Performance and Security Analysis of Privacy-Preserved IoT Applications

BRIANNA DRÎNGĂ, University of Twente, The Netherlands

The Internet of Things (IoT) is gradually becoming ubiquitous in various domains of our daily lives, its applications ranging from smart homes and wearable devices to healthcare and other sectors. IoT applications generate a large amount of data, which requires privacy preservation due to its sensitive nature. Like other web-based information systems, IoT must cope with the variety of Cyber Security and privacy risks that now disrupt organizations and have the potential to hold entire industries and even countries' data for ransom. However, ensuring privacy preservation in IoT applications is challenging, as it involves protecting data while maintaining the desired level of performance and security. This thesis aims to analyze the privacy-preserving methods employed in IoT applications, conduct a performance analysis of these techniques based on latency and throughput, and assess the security of these techniques against various attacks.

Additional Key Words and Phrases: privacy-preserving, security, performance, IoT

1 INTRODUCTION

The Internet of Things (IoT) is a network of interconnected physical devices and objects embedded with sensors, software, and connectivity capabilities to enable the exchange of data and information over the Internet [1]. Currently, around 31 billion "things" are connected, and it is estimated that this number will reach 75 billion by 2025 [2], covering a broad spectrum of applications like home automation [3], wearables [4], transportation [5] or augmented reality [6]. With the increasing use of IoT devices and applications, the growing number of cyberthreats targeting these devices and their applications has increased the need for privacy preservation in IoT environments [7], as the widespread adoption of IoT technology has raised significant concerns about the loss of control over the collection and sharing of personal data. Therefore, it is crucial to protect sensitive data while ensuring the desired level of performance and security. It is also essential to examine the privacy-preserving techniques utilized in IoT applications outlined in the existing literature. This analysis is essential in developing effective privacy-preserving IoT applications that are optimized for performance and security.

This thesis proposal seeks to conduct a thorough examination of privacy-preserving methods utilized in IoT applications, encompassing an analysis of their performance in terms of latency and throughput. Additionally, it aims to evaluate the security of these techniques against various forms of attacks for diverse use cases within IoT applications.

The findings of this study will contribute to the advancement of more efficient and secure privacy-preserving methods for IoT applications. The outcomes will prove advantageous to organizations and individuals utilizing IoT applications and devices, as they will be able to safeguard their data more efficiently and effectively.

1.1 Background

The first primitive IoT device was a remotely controllable toaster, introduced in 1990 as a proof-of-concept [8]. A decade later, the RFID-based item identification system [9] marked the first large-scale smart device application. Nowadays, IoT devices, e.g., thermostats autonomously adjusting temperatures, already become the industry standard, and hundreds of new devices are connected to the Internet every minute [10]. However, security is a crucial concern within IoT [11], and as the risks of frequent attacks persist, research on IoT security has gained increasing popularity [12].

1.2 Problem Statement and Motivation

This section outlines the problem statement motivating the research questions outlined in Section 1.3.

The increasing number of connected devices in the Internet of Things (IoT) has led to the generation of vast amounts of sensitive data, making security and privacy preservation of utmost importance. These devices and their applications have become integral to our daily lives, from monitoring our health to controlling home appliances. However, the ease of access to these devices makes them vulnerable to attacks by malicious actors who can exploit their weaknesses to gain access to sensitive data or control the devices themselves.

Approximately 30 years after the birth of IoT, IoT security challenges have become increasingly significant. The widespread use and interconnectivity of IoT devices make them vulnerable to cyberattacks, which can significantly impact multiple stakeholders [2]. More traditional Information Technology (IT) security goals mainly focused on ensuring confidentiality, integrity, and accountability of systems and messages. However, when applied to IoT devices, these measures present limitations, e.g., due to computing power [13].

IoT devices frequently exhibit security vulnerabilities that are challenging to remove. HP's report revealed that 70% of IoT products have security vulnerabilities; each device contains 25 on average. The most well-known incident occurred in 2016 when the Mirai virus took over hundreds of thousands of IoT devices and used them to build botnets. It carried out Tbps-level denial-of-service (DoS) attacks on targets such as the DNS service provider Dyn, creating major problems such as partial Internet outages in the United States [14].

