mostarda, L., 2020. Survival study on blockchain based 6G-enabled mobile edge computation for IoT automation. *IEEE access*, *8*, pp.143453-143463.

[27] Bansal, M., Chana, I. and Clarke, S., 2020. A survey on iot big data: current status, 13 v's challenges, and future directions. *ACM Computing Surveys (CSUR)*, *53*(6), pp.1-59.