Therefore, the increasing prevalence of Internet of Things devices in various aspects of our lives has generated a large amount of sensitive data, which amplified the need for privacy-preserving

TScIT 39, July 7, 2023, Enschede, The Netherlands

IoT, as IoT systems that are not properly controlled, can pose a threat to user privacy and even cause physical harm if the sensors, actuators, or other connected devices are exploited for malicious purposes [2].

1.3 Aim and Research Questions

This research aims to evaluate the effectiveness of existing privacy-preserving techniques used in IoT applications by comprehensively analyzing their performance and security against different attacks. The ultimate goal of the project is to propose an optimized privacy-preserving technique for IoT applications based on the performance and security analysis conducted.
The problem statement will lead to the following research questions:
RQ1: What are the current privacy-preserving techniques deployed in IoT applications presented in the literature?
RQ2: How do different privacy-preserving techniques used in IoT applications perform regarding latency and throughput?
RQ3: What are the attacks and the vulnerabilities of the current privacy-preserving techniques against various attacks such as denial of service (DoS) attacks and man-in-the-middle (MITM) attacks?
RQ4: Can an optimized privacy-preserving technique be proposed for IoT applications that offer better performance and security compared to the existing techniques?
In conclusion, research questions 1, 2, and 3 will be answered through the literature review. The last research question will be answered through the proposed solutions for addressing the limitations of the current privacy-preserving techniques.
The rest of this paper will be structured as follows. First, in section 2, we explore related work covering research evaluations and findings on privacy-preserving techniques deployed in IoT applications. In Section 3.2, we detail the setup of our experiments, and in 3.1, we describe the software tools needed for conducting the experiment. Moreover, in 4, we present the results achieved from our experiment and assess the performance of different privacy-preserving techniques used in IoT applications regarding latency and throughput. Section 5 goes into a critical analysis of our results, addresses the limitations, performs a comparative performance analysis before and after preserving privacy, and proposes potential solutions to address the vulnerabilities discovered. Finally, Section 6 concludes the research and presents areas of interest for further research.

2 RELATED WORK

This section highlights related work in privacy-preserving techniques for IoT applications.
Several published surveys in the literature have addressed security in IoT from different angles. Ziegeldorf et al. [15] analyzed the privacy issues in IoT, focusing on classifying the various privacy threats and pointing out the challenges in privacy-preserving IoT. The authors summarize existing privacy threats into seven categories and review them in light of the evolving IoT, as the article stresses the need for privacy-aware solutions for the IoT that balance business interests and customers' privacy requirements.
Authors in [16] proposed a Homomorphic Encryption Protocol that allows computations to be performed on encrypted data without the need to decrypt it. The research involves a performance analysis that shows that this protocol is useful for preserving the privacy of the data while allowing for useful computations to be performed on it. The main advantage is that data can be analyzed without exposing the underlying data to anyone. However, the downside of this protocol is that it can be computationally expensive, leading to higher latency and energy consumption. The performance analysis done by the authors led to an optimization to reduce the secret key size.
Another privacy-preserving technique presented in literature is the Differential Privacy Protocol in the work by Dwork [17]. The authors conducted a comprehensive survey of Differential Privacy Protocol and its applications in privacy-preserving data analysis, the advantage of this protocol being that it provides a rigorous mathematical framework for ensuring privacy by masking the identity of individuals, however, at the expense of decreased accuracy due to added noise. The research concluded that, although ameliorated to some extent using Gaussian noise, the noise grows with the complexity of the queries applied to the database. Therefore, the paper concluded that this increase is essential for privacy guarantees and should be a subject of ongoing research.
The research done by Lindell and Pinkas in [18] illustrated another privacy-preserving protocol by conducting a privacy and performance analysis of the Secure Multi-party Computation (MPC) Protocol; this protocol allows multiple parties to jointly compute a function without revealing their private inputs to each other. The authors showed that MPC had made significant progress in recent years and has become fast enough and recognized by the industry to be deployed in practice. However, deploying MPC requires expertise, and additional breakthroughs are needed to make it practical for large data sets and complex problems.
Sharaf-Dabbagh et al. [19] proposed a novel authentication framework for IoT environments based on device fingerprinting techniques. The proposed model overcomes the limitations of existing device fingerprinting methods, such as assuming fixed fingerprints, by leveraging transfer learning techniques to accurately identify emulation attacks and to distinguish between abnormal fingerprints caused by environmental factors and those resulting from attacks. The simulation results illustrate an improvement in authentication accuracy by up to 8% compared to conventional authentication techniques. Overall, the proposed framework is a step towards developing IoT-specific security and privacy solutions, but further research is needed to address the remaining challenges in IoT security.

3 RESEARCH METHODOLOGY

This section outlines the steps taken to address the research questions (RQs) mentioned in Section 1.3 and provides an overview of the materials and tools used in the experiment.

For RQ1, a literature review was conducted to identify existing privacy-preserving techniques in IoT applications. This included examining potential security threats, vulnerabilities, and mitigation strategies. We created a simulated IoT environment, as described in Section 3.1, where privacy-preserving techniques such as differential privacy and data anonymization were implemented.
RQ2 involved evaluating the performance of the privacy-preserving techniques in the simulated IoT environment. Data on latency and throughput was collected under various scenarios, considering the number of devices and data size. Statistical analysis techniques were applied to compare the performance metrics of each technique. The findings of the experiment can be found in Section 4.
For RQ3, the security of the privacy-preserving techniques was analyzed against different attack scenarios. Eavesdropping, data tampering, and Man-in-the-Middle (MITM) attacks were simulated in the IoT environment. The impact of these attacks on data privacy and security was assessed, and the effectiveness of the privacy-preserving techniques against the various attack scenarios was evaluated. Based on the security analysis, potential improvements to the privacy-preserving techniques were identified, such as enhancing the encryption algorithm and incorporating additional security measures to prevent such attacks.
Lastly, RQ4, which aimed to propose an optimized privacy-preserving technique for IoT applications, was not achieved. The research assessed the effectiveness of privacy-preserving techniques in mitigating different attack scenarios but an optimized method is still to be proposed, as the experimental results showed that privacy-preserved IoT applications still present security vulnerabilities.

3.1 Measurement tools

This subsection describes the tools used in conducting the experiment and the packages used to implement each privacy-preserving technique.
The experiment was conducted on a Dell XPS laptop, with 16GB RAM, and Intel Core i7. The high RAM capacity allowed for the efficient execution of resource-intensive operations within the Mininet framework, ensuring smooth emulation and accurate measurements. The tools used in this research are software packages and all of the tools are contained in one Virtual Machine (VM) running in VMware Workstation 17 Player.
To create the network topology of my IoT environment, the following packages and modules were used:

- Mininet: provides the network emulation framework to simulate a realistic network environment by creating virtual hosts, switches, and links.
- Mininet Topo: allows defining the custom network topologies by creating the required hosts and switches and establishing the connections between them. Within the Mininet package, we used mininet.net, mininet.node, and mininet.link modules to configure the network and define the connections between devices.
- Mininet Link: enables to establishment and control of the links between the hosts and the switch, specifying parameters like bandwidth, delay, and packet loss.

To implement the privacy-preserved techniques in my simulated IoT environment, we used the following packages and libraries:

- Diffprivlib: Python library for applying differential privacy techniques to my network data. The library provided me with a set of algorithms for adding noise, perturbing data, and aggregating statistics while ensuring privacy guarantees. By leveraging these functionalities, we could protect sensitive information exchanged between devices by introducing controlled randomness. Moreover, *diffprivlib* allowed us to quantify the privacy guarantees and control the level of privacy and accuracy trade-offs through privacy parameters.
- SEAL-Python: Python version of SEAL library for applying fully homomorphic encryption (FHE); it enables computations to be performed on encrypted data without decryption, preserving the privacy and security of sensitive information. We used the library's functionalities to encrypt the received data packet from each IoT device, perform homomorphic addition operations on the encrypted data, and decrypt the result to obtain the processed data.
- Anonimyzedf: Python library for applying Data Anonymization, offering methods for generalization, suppression, randomization, and perturbation of data attributes. Using the library, we could remove and obfuscate personally identifiable information (PII) from the sample data we generated, to prevent the identification of individuals.

3.2 Measurement environment

This subsection explains how we set up the measurement environment which comprises a virtual network with multiple interconnected hosts, switches, and controllers, emulating the behavior of a real IoT network.
To create the IoT simulation, we established the measurement environment using Mininet, an open-source network emulator providing a flexible and controllable platform for creating a simulated IoT network environment, enabling precise measurements and analysis. To set up the environment in Mininet, we followed the instructions for Windows provided by its documentation and first downloaded the Mininet VM Image. As Mininet is primarily designed for Linux-based operating systems, we set up the Virtual Machine (VM) running Ubuntu on my Windows host machine using the VMware Workstation 17 Player virtualization system.
To ensure a clean and isolated environment, we created a virtual environment using *virtualenv* where we installed the necessary libraries and packages required for security analysis, performance measurement, and IoT simulation, utilizing Python's package manager, pip. For more details regarding the use of packages, see Section 3.1. To begin the experiment, we started the virtual machine and logged in to Mininet. Once logged in, we used a text editor to write a Python script to generate the network topology, apply privacy-preserving techniques, simulate attacks, and measure performance metrics. We navigated to the directory where

the script was located in the Mininet terminal and ran it using the Python interpreter after importing the necessary packages described in Section 3.1. This established measurement environment was the foundation for conducting the security and performance analysis of privacy-preserved IoT applications.

4 RESULTS

This section outlines the results of the methodology described in Section 3, based on evaluating the effectiveness of privacy-preserving techniques in terms of throughput and latency.
All privacy-preserving techniques were applied to a start topology network of 20 IoT devices connected via a switch, configured as described in Section 3.2. Table 1 gives an overview of the privacy-preserving techniques employed in our IoT environment, along with the libraries used for implementation. We created a Python script for each privacy-preserving technique applied to our simulated IoT environment, yielding average values for throughput and latency. To ensure the high accuracy of these results, we ran each script 100 times by adding a loop to iterate over the desired number of runs. After each run, the measurements of throughput and latency are stored in separate lists. Once the loop completes, the script calculates the average throughput and latency by summing up the results of the measurements and dividing by the total number of runs.
To evaluate the effectiveness of each privacy-preserving technique, we created different scripts using a consistent network topology, described in Section 3.2, simulated a different attack scenario for each script, and measured the network's performance in terms of throughput and latency. Subsequently, we implemented the selected privacy-preserving techniques and repeated the attack simulations, again measuring the network's throughput and latency. We compared the results before and after privacy-preserving to assess whether the privacy-preserving methods applied to the environment can be used in preventing or mitigating the simulated attack and to which extent enhancing security negatively impacts performance.
The first privacy-preserving technique applied to our simulated IoT environment is Differential Privacy (DP). DP ensures that the privacy of individual device values is protected while still allowing statistical analysis. By adding noise to the data, we can safeguard sensitive information even in the presence of potential adversaries or data breaches. To perform the experiment, we ran the Python script in the Mininet terminal, which applies DP using the *diffprivlib* package explained in Section 3.1, along with Numpy, which is used to generate random data and perform mathematical operations. The Laplace mechanism is specifically designed to provide privacy guarantees by introducing noise proportional to the sensitivity of the data and inversely proportional to the desired privacy parameter (epsilon). This mechanism provides strong privacy guarantees and has been widely adopted in various privacy-preserving applications.
To calculate the throughput, we simulated a specific number of differential privacy operations, measuring the elapsed time and dividing the number of operations by the elapsed time. For measuring the latency, the code performs the same set of operations and measures the time taken for each operation. The average latency is calculated by averaging the measured times.
To assess the effectiveness of DP in mitigating an attack, we implemented a data tampering attack, which we simulated by modifying a specific set of data values collected from the IoT devices before applying the privacy-preserving techniques. By leveraging the inherent noise injection and aggregation steps in the DP mechanism, we observed that the impact of the data tampering attack was significantly reduced. The added noise, derived from the Laplace distribution, acted as a protective layer, rendering the tampered values indistinguishable from the original data. As a result, applying DP effectively masked the tampered values and prevented adversaries from accurately determining the original data, and even when the tampered data was aggregated and analyzed, the privacy guarantees provided by DP remained intact.
Data Anonymization is the second privacy-preserving technique applied to our virtual network of IoT devices. By applying various anonymization techniques, such as generalization, suppression, or randomization, data is transformed in a way that makes it difficult or impossible to identify specific individuals or sensitive attributes directly. We generated sample data stored in a .txt file on each IoT device. The data represents personal information such as names, ages, and salaries, on which we apply anonymization techniques to transform it. The script initiates data transfer among devices by simulating network traffic, such as sending ping requests.
To calculate the throughput, the code measures the data transfer time between devices. It starts a timer before initiating data transfer among devices, and once the data transfer is complete, the timer is stopped, and the elapsed time is recorded. The average throughput is calculated by dividing the amount of data transferred by the elapsed time. For measuring the latency, the script captures the time it takes for each ping request to receive a response, representing the round-trip time. We extract the latency values from the ping responses and calculate the average latency between all pairs of devices in the network.
To evaluate the efficacy of data anonymization in securing an IoT environment, we implemented a man-in-the-middle (MITM) attack that involved intercepting and manipulating the communication between IoT devices in the network. We employed packet sniffing techniques to capture the network traffic passing through our environment and, to intercept and analyze the packets exchanged between the IoT devices, we used Scapy, a packet manipulation library written in Python. By intercepting the packets, we were able to modify their content, introducing unauthorized changes to the data being transmitted. We forwarded the manipulated packets to their intended recipients, simulating the actions of a malicious actor. This allowed us to evaluate the vulnerabilities and limitations of data anonymization techniques in the face of MITM attacks.
The application of data anonymization proved to be a valuable defense mechanism in mitigating the impact of the MITM attack. While effective in protecting sensitive information, can still present certain vulnerabilities that adversaries may exploit. One such vulnerability is the potential re-identification of individuals based on the released anonymized data. Although generalization,

suppression, and randomization techniques are applied to disguise personal information, advanced de-anonymization attacks, such as inference or background knowledge attacks, can exploit patterns, correlations, or external datasets to identify individuals. However, even if an attacker gains unauthorized access to the anonymized data, the information lacks direct identifiers or sensitive details, making it difficult for the attacker to obtain sensitive information from the intercepted traffic.

The third privacy-preserving technique we employed in our simulated IoT environment is Fully Homomorphic Encryption (FHE). We performed data processing tasks on the switch device by simulating the processing of packets received from IoT devices. To demonstrate FHE, we created an instance of the SEAL context with appropriate parameters for encryption and set up the necessary encryption keys. Using the encryptor, we encrypted the packet data received from each IoT device and performed homomorphic computations such as addition and multiplication to the encrypted data. Finally, we decrypted the result to obtain the processed data. The main advantage of homomorphic encryption is that it allows data to be analyzed without exposing the underlying data to anyone. However, the downside of this protocol is that it can be computationally expensive, leading to higher latency and energy consumption.

To assess the throughput, we measured the number of FHE operations (e.g., encryption, decryption, or computations such as addition) that can be processed per unit of time. We executed the selected workloads multiple times and recorded the time taken for each execution. The average throughput was calculated based on these measurements. We measured latency by measuring the time it takes for a single FHE operation to complete and, subsequently, we calculated the average latency by executing the selected workloads and recording the time taken for each operation.

To evaluate the efficacy of FHE in securing IoT applications, we performed an eavesdropping attack by employing network sniffing tools to intercept and capture the encrypted data transmitted between the IoT devices. Despite the encrypted nature of the data, traditional encryption methods would still be susceptible to eavesdropping attacks, as the ciphertext could be intercepted and stored for potential decryption attempts. However, with the application of FHE, the intercepted encrypted data remained secure and resistant to decryption attempts by the eavesdropper. FHE allowed computations to be performed directly on the encrypted data, enabling secure data processing without the need for decryption. This capability provided an additional layer of protection against eavesdropping attacks, as the sensitive information remained encrypted throughout the entire data lifecycle. By demonstrating the efficacy of FHE in mitigating eavesdropping attacks, our work highlighted the importance of leveraging advanced cryptographic techniques to ensure the confidentiality and privacy of IoT data, even in the presence of malicious actors. Therefore, all results showed that the overall security and performance of the network were preserved through the application of differential privacy against a data tampering attack, data anonymization against a MITM attack, and FHE against eavesdropping. When DP and FHE were applied to the network under attack, the average throughput resulting from the experiment increased compared to the results given by the same environment without privacy preservation. Moreover, the average latency decreased, hence how privacy-preserving methods secure IoT environments, but do not always ensure full protection against attacks. In the case of data anonymization, a MITM attack was still possible, although its impact was improved through privacy preservation.

By considering the trade-off between privacy and performance, our findings demonstrate that the incorporation of privacy-preserving methods in the IoT environment provided a valuable layer of protection against data tampering attacks while ensuring that the system's throughput and latency remained within acceptable limits.

Table 1. Overview of privacy-preserved techniques applied to the IoT environment.

Privacy-preserving method	Differential Privacy	Data Anonymization	Fully Homomorphic Encryption
Library	NumPy	AnonymizeDF	SEAL-Python
Language	Python	Python	Python
Additional Libraries	-	pandas	-

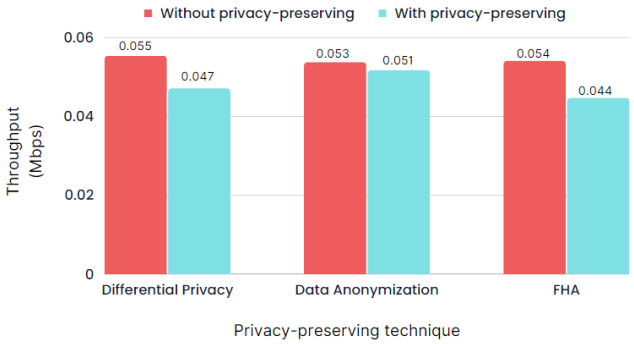

Fig. 1. Average throughput in the IoT environment with and without the implementation of privacy-preserving techniques.

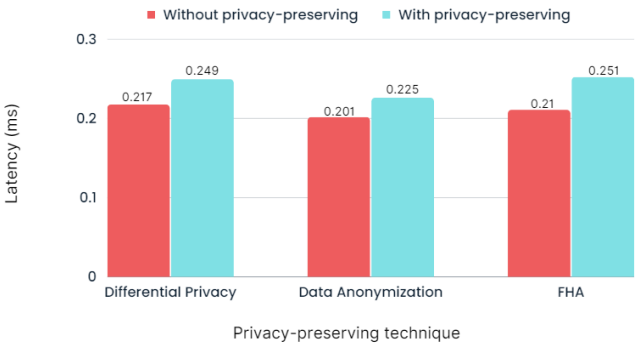

Fig. 2. Average latency in the IoT environment with and without the implementation of privacy-preserving techniques.

5 DISCUSSION

This section will highlight the findings of our research based on the experimental results. The conducted experiment proves the impact of privacy-preserving techniques on both security and performance. On the one hand, the attacks performed in the simulated environment suggest that Differential Privacy, Data Anonymization, and Fully Homomorphic Encryption strengthen the security of the simulated IoT environment. On the other hand, the evaluation of the performance metrics in terms of throughput and latency of the IoT environment shows how privacy-preserving techniques decrease performance, as it has the effect of increasing latency and decreasing throughput.

When applying Differential Privacy, the results prove that adding noise affects the data's accuracy and utility. The noise introduced by the differential privacy mechanism causes distortions in the data, resulting in a reduction in throughput, as illustrated by Fig. 1, and an increase in latency, as shown by Fig. 2. The amount of noise introduced impacts the trade-off between security and performance. Higher noise levels resulted in greater privacy protection but at the cost of reduced data quality and throughput. Under the simulation of an attack, when differential privacy was applied, the findings mentioned in Section 4 illustrate that differential privacy effectively prevents a data tampering attack. Based on the security and performance analysis conducted, a potential improvement to DP algorithms would involve developing an optimized mechanism for allocating the amount of noise added to the data. Increasing the amount of noise allows for stronger privacy guarantees but impacts the accuracy of the data analysis. On the other hand, a smaller amount of noise may provide better data utility but will compromise privacy protection. Therefore a potential solution would focus on developing a dynamic budget allocation and adaptive privacy mechanisms to optimize the trade-off between privacy and data utility.

Regarding Data Anonymization, the impact on the throughput and latency depends on the computational complexity of the techniques used for transforming and modifying data to remove identifiable information. Since data anonymization ensures that sensitive information is protected by replacing identifiable data with anonymized values, the throughput of the network decreases due to the additional processing required for anonymization. Based on the results shown by 1, the reduction in throughput is lower than in the other two cases of privacy-preserved methods. The latency is higher due to the additional computational overhead introduced by anonymization. Under a man-in-the-middle (MITM) attack, data anonymization is not fully effective in mitigating the attack. This results in reduced throughput and increased latency, as legitimate communication is hindered or altered by the attacker's actions. Based on the experimental results, the potential improvements of data anonymization could involve privacy-preserving data sharing by developing more secure protocols and frameworks for sharing anonymized data, such that privacy is maintained even when data is shared between different entities or organizations. Another point of improvement could be to combine multiple anonymization techniques to create a multi-layered approach when securing data.

Applying Fully Homomorphic Encryption to the network above has a significant impact on the throughput and latency based on the findings presented in Section 4. FHE is a computationally intensive process that involves complex mathematical operations on encrypted data. These operations introduce additional computational overhead, increasing processing time and reducing throughput. The encryption and decryption processes add latency to the data transfer, further increasing the overall latency of the network. The impact on throughput and latency is the highest among all three privacy-preserving methods employed in this research, as illustrated by Fig. 1 and Fig. 2. To our knowledge, this is the first work that makes a comparison between the average throughput and latency instead of comparing the throughput given by each homomorphic operation. Regarding potential improvements, FHE could be strengthened through performance optimization of FHE schemes, to reduce the computational overhead associated with performing computations on encrypted data, especially for resource-constrained IoT devices. Moreover, to address potential vulnerabilities in FHE key management systems, secure key distribution, storage, and revocation should be ensured. Further enhancements could be made to enhance the security of the encryption scheme by addressing potential side-channel attacks on FHE implementations, such as timing or power analysis attacks.

Therefore, all the experimental results illustrate how the network's performance under attack is better when a privacy-preserving technique is used. Differential privacy is effective against data tampering attacks, as FHE efficiently mitigates eavesdropping attacks. In the case of a MITM attack, the data anonymization proved to reduce the impact of the attack but did not fully mitigate it. Hence privacy-preserved IoT applications are still vulnerable to attacks and require further improvements and security measures.

6 CONCLUSION

In conclusion, this study has highlighted the growing need for enhanced security measures in the realm of IoT, driven by the widespread adoption of IoT devices and the consequent generation of sensitive data. The research has undertaken a comprehensive analysis of existing privacy-preserving methods, evaluating their performance in terms of latency and throughput while also assessing their efficacy in safeguarding against various types of attacks.

The experimental findings have revealed that privacy-preserving techniques offer only partial effectiveness in mitigating or preventing attacks within IoT applications. Furthermore, it has been observed that these techniques can lead to decreased performance in terms of throughput and latency. Consequently, there is room for improvement in privacy-preserving strategies to address the vulnerabilities still existing within IoT applications, such as data tampering, Man-in-the-Middle (MITM) attacks, and eavesdropping.

In future work, it is imperative to explore potential enhancements for privacy-preserving techniques to bolster their resilience

against attacks. The research underscores the ongoing necessity to fortify privacy-preserved IoT applications and emphasizes the importance of devising robust countermeasures to combat evolving security threats.

REFERENCES

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[2] Eryk Schiller, Andy Aidoo, Jara Fuhrer, Jonathan Stahl, Michael Ziörjen, and Burkhard Stiller. Landscape of iot security. *Computer Science Review*, 44:100467, 2022.

[3] Manuel Suárez-Albela, Paula Fraga-Lamas, Tiago M Fernández-Caramés, Adriana Dapena, and Miguel González-López. Home automation system based on intelligent transducer enablers. *Sensors*, 16(10):1595, 2016.

[4] Santiago Barro-Torres, Tiago M Fernández-Caramés, Héctor J Pérez-Iglesias, and Carlos J Escudero. Real-time personal protective equipment monitoring system. *Computer Communications*, 36(1):42–50, 2012.

[5] Paula Fraga-Lamas, Tiago M Fernández-Caramés, and Luis Castedo. Towards the internet of smart trains: A review on industrial iot-connected railways. *Sensors*, 17(6):1457, 2017.

[6] Oscar Blanco-Novoa, Tiago M Fernandez-Carames, Paula Fraga-Lamas, and Miguel A Vilar-Montesinos. A practical evaluation of commercial industrial augmented reality systems in an industry 4.0 shipyard. *Ieee Access*, 6:8201–8218, 2018.

[7] Vikas Hassija, Vinay Chamola, Vikas Saxena, Divyansh Jain, Pranav Goyal, and Biplab Sikdar. A survey on iot security: application areas, security threats, and solution architectures. *IEEE Access*, 7:82721–82743, 2019.

[8] John Romkey. Toast of the iot: the 1990 interop internet toaster. *IEEE Consumer Electronics Magazine*, 6(1):116–119, 2016.

[9] V Rajaraman. Radio frequency identification. *Resonance*, 22:549–575, 2017.

[10] Patel Mark, Shangkuan Jason, and Thomas Christopher. What's new with the internet of things? *McKinsey & Company, New York, NY, USA, Tech. Rep*, 2017.

[11] Wan Haslina Hassan et al. Current research on internet of things (iot) security: A survey. *Computer networks*, 148:283–294, 2019.

[12] Miao Yu, Jianwei Zhuge, Ming Cao, Zhiwei Shi, and Lin Jiang. A survey of security vulnerability analysis, discovery, detection, and mitigation on iot devices. *Future Internet*, 12(2):27, 2020.

[13] Sabrina Sicari, Alessandra Rizzardi, Luigi Alfredo Grieco, and Alberto Coen-Porisini. Security, privacy and trust in internet of things: The road ahead. *Computer networks*, 76:146–164, 2015.

[14] Constantinos Kolias, Georgios Kambourakis, Angelos Stavrou, and Jeffrey Voas. Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84, 2017.

[15] Jan Henrik Ziegeldorf, Oscar Garcia Morchon, and Klaus Wehrle. Privacy in the internet of things: threats and challenges. *Security and Communication Networks*, 7(12):2728–2742, 2014.

[16] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 169–178, 2009.

[17] Cynthia Dwork. Differential privacy: A survey of results. In *Theory and Applications of Models of Computation: 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings 5*, pages 1–19. Springer, 2008.

[18] Yehuda Lindell. Secure multiparty computation (mpc). *Cryptology ePrint Archive*, 2020.

[19] Yaman Sharaf-Dabbagh and Walid Saad. On the authentication of devices in the internet of things. In *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–3. IEEE, 2016